

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΑ, ΤΜΗΜΑ ΨΗΦΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ



Μεταπτυχιακή Διπλωματική

# **Ανάπτυξη οντολογίας για την βελτιστοποίηση διαχείρισης και διαλειτουργικότητας μεταξύ ετερογενών δικτύων υψηλών ταχυτήτων**

Υπ. Καθηγητής: Δρ. Π. Δεμέστιχας

Εκπόνηση:

Α. Σταθόπουλος – Λαμπρέλλης

ΜΕ/0690

Στην μητέρα μου για την παρότρυνση της να επιδιώκω πάντα κάτι καλύτερο για στην μόρφωση μου.

Στην φίλη μου Κάτια για την πολύτιμη βοήθεια και εμπύχωση της.

## Περίληψη

Ο κόσμος των ασυρμάτων επικοινωνιών διανύει μια περίοδο συνεχών αλλαγών και ανακατατάξεων, οι οποίες έχουν φέρει μια ειρηνική επανάσταση στον τρόπο λειτουργίας κάθε ασύρματης (αλλά και ενσύρματης) συσκευής και αντιστοίχως κάθε δικτύου με το οποίο αυτή διασυνδέεται. Έτσι, η μετάβαση από τα δίκτυα δεύτερης γενιάς στα δίκτυα τρίτης, αλλά και πέραν της τρίτης γενιάς, έχει φέρει σειρά νέων υπηρεσιών και εφαρμογών. Αυτό διότι βασικός στόχος όλων των προαναφερθέντων αλλαγών είναι η παροχή καινοτόμων υπηρεσιών στο χρήστη.

Με αφετηρία το παραπάνω σκεπτικό, η παρούσα διπλωματική κινείται στο χώρο των τηλεπικοινωνιών και αποσκοπεί στο να συμβάλει στη διαμόρφωση μια κοινής βάσης, ενός κοινού τρόπου αναφοράς, μιας κοινής «οντολογίας», με χρήση της οποίας θα είναι ευκολότερη κάθε ενασχόληση με μια νέα (ή και με μια παλιά) τεχνολογία επικοινωνιών.

Για αυτό το σκοπό, στο πρώτο κεφάλαιο της εργασίας παρουσιάζονται κάποιες γενικές έννοιες για τα ενσύρματα και ασύρματα δίκτυα, όπως οι μεταπομπές, οι τρόποι κωδικοποίησης δεδομένων, αλλά και οι εφαρμογές των τηλεπικοινωνιών που χρησιμοποιούμε σήμερα, αλλά και κάποιες νέες που προς ώρα αποτελούν οράματα.

Στο δεύτερο κεφάλαιο της εργασίας παρουσιάζονται οι διάφορες τεχνολογίες δικτύων υποδομής, όπως τα LAN, WLAN, WAN, WiMAX, DVB/DAB, κλπ, ενώ παράλληλα γίνεται αναφορά στα νεοφερθέντα δίκτυα πέραν της τρίτης γενιάς (B3G) και στις αλλαγές που επιφέρουν, αλλά και τις προσμονές που εγείρουν. Παράλληλα εισάγονται οι έννοιες των δυναμικά αναδιαρθρούμενων (reconfigurable), και των γνωσιακών (cognitive) δικτύων, οι οποίες είναι εξαιρετικά χρήσιμες για κάθε ερευνητή στο χώρο των επικοινωνιών.

Κατόπιν, στο τρίτο κεφάλαιο της εργασίας γίνεται μια γενική αναφορά στις οντολογίες που αποτελούν και το βασικό αντικείμενο της παρούσης εργασίας, ως τρόπων κοινής αναφοράς σε ένα δεδομένο θέμα. Παράλληλα, παρουσιάζονται παραδείγματα από τη χρήση των οντολογιών στη φιλοσοφία, την πληροφορική και τη γενετική, ενώ παράλληλα

παρουσιάζονται μέθοδοι ανάπτυξης οντολογιών. Ως εργαλείο τέτοιων μεθόδων παρουσιάζεται η γλώσσα οντολογίας ιστού OWL.

Στο τέταρτο κεφάλαιο της εργασίας παρουσιάζεται μια οντολογία για δίκτυα B3G και δυναμικά αναδιαρθρούμενα, η οποία έχει αναπτυχθεί ως πρότυπο P1900 της IEEE. Το πρότυπο αυτό παρουσιάζεται αναλυτικά και αποσκοπείται να δοθεί μια κριτική ματιά στα χαρακτηριστικά του.

Στο πέμπτο κεφάλαιο της εργασίας παρουσιάζεται η υλοποίηση της οντολογίας του πρωτοκόλλου IEEE P1900 με την χρήση του εργαλείου Protégé και επιπρόσθετα η αντιστοίχιση του μοντέλου κάνοντας χρήση αντικειμενοστραφούς σχεδιασμού.

Τέλος, το έκτο κεφάλαιο της εργασίας περιέχει ορισμένα συμπεράσματα μαζί με προτάσεις για βελτίωση του παρουσιασθέντος προτύπου, αλλά και γενικότερες προτάσεις για το μέλλον των οντολογιών ασυρμάτων επικοινωνιών.

## Abstract

The wireless communications world is undergoing a period of continuous changes and reformations, which have brought a peaceful revolution in the way every wireless (also wired) device operates, as well as in the way it is interconnected with the backbone network. The transition, thus, from 2G networks to 3G networks, as well as to B3G networks, has brought about a series of new services and applications. In any case, this is the motivation for the aforementioned changes, i.e. the provision of innovative services to the users.

Aligned with the aforementioned thoughts, this thesis is positioned within the telecom world and aims at contributing to the adoption of a common basis, a common way of thinking, a common “ontology”, through which every involvement in a new (also existing) telecommunications technology, shall be facilitated.

For this purpose, the first chapter of the thesis presents some general concepts for wired and wireless networks, such as handoffs, coding schemes, as well as telecommunications applications used today and to be used tomorrow (While existing today as a vision).

The second chapter of the thesis presents various infrastructure network technologies, such as LAN, WLAN, WAN, WiMAX, DVB/DAB, etc. In parallel, a reference is made to the newly born B3G systems and the changes they incur, as well as the desires they emerge. Additionally, the concepts of reconfigurable and cognitive networks are introduced, which are considerably important for every researcher in the area of communications.

In the sequel, the third chapter refers to ontologies, which constitute the basic subject of this thesis, as a way to commonly refer to a given subject. Additionally, numerous examples of ontologies are presented, deriving from philosophy, information technology and genetics. Furthermore, methods of creating ontologies are presented, focusing on OWL, as a tool for such methods.

The fourth chapter of the thesis presents an ontology for B3G networks, focusing on reconfigurable networks, which has been developed as an IEEE standard under the denomination P1900. The standard is presented in detail, in order to criticize its characteristics and functionality.

The fifth chapter of the thesis presents the implementation of the ontology of IEEE 1900 with the use of Protégé plus the mapping of the ontology using object oriented theory.

The sixth chapter of the thesis includes a number of useful conclusions, as well as thoughts that could eventually improve the standard, as well as thoughts related to the future of ontologies in wireless networks.

## Περιεχόμενα

Περίληψη .....	3
Abstract .....	5
Περιεχόμενα .....	7
1.1 Γενική Εισαγωγή .....	11
1.2 Γενικές έννοιες .....	12
1.2.1 Συστήματα B3G .....	12
1.2.2 Δικτυακός εξοπλισμός (Βασικές έννοιες δικτύων) .....	12
1.2.2.1 Ενσύρματα δίκτυα .....	13
1.2.2.2 Ασύρματα δίκτυα .....	13
1.2.2.3 Κινητά δίκτυα .....	13
1.2.3 Μεταπομπές (handoffs) .....	14
1.2.3.1 Vertical HO .....	14
1.2.3.2 Horizontal HO .....	14
1.2.3.3 Hard HO .....	15
1.2.3.4 Soft HO .....	15
1.2.4 Κωδικοποιήσεις δεδομένων .....	15
1.2.4.1 Multiple In Multiple Out .....	17
1.2.5 Τύποι μετάδοσης .....	17
1.2.6 Κυψέλες (Cells) .....	18
1.2.7 Περιαγωγή (Roaming) .....	18
1.2.8 Quality of Service (QoS) .....	19
1.3 Εφαρμογές .....	19
1.3.1 Μεταφορά δεδομένων (Data) .....	20
1.3.2 Voice Over IP (VoIP) / Audio .....	20
1.3.3 Video .....	21

1.3.4 Απαιτήσεις εφαρμογών .....	21
1.3.5 Υφιστάμενοι περιορισμοί .....	23
1.4 Στόχος της πτυχιακής εργασίας .....	23
2.1 Τεχνολογίες δικτύων υποδομής .....	25
2.1.1 Τύποι δικτύων (Circuit Switching, Packet Switching, Virtual Circuit Switching) .....	25
2.1.1.1 Αξιόπιστα (TCP) και μη αξιόπιστα (UDP) πρωτόκολλα επικοινωνίας .....	27
2.1.2 Είδη δικτύων .....	28
2.1.2.1 Local Area Network (LAN) .....	28
2.1.2.2 Wide Area Network (WAN) .....	31
2.1.2.3 Wireless Local Area Network (WLAN) .....	32
2.1.2.4 WiMax .....	36
2.1.2.5 Δίκτυα 3G .....	38
2.1.2.6 Personal Area Network (PAN) .....	40
2.1.2.7 Satellite Networks .....	42
2.1.2.8 Ψηφιακά δίκτυα ευρείας εκπομπής (DVB / DAB) .....	43
2.1.2.9 Global Area Networks (GAN) .....	45
2.1.3 Συγκριτικά στοιχεία δικτύων .....	46
2.2. Χαρακτηριστικά συστημάτων B3G .....	47
2.3. Η διαχείριση δικτύων B3G .....	49
2.4. Η έννοια της δυναμικής αναδιάρθρωσης .....	51
2.5 Τα cognitive δίκτυα επικοινωνιών .....	54
2.6 Προκλήσεις σχετικές με τη χρήση του φάσματος στα δίκτυα B3G .....	56
2.7 DSNPM .....	59
2.8 Αξιολόγηση δικτύων B3G .....	61
3.1 Εισαγωγή – Τι είναι «οντολογία» .....	63
3.2 Παραδείγματα οντολογιών .....	64
3.2.1 Οντολογία και φιλοσοφία .....	64
3.2.2 Οντολογία και πληροφορική .....	65



3.2.3	Οντολογία και γενετική .....	66
3.3	Μέθοδοι ανάπτυξης .....	68
3.3.1	Γενική μεθοδολογία ανάπτυξης οντολογίας.....	68
3.3.2	Παράδειγμα - η γλώσσα οντολογίας ιστού OWL.....	71
3.3.2.1	Γενικά στοιχεία .....	71
3.3.2.2	Οι υπο-γλώσσες της OWL .....	74
4.1	Εισαγωγή .....	77
4.2	Περιβάλλον (context) 1900.4 .....	78
4.3	Σενάρια - Use Cases.....	80
4.4	Αρχιτεκτονική .....	83
5.1	Σκοπός – ευθύνες και λειτουργικές δυνατότητες της οντολογίας για δίκτυα B3G .....	91
5.1.1	Εισαγωγή .....	91
5.1.2	Σκοπός και λειτουργικές δυνατότητες οντολογίας .....	93
5.2	Το Protégé 2000.....	94
5.3	Περιγραφή και αναπαράσταση οντολογίας .....	96
5.3.1	Περιγραφή.....	96
5.3.2	Αναπαράσταση.....	96
5.3.2.1	OSM .....	99
5.3.2.2	NRM.....	99
5.3.2.3	TRM.....	102
5.3.2.4	Interface OSM - NRM.....	103
5.3.2.5	Interface NRM – RMC.....	103
5.3.2.6	Interface NRM – RRC .....	104
5.3.2.7	Interface TRM - TMC.....	104
5.3.2.8	Interface NRM-TRM.....	105
5.3.2.9	Interface TRM - TRC.....	106

5.3.2.10 Reconfiguration and measurement SAP .....	106
5.3.3 Οπτική αναπαράσταση με την χρήση του εργαλείου PROTÉGÉ.....	107
5.4 Ανάπτυξη οντολογίας σε αντικειμενοστραφή μοντέλο .....	113
5.4.1 Ανάλυση ανάπτυξης .....	114
5.4.2 Παρουσίαση βασικών λειτουργιών .....	116
5.4.3 Οπτική παρουσίαση του αντικειμενοστραφούς μοντέλου.....	121
5.5 Αξιολόγηση οντολογίας και σύγκριση με το πρότυπο P1900.....	125
6.1 Σύνοψη και συμπεράσματα .....	135
6.2 Γενικές κατευθύνσεις .....	137
7.1 Βιβλία .....	139
7.2 Άρθρα.....	139
7.3 Εργασίες / Διατριβές.....	144
7.4 URLs.....	144
8.1 Πηγαίος Κώδικας .....	146
8.2 Διάγραμμα κλάσεων της οντολογίας P1900 .....	195
8.3 Υλοποίηση κλάσης SAP.....	201

# 1. Εισαγωγή

## 1.1 Γενική Εισαγωγή

Από το 1962 όπου διατυπώθηκε η ιδέα της δικτύωσης των υπολογιστών από τους Paul Baran, Donald Davies και Leonard Kleinrock μέχρι σήμερα, ο χώρος έχει εξελιχθεί δραματικά. Η δικτύωση αν και αρχικά χρησιμοποιήθηκε για ακαδημαϊκούς και στρατιωτικούς σκοπούς, σήμερα θα μπορούσε κανείς να πει πως είναι αναπόσπαστο κομμάτι την πληροφορικής. Το Arpanet, το πρώτο δίκτυο υπολογιστών στο οποίο χρησιμοποιήθηκε η τεχνολογία της ανταλλαγής πακέτων (packet switching) εξελίχθηκε στο Internet το οποίο συνέβαλε στην ιδέα του «παγκόσμιου χωριού».

Με την πάροδο του χρόνου η τεχνολογία εξελίχθηκε σε όλους τους τομείς. Όσο η τεχνολογία εισέβαλε στην ζωή του χρήστη τόσο η ανάγκη δικτύωσης πέραν του ηλεκτρονικού υπολογιστή έκανε την εμφάνιση της. Η διάθεση συσκευών όπως Laptops, PDAs και κινητών τηλεφώνων έδωσαν την δυνατότητα κινητικότητας του χρήστη. Έτσι δημιουργήθηκαν πολλές τεχνολογίες δικτύωσης. Η κάθε μία εξασφάλιζε ένα τρόπο δημιουργίας δικτύου μεταξύ διαφόρων συσκευών. Το επόμενο λογικό βήμα ήταν η επίτευξη ενός ενιαίου δικτύου με χρήση των επιμέρους τεχνολογιών.

Ακόμη δημιουργήθηκαν πολλές υπηρεσίες οι οποίες στηριχτήκαν στην ύπαρξη δικτύου. Και αυτές εξελίχθηκαν παράλληλα με τα δίκτυα. Καταυτόν τον τρόπο οι βασικές υπηρεσίες που παρέχονταν από την δικτύωση υπολογιστών εμπλουτίστηκαν με πολυμεσικές υπηρεσίες και όχι μόνο.

Σε αυτή την πτυχιακή εργασία θα αναφερθούν τόσο οι σύγχρονες τεχνολογίες δικτύωσης όσο και οι υπηρεσίες που παρέχονται. Θα εξεταστούν οι απαιτήσεις των τελευταίων και θα αναζητηθεί μια οντολογία για την βελτιστοποίηση διαχείρισης και διαλειτουργικότητας μεταξύ ετερογενών δικτύων υψηλών ταχυτήτων.

## 1.2 Γενικές έννοιες

Κρίνεται σκόπιμο πριν την εκτενή ανάλυση του θέματος να πραγματοποιηθεί μια εισαγωγή σε κάποιες γενικές έννοιες της σύγχρονης δικτύωσης.

### 1.2.1 Συστήματα B3G

Τα συστήματα B3G (τα οποία αναφέρονται και ως 4G) είναι σύγχρονα συστήματα δικτύωσης στα οποία θα γίνει εκτενέστερη αναφορά στην συνέχεια. Ο λόγος αυτής της αναφοράς είναι πως το προαναφερθέν είναι ένα σύστημα δικτύωσης το οποίο χρησιμοποιεί πολλαπλούς τύπους δικτύων για την παροχή υπηρεσιών. Τα συστήματα B3G βασίζονται στην τεχνολογία της ασύρματης δικτύωσης και παρέχουν πολλές υπηρεσίες μέσω του Internet Protocol (IP). Χρησιμοποιούνται κυρίως από κινητά τερματικά και ενοποιούν δίκτυα data κινητών επικοινωνιών με ασύρματα δίκτυα (WLANs) όπου αυτά είναι διαθέσιμα. Βασική διαφοροποίηση τους από τα 3G δίκτυα εκτός από την ενοποίηση η οποία ήδη αναφέρθηκε είναι οι μεγαλύτερες ταχύτητες λόγω διαφορετικών τεχνολογιών (HSPA έναντι UMTS). Η παρεχόμενη ταχύτητα είναι ένα σημαντικό δεδομένο καθώς αίρει όποιους φραγμούς για την παροχή multimedia υπηρεσιών δημιουργούσαν τα δίκτυα προηγούμενων γενεών. Άλλο ένα πολύ σημαντικό χαρακτηριστικό είναι η ενσωμάτωση του Quality of Service (QoS) όπως και η εκμετάλλευση του πρωτοκόλλου IPv6. Ακόμη άξιο αναφοράς είναι πως τα εν λόγω συστήματα βρίσκονται ακόμα υπό ανάπτυξη και δεν αποτελούν οριστικοποιημένο πρότυπο.

### 1.2.2 Δικτυακός εξοπλισμός (Βασικές έννοιες δικτύων)

Στην συνέχεια κρίνεται σκόπιμη μια αναφορά στον βασικό εξοπλισμό δικτύωσης με στόχο την επεξήγηση της λειτουργίας του. Στην συνέχεια της εργασίας θα πραγματοποιηθεί εκτενέστερη αναφορά αλλά και αναφορά σε περισσότερες τεχνολογίες δικτύωσης.

#### **1.2.2.1 Ενσύρματα δίκτυα**

Τα βασικά τμήματα ενός ενσύρματου δικτύου είναι η κάρτα δικτύου, η οποία επιτρέπει την σύνδεση του τερματικού στο δίκτυο μέσω του φυσικού μέσου, τα switches τα οποία ανήκουν στο backbone του δικτύου, συνδέονται σε αυτά τα τερματικά και αναλαμβάνουν την μεταφορά δεδομένων από τον αποστολέα στον παραλήπτη και οι γέφυρες οι οποίοι ενώνουν υποδίκτυα σε ένα ενιαίο δίκτυο.

#### **1.2.2.2 Ασύρματα δίκτυα**

Τα βασικά τμήματα ενός ασύρματου δικτύου είναι η ασύρματη κάρτα δικτύου, η οποία επιτρέπει στα τερματικά την πρόσβαση στο δίκτυο, τα access points τα οποία αναλαμβάνουν την μεταφορά δεδομένων μεταξύ των συσχετισμένων τερματικών σταθμών, οι αναμεταδότες οι οποίοι αναλαμβάνουν την επέκταση της ακτίνας κάλυψης ενός ασύρματου δικτύου ενισχύοντας το σήμα ενός access point αλλά και οι ασύρματες γέφυρες οι οποίες αναλαμβάνουν την ένωση υποδικτύων WLAN σε ένα ενιαίο δίκτυο.

#### **1.2.2.3 Κινητά δίκτυα**

Τα βασικά τμήματα ενός δικτύου κινητών επικοινωνιών είναι το κινητό τερματικό, η συσκευή η οποία έχει την δυνατότητα να έχει πρόσβαση στο δίκτυο, ο σταθμός βάσης, με τον οποίο κάθε κινητό τερματικό συσχετίζεται και το κυρίως δίκτυο μέσω του οποίου

δρομολογούνται τα δεδομένα. Το κυρίως δίκτυο είναι αυτό το οποίο διαθέτει και διεπαφές για δρομολόγηση κλήσεων σε άλλους παρόχους όπως δίκτυα σταθερών και κινητών επικοινωνιών.

### **1.2.3 Μεταπομπές (handoffs)**

Δεδομένου ότι η πτυχιακή αναφέρεται στην διαλειτουργικότητα μεταξύ ετερογενών δικτύων ένας σημαντικός όρος ο οποίος πρέπει να αναφερθεί είναι η μεταπομπή. Ως μεταπομπή ορίζεται η διαδικασία αποσυσχέτισης και συσχέτισης ενός τερματικού σταθμού σε διαφορετικούς σταθμούς κάποιου δικτύου. Στην συνέχεια αναλύονται οι τύποι handoffs. Το κρίσιμο σημείο είναι πως η μεταφορά του τερματικού γίνεται αδιάλειπτα.

#### **1.2.3.1 Vertical HO**

Ως Vertical Handoff ορίζεται η μεταπομπή με αλλαγή τύπου συνδεσιμότητας. Αν ένας τερματικός σταθμός διαθέτει δύο τρόπους σύνδεσης στο δίκτυο, σε περίπτωση που το δίκτυο υποστηρίζει και τους δύο τρόπους και κριθεί πως το ο τρόπος σύνδεσης που δεν χρησιμοποιείται είναι η βέλτιστη επιλογή τότε το τερματικό αλλάζει τρόπο σύνδεσης.

#### **1.2.3.2 Horizontal HO**

Ως Horizontal Handoff ορίζεται η μεταπομπή με διατήρηση του τύπου συνδεσιμότητας. Αν ένας τερματικός σταθμός βρεθεί μεταξύ δύο σταθμών βάσης του δικτύου στο οποίο ανήκει και αυτός που δεν χρησιμοποιεί κριθεί βέλτιστη επιλογή τότε το τερματικό μεταφέρεται στον δεύτερο.

### 1.2.3.3 Hard HO

Ως Hard Handoff ορίζεται η μεταπομπή κατά την οποία η σύνδεση που χρησιμοποιείται διακόπτεται πριν την μεταφορά στην επόμενη σύνδεση. Ο τρόπος αυτός είναι γνωστός και ως Brake-before-make. Το σημαντικό σημείο σε αυτού του τύπου τις μεταπομπές είναι η ελαχιστοποίηση του χρόνου διάρκειας μεταξύ της αποσυσχέτισης και συσχέτισης έτσι ώστε να μην παρατηρηθεί μη διαθεσιμότητα της υπηρεσίας.

### 1.2.3.4 Soft HO

Ως Soft Handoff ορίζεται η μεταπομπή κατά την οποία η σύνδεση που χρησιμοποιείται και η σύνδεση που θα χρησιμοποιηθεί μετά την μεταπομπή χρησιμοποιούνται παράλληλα για ένα μικρό χρονικό διάστημα ώστε να επιτευχθεί ομαλή μεταπομπή. Οι διαφορές με τον προηγούμενο τρόπο είναι η αυξημένη χρήση πόρων του δικτύου και η ομαλότερη μετάβαση.

## 1.2.4 Κωδικοποιήσεις δεδομένων

Δεδομένου πως θα πραγματοποιηθεί εκτενής αναφορά σε 3G συστήματα, τα οποία στηρίζονται κατά κόρον στην ασύρματη μετάδοση δεδομένων κρίνεται σκόπιμη η αναφορά στις διαθέσιμες κωδικοποιήσεις σύμφωνα με τις οποίες επιτρέπεται ταυτόχρονη πρόσβαση πολλαπλών τερματικών στο ίδιο φυσικό μέσο (multiple access). Δεδομένου πως το φυσικό μέσο είναι κοινό αυτό επιτυγχάνεται μέσω της πολυπλεξίας δεδομένων (multiplexing). Κάποιες από τις βασικές μεθόδους πρόσβασης είναι οι ακόλουθες.

Frequency Division Multiple Access (FDMA): Κατά το FDMA πραγματοποιείται κατάτμηση μίας συχνότητας σε υποσυχνότητες κάθε μία από τις οποίες χρησιμοποιεί το κάθε τερματικό. Δεδομένου ότι το κάθε ένα χρησιμοποιεί διαφορετική συχνότητα δεν υπάρχουν παρεμβολές μεταξύ τους.

Time Division Multiple Access (TDMA): Κατά το TDMA πραγματοποιείται χρονική κατάτμηση του συνόλου των συχνοτήτων. Σύμφωνα με το προηγούμενο δημιουργούνται κάποιες χρονικές θυρίδες (time slots) τις οποίες μπορεί να εκμεταλλευτεί το κάθε τερματικό για την επικοινωνία του με το δίκτυο. Ακόμη πολλαπλά time slots μπορούν να ενωθούν ώστε να παρέχουν στο τερματικό μεγαλύτερη διάρκεια χρήσης του φυσικού μέσου σε περίπτωση που τα υπόλοιπα τερματικά δεν έχουν δεδομένα προς αποστολή.

Space Division Multiple Access (SDMA): Κατά το SDMA πραγματοποιείται χωρική κατάτμηση. Πιο συγκεκριμένα σε αντίθεση με τα προηγούμενα ο πομπός διατηρεί πληροφορίες σχετικά με την γεωγραφική θέση του δέκτη και με αυτό τον τρόπο η εκπομπή πραγματοποιείται μέσω «έξυπνων» κεραιών (smart antennas) έτσι ώστε να μην είναι μόνο κατευθυντική αλλά και γεωγραφικά προσδιορισμένη προς τον δέκτη.

Spread Spectrum Multiple Access (SSMA): Το SS έκανε αρχικά την εμφάνιση του σε επικοινωνίες στις οποίες η ασφάλεια και η αντοχή σε παρεμβολές ήταν βασικοί παράγοντες, όπως τα στρατιωτικά δίκτυα επικοινωνιών. Η χρήση του έγινε γνωστή κατά τον δεύτερο παγκόσμιο πόλεμο αλλά η διάθεσή του σε εμπορικά συστήματα πραγματοποιήθηκε αρκετά χρόνια μετά. Σύμφωνα με το SS γίνεται χρήση διαφορετικών φέρονων συχνοτήτων για την διάδοση του σήματος. Φυσικά το εύρος μετάδοσης είναι κατά πολύ μεγαλύτερο από αυτό που αντιστοιχεί στην πληροφορία που μεταφέρεται. Επίσης η δομή του συνόλου συχνοτήτων προσομοιάζει αυτή του θόρυβο διαύλου. Βασική υποκατηγορία του SSMA είναι το Code Division Multiple Access (CDMA) και ειδικότερα η τεχνολογία Wideband-CDMA (W-CDMA), το οποίο χρησιμοποιείται από τα 3G δίκτυα. Κατά το CDMA επιτρέπεται στα τερματικά να μοιράζονται την ίδια συχνότητα καθ' όλη την χρονική διάρκεια. Αυτό επιτυγχάνεται μέσω της χρήσης κωδικών (cipher) οι οποίοι είναι ορθογώνιου μεταξύ τους. Με αυτόν τον τρόπο κατά την αποδιαμόρφωση του σήματος από



το τερματικό σταθμό δεν υπάρχουν παρεμβολές. Μειονέκτημα αποτελεί το γεγονός πως ο αριθμός των ορθογώνιων κωδικών είναι πεπερασμένος και μικρός.

#### **1.2.4.1 Multiple In Multiple Out**

Ιδιαίτερη αναφορά κρίνεται σκόπιμο να πραγματοποιηθεί στην τεχνολογία Multiple In Multiple Out (MIMO). Η συγκεκριμένη τεχνολογία χρησιμοποιείται σε συνεργασία με την κωδικοποίηση Orthogonal Frequency Division Multiple Access για την επίτευξη βελτιστοποίησης στην χωρική κατάτμηση. Σύμφωνα με την προαναφερθείσα τεχνολογία χρησιμοποιούνται «έξυπνες κεραιές» και στις δύο άκρες (αποστολέας – παραλήπτης) ώστε να πραγματοποιείται επιλογή του βέλτιστου σήματος ή συνδυασμός των δύο για την βέλτιστη λήψη σε χώρους με κακή ποιότητα σήματος.

#### **1.2.5 Τύποι μετάδοσης**

Η μεταφορά δεδομένων σε ένα δίκτυο μπορεί να πραγματοποιηθεί με διαφορετικούς τρόπους. Αυτό καθορίζεται τόσο από τον τύπο του δικτύου όσο και από τον εξοπλισμό. Αρχικά η επικοινωνία μπορεί να χαρακτηρίζεται ως μονόδρομη (simplex) και αμφίδρομη (duplex). Στην μονόδρομη επικοινωνία υπάρχει ένας πομπός και ένας η περισσότεροι δέκτες και οι ρόλοι αυτοί είναι στατικοί. Αντίθετα στην αμφίδρομη κάθε τερματικό που ανήκει στο δίκτυο αποτελεί πομποδέκτη έτσι ώστε να μπορεί να λαμβάνει και να στέλνει δεδομένα. Υποκατηγορίες της duplex επικοινωνίας αποτελούν το semi duplex και το full duplex. Κατά την semi duplex επικοινωνία ένα τερματικό μπορεί ή να στέλνει ή να λαμβάνει δεδομένα ενώ κατά την full duplex και τα δύο πραγματοποιούνται παράλληλα. Ακόμη άλλος ένας διαχωρισμός έγκειται στους παραλήπτες. Έτσι χαρακτηρίζεται ως unicasting ή μετάδοση από τον αποστολέα απευθείας στον παραλήπτη, ως multicasting η

μετάδοση από τον αποστολέα σε συγκεκριμένους παραλήπτες και ως broadcasting η μετάδοση από τον αποστολέα σε όλα τα τερματικά που ανήκουν σε ένα δίκτυο.

### **1.2.6 Κυψέλες (Cells)**

Ως κυψέλη νοείται η ακτίνα κάλυψης που δημιουργείται από πολλά συστήματα ασύρματης δικτύωσης. Ο όρος cell χρησιμοποιείται καθώς βασική επιδίωξη είναι η προσομοίωση της ακτίνας κάλυψης στο συγκεκριμένο σχήμα. Η χρήση τους προσφέρει το πλεονέκτημα πως σε περίπτωση ύπαρξης πολλαπλών cells μπορεί να καλυφθεί μία περιοχή με ελαχιστοποίηση των κενών κάλυψης αλλά και την ελαχιστοποίηση επικαλύψεων σήματος. Φυσικά η ακτίνα ενός δικτύου δεν είναι στατική και το σχήμα είναι δυναμικά μεταβαλλόμενο.

Οι κυψέλες χωρίζονται σε κατηγορίες σύμφωνα με το μέγεθος τους. Pico Cell θεωρείται η ακτίνα κάλυψης ενός WLAN, Micro Cell η ακτίνα κάλυψης μίας κεραίας κινητών επικοινωνιών, Macro Cell η ακτίνα κάλυψης ενός δικτύου κινητών επικοινωνιών ή ενός δικτύου WiMax ενώ Mega Cell η ακτίνα κάλυψης ενός δορυφόρου.

### **1.2.7 Περιαγωγή (Roaming)**

Ο όρος περιαγωγή χρησιμοποιείται ως επί το πλείστον στις κινητές επικοινωνίες. Κατά την κατάσταση περιαγωγής ένα τερματικό συνδέεται στο δίκτυο μέσω συγκεκριμένης τεχνολογίας κάνοντας χρήση της υποδομής διαφορετικού παρόχου. Σύμφωνα με αυτό σε κάθε δίκτυο χρησιμοποιούνται πράκτορες (agents) για την διατήρηση των στοιχείων των τερματικών που συμμετέχουν σε αυτά. Οι agents χωρίζονται σε δύο τύπους. Στους home agents και στους foreign agents. Μεταξύ των δύο πραγματοποιείται ανταλλαγή δεδομένων με τα στοιχεία για την καταχώριση όποιου τερματικού επιθυμεί πρόσβαση σε κάποιο ξένο δίκτυο. Μέσω αυτών των στοιχείων πραγματοποιείται τόσο η δρομολόγηση των

δεδομένων που αφορούν το τερματικό, όσο και τις διαδικασίες που αφορούν το accounting τμήμα του παρόχου.

### **1.2.8 Quality of Service (QoS)**

Η ποιότητα υπηρεσίας αναφέρεται στο επίπεδο ποιότητας μιας παρεχόμενης υπηρεσίας σε ένα χρήστη. Ο κάθε χρήστης, αιτείται μιας υπηρεσίας, πχ δεδομένων. Η υπηρεσία αυτή μπορεί να του προσφερθεί σε πολλά επίπεδα ποιότητας, αναφορικά, επί παραδείγματι, με το ρυθμό δεδομένων στον οποίο παρέχεται. Αυτό δηλώνει ότι άλλη ευχαρίστηση θα λάβει ο χρήστης αν του προσφερθεί μια υπηρεσία σε ένα δεδομένο ρυθμό και άλλη θα λάβει αν του προσφερθεί στο δεκαπλάσιο ρυθμό. Άρα, η υπηρεσία μπορεί να παρέχεται σε πολλά επίπεδα ποιότητας (quality of service – QoS levels). Τα επίπεδα αυτά, στη γενική περίπτωση δεν είναι εγγυημένα, αλλά υπάρχουν διάφοροι μηχανισμοί που αποσκοπούν στη διασφάλισή τους και συνήθως αυτοί εφαρμόζονται.

### **1.3 Εφαρμογές**

Βασικό κομμάτι για ένα δίκτυο είναι οι εφαρμογές οι οποίες είναι διαθέσιμες. Πέραν της σύνδεσης δύο ή περισσότερων τερματικών, οι εφαρμογές είναι αυτές οι οποίες καθορίζουν το πώς μπορούν να αλληλεπιδρούν μεταξύ τους αλλά και τι θα προσφέρεται στον χρήστη. Αρχικά η βασική διαθέσιμη υπηρεσία ήταν η μεταφορά δεδομένων. Με την πάροδο του χρόνου παράλα αυτά το τελευταίο έχει αλλάξει άρδην. Ένα πλήθος εφαρμογών είναι διαθέσιμο. Κάθε από αυτές τις εφαρμογές έχουν κάποιες απαιτήσεις από το δίκτυο Στην συνέχεια θα αναφερθούν συνοπτικά κάποιες από τις πιο ευρέως διαδεδομένες.

### **1.3.1 Μεταφορά δεδομένων (Data)**

Όπως ήδη αναφέρθηκε η μεταφορά δεδομένων είναι από τις αρχικές και τις πιο γνωστές υπηρεσίες δικτύων. Κάθε τερματικό μπορεί να διακινεί δεδομένα τα οποία μπορούν να αφορούν διαφορετικές υπηρεσίες. Ακόμη συνήθως γίνεται χρήση και διαφορετικών πρωτοκόλλων. Μέσω του πρωτοκόλλου File Transfer Protocol (FTP) το οποίο είναι ένα Server – Client based πρωτόκολλο το κάθε τερματικό μπορεί να συνδεθεί με ή χωρίς αυθεντικοποίηση σε έναν εξυπηρετητή στον οποίο μπορεί σύμφωνα με τα δικαιώματα του λογαριασμού του να μεταφορτώσει ή να στείλει δεδομένα στον εξυπηρετητή, να διαγράψει αρχεία ή να αλλάξει την δομή. Μέσω του πρωτοκόλλου Hyper Text Transfer Protocol (HTTP) μπορεί να συνδεθεί σε κάποιον εξυπηρετητή και να μεταφορτώσει ιστοσελίδες και διαδικτυακό περιεχόμενο. Μέσω των πρωτοκόλλων Post Office Protocol (POP) και Simple Mail Transfer Protocol (SMTP) μπορεί να αποστείλει και να παραλάβει ηλεκτρονικό ταχυδρομείο. Τέλος γνωστές υπηρεσίες που βασίζονται σε Data είναι τα προγράμματα άμεσης επικοινωνίας (messengers) και η πρόσβαση σε βάσεις δεδομένων.

### **1.3.2 Voice Over IP (VoIP) / Audio**

Η επόμενη υπηρεσία στην οποία θα γίνει αναφορά είναι η μεταφορά ήχου και φωνής μέσω δικτύου. Το πρωτόκολλο Voice Over IP είναι ένα σχετικά καινούργιο πρωτόκολλο το οποίο άλλαξε τα δεδομένα στον χώρο των τηλεπικοινωνιών. Μέσω αυτού του πρωτοκόλλου δίνεται η δυνατότητα πραγματοποίησης κλήσεων μέσω ενός δικτύου οι οποίες μπορούν να τερματίζονται ακόμα και σε δίκτυα κινητής ή σταθερής τηλεφωνίας. Ο λόγος που έγινε γνωστό το συγκεκριμένο πρωτόκολλο είναι πως λόγω της τεχνολογίας που χρησιμοποιεί (Packet switching έναντι Circuit Switching) η οποία θα αναφερθεί στην συνέχεια, εξασφαλίζεται μικρότερος κόστος χωρίς αυτό απαραίτητα να έχει επιπτώσεις στην ποιότητα επικοινωνίας. Ακόμη άλλη μια ευρέως διαδεδομένη υπηρεσία είναι η παροχή ραδιοφώνου

μέσω digital audio broadcasting (DAB). Βασικό πλεονέκτημα της είναι η παράκαμψη του περιορισμού της ακτίνας εκπομπής για κάποιο σταθμό.

### **1.3.3 Video**

Στην κατηγορία του video υπάρχει μια εύρος υπηρεσιών προς παράθεση. Αρχικά η εξέλιξη της τηλεφωνίας είναι η αντικατάσταση του ήχου με video. Με αυτό τον τρόπο μπορεί να πραγματοποιηθεί μια video κλήση. Επίσης μέσω της υπηρεσίας του video conference προσφέρεται η δυνατότητα συμμετοχής πολλών χρηστών σε ένα ιδεατό χώρο συνάντησης. Η συγκεκριμένη υπηρεσία δίνει μια εναλλακτική στους περιορισμούς χώρου και απόστασης για την συμμετοχή σε κάποια εταιρική συνάντηση ή για ακαδημαϊκούς σκοπούς μέσω της παροχής του e-learning. Στην πρώτη περίπτωση βασικό πλεονέκτημα αποτελεί η μείωση του κόστους που προκύπτει από τις μετακινήσεις των στελεχών. Τέλος άλλη μια γνωστή υπηρεσία είναι η μεταφορά κινούμενης εικόνας (video) και η αναπαραγωγή του κατά την διάρκεια της μεταφοράς (streaming). Το video streaming χρησιμοποιείται σε πολλές περιπτώσεις. Μπορεί να χρησιμοποιηθεί από ένα Closed Captioned TV (CCTV) σύστημα για την σύγχρονη μεταφορά του video, για PayTV η ακόμα και για videoconference.

### **1.3.4 Απαιτήσεις εφαρμογών**

Κάθε εφαρμογή που εντάσσεται στις προηγούμενες υπηρεσίες έχει κάποιες απαιτήσεις. Αρχίζοντας από την μεταφορά δεδομένων αν και δεν υπάρχει κάποια απαίτηση σε ταχύτητα δικτύου είναι λογικό το γεγονός πως όσο μεγαλύτερη είναι η παρεχόμενη ταχύτητα τόσο γρηγορότερα ολοκληρώνονται οι μεταφορές. Το βασικό πρωτόκολλο μεταφοράς είναι το File Transfer Protocol (FTP). Συνήθως οι εφαρμογές που υποστηρίζουν voice δεν απαιτείται μεγάλο bandwidth λόγω της δειγματοληψίας και κατόπιν της

κωδικοποίησης που γίνεται παρόλα αυτά απαιτείται όσο το δυνατόν μικρότερη καθυστέρηση στο δίκτυο ώστε να μην παρατηρούνται διαλείψεις στην συνομιλία. Ένα από τα πιο γνωστά πρωτόκολλα είναι το VoIP. Τέλος στην μεταφορά video απαιτείται τόσο αρκετό bandwidth ώστε να μην χάνονται frames αλλά και μικρό delay.

Δεδομένο αποτελεί πως σε ένα δίκτυο δεν παρέχεται μόνο μία από της προαναφερθείσες υπηρεσίες αλλά συνδυασμοί τους ή και το σύνολο τους αναλόγως το δίκτυο. Για αυτό τον λόγο έχουν αναπτυχθεί πρωτόκολλα χαρακτηρισμού και διαχείρισης εφαρμογών. Κάνοντας χρήση τέτοιων πρωτοκόλλων τα δεδομένα που μεταφέρονται μπορούν να ακολουθήσουν κανόνες που εξασφαλίζουν την σωστή λειτουργία της εφαρμογής για την οποία αυτά προορίζονται. Ένα τέτοιο πρωτόκολλο είναι το Quality of Service (QoS). Στην συνέχεια της εργασίας θα γίνει εκτενέστερη αναφορά τόσο στην δομή του όσο και στον τρόπο λειτουργίας του.

Στον πίνακα που ακολουθεί εμφανίζονται οι κατηγορίες εφαρμογών και οι απαιτήσεις τους από το δίκτυο.

	Loss tolerance	Delay sensitive	Minimum bandwidth demand
Data (file transfer, email, file browsing)	No	No	No
Voice (VoIP)	Yes	Yes	Yes
Video (streaming)	Yes	Yes	Yes

Πίνακας 1.1: Συνοπτικός πίνακας απαιτήσεων εφαρμογών

### 1.3.5 Υφιστάμενοι περιορισμοί

Έχοντας εξετάσει τις απαιτήσεις των εφαρμογών σε θεωρητικό επίπεδο κρίνεται σκόπιμο να πραγματοποιηθεί αναφορά και τους περιορισμούς που προκύπτουν από το δίκτυο. Η χρήση ενός δικτύου παρουσιάζει αρκετές αποκλίσεις από την θεωρητική του φύση. Είτε το δίκτυο είναι βασισμένο σε ενσύρματο φυσικό μέσο είτε χρησιμοποιεί ως φυσικό μέσο τον αέρα πολλοί παράγοντες δημιουργούν περιορισμούς. Σε ένα ενσύρματο δίκτυο η εξασθένιση του σήματος, τα collisions πακέτων δεδομένων, τα φορτία των δρομολογητών και ο όγκος δεδομένων που διακινούνται μπορούν να προκαλέσουν καθυστερήσεις ή απώλειες πακέτων. Τα ασύρματα δίκτυα είναι ακόμα πιο ευπαθή λόγω παρεμβολών δεδομένου πως το μέσο είναι δυναμικά μεταβαλλόμενο.

Τα προηγούμενα αφορούν τα δίκτυα σαν αυτόνομα δίκτυα. Ακόμα περισσότεροι περιορισμοί προκύπτουν όταν γίνεται αναφορά σε ετερογενή δίκτυα, τα οποία αποτελούνται από συνδυασμούς των προαναφερθέντων δικτύων. Έτσι για παράδειγμα όταν ένα δίκτυο αποτελείται από ένα LAN και ένα WLAN και η εφαρμογή που χρησιμοποιείται από το τερματικό είναι streaming video, το γεγονός ότι λειτουργεί αδιάλειπτα όσο το τερματικό συνδέεται μέσω της ενσύρματης υποδομής δεν εξασφαλίζει το ίδιο σε περίπτωση μεταφοράς του στην ασύρματη υποδομή του.

### 1.4 Στόχος της πτυχιακής εργασίας

Σε αυτή την διπλωματική στόχο αποτελεί η εύρεση της κατάλληλης οντολογίας για την διαχείριση τέτοιων καταστάσεων σε ετερογενή δίκτυα. Αυτό σημαίνει η διασφάλιση του γεγονότος πως τα τερματικά που ανήκουν σε ένα δίκτυο θα μπορούν να χρησιμοποιούν όποιο συνδυασμό εφαρμογών επιθυμεί ο χρήστης χωρίς να παρατηρείται διαφοροποίηση

καθ' όλη την διάρκεια της παραμονής του τερματικού στο δίκτυο, ανεξαρτήτου της τεχνολογίας της υποδομής που χρησιμοποιεί.

Για να πραγματοποιηθεί κάτι τέτοιο κρίνεται σκόπιμη η εκτενής αναφορά στο θεωρητικό υπόβαθρο των διαθέσιμων τεχνολογιών δικτύωσης, των υπάρχουσών τεχνολογιών διασφάλισης ποιότητας όπως και των διαθέσιμων μηχανισμών διαχείρισης δικτύων. Μετά την κατανόηση των προηγούμενων θα πραγματοποιηθεί μια γενική αναφορά στην έννοια της οντολογίας και στην συνέχεια η ανάπτυξη και η αιτιολόγηση της προτεινόμενης οντολογίας. Αφετηρία αυτής θα είναι το πρότυπο P1900 της IEEE, το οποίο και θα αναλυθεί ενδελεχώς.



## 2. Συστήματα πέραν της τρίτης γενιάς (B3G systems)

Όπως ήδη αναφέρθηκε, με την πρόοδο της τεχνολογίας μεγάλη έμφαση δίνεται στην κινητικότητα του χρήστη ο οποίος συμμετέχει σε ένα δίκτυο. Στο ακόλουθο κεφάλαιο κρίνεται σκόπιμη η ανασκόπηση και η επεξήγηση του συνόλου της θεωρίας των βασικών τεχνολογιών δικτύωσης, αλλά και του συνόλου των διαθέσιμων σύγχρονων τύπων δικτύων. Κατόπιν αυτού και δεδομένου του γεγονότος πως τα B3G συστήματα αποτελούν ένα ενιαίο δίκτυο το οποίο αποτελείται από ετερογενή δίκτυα θεωρείται ευκολότερη η αναφορά στα χαρακτηριστικά τέτοιων δικτύων.

### 2.1 Τεχνολογίες δικτύων υποδομής

#### 2.1.1 Τύποι δικτύων (*Circuit Switching, Packet Switching, Virtual Circuit Switching*)

Ο βασικός διαχωρισμός οποιουδήποτε δικτύου έγκειται στον τρόπο που εγκαθιδρύονται οι συνδέσεις. Οι βασικοί τρόποι συνδέσεων είναι δύο. Το Circuit Switch και το Packet Switched. Με αυτό τον τρόπο τα δίκτυα χωρίζονται σε δύο βασικές κατηγορίες. Στα Circuit Switched (CS) δίκτυα και στα Packet Switched (PS) δίκτυα. Χρονικά τα CS δίκτυα δημιουργήθηκαν πρώτα και χρησιμοποιούνται ακόμα για την παροχή Plain Old Telephony System (POTS) τηλεφωνικών συνδέσεων και κινητών επικοινωνιών βασισμένων στην τεχνολογία Global System for Mobile communications (GSM). Η βασική θεωρία πίσω από

αυτού του τύπου τα δίκτυα είναι η εγκαθίδρυση φυσικών συνδέσεων σε όλο το μήκος του δικτύου μεταξύ των σταθμών που επιθυμούν να επικοινωνήσουν. Σε μία τέτοια περίπτωση η ύπαρξη μιας μόνιμης σύνδεσης εγγυάται την επικοινωνία χωρίς διαλείψεις καθώς το path των δεδομένων είναι συγκεκριμένο, γεγονός το οποίο επιτρέπει τον υπολογισμό του delay το οποίο είναι σταθερό.

Στην συνέχεια κρίνεται σκόπιμη η αναφορά σε βασικούς περιορισμούς που απορρέουν από την συγκεκριμένη τεχνολογία. Αρχικά η ύπαρξη δεσμευμένων κυκλωμάτων για κάποια χρονική διάρκεια σε συνδυασμό με την ύπαρξη ορισμένου αριθμού φυσικών κυκλωμάτων δημιουργεί τον περιορισμό των υποστηριζόμενων χρηστών. Όσο αυξάνεται ο αριθμός των χρηστών κάποιου δικτύου τεχνολογίας CS τόσο αυξάνεται η πιθανότητα μη εξυπηρέτησης των χρηστών δεδομένου πως ο αριθμός των κυκλωμάτων παραμένει σταθερός. Ακόμη πρέπει να ληφθεί σοβαρά υπ' όψιν πως η διατήρηση κάποιων κυκλωμάτων για συγκεκριμένο χρονικό διάστημα εμπεριέχει υψηλό κόστος. Ο τελευταίος ήταν και ένας από τους βασικότερους λόγους μεταστροφής στα δίκτυα τεχνολογίας PS.

Η λογική εξέλιξη των CS δικτύων είναι τα PS δίκτυα. Τα δίκτυα αυτά χωρίζουν τα δεδομένα προς μεταφορά σε αυτόνομα τμήματα τα οποία και ονομάζονται πακέτα και κατόπιν δρομολογούνται μέσω διαφορετικών διαδρομών από τον αποστολέα στον παραλήπτη. Κατά τον κατακερματισμό των δεδομένων σε πακέτα πραγματοποιείται αρίθμηση των τελευταίων έτσι ώστε να είναι δυνατή η επανασυναρμολόγηση τους από τον παραλήπτη. Δεδομένης της ύπαρξης δρομολογητών από τους οποίους περνάνε πακέτα από διάφορους αποστολείς προς διάφορους παραλήπτες διατίθενται τα χαρακτηριστικά τόσο της ύπαρξης ουράς αναμονής και της διατήρησης προσωρινής αποθήκευσης δεδομένων.

Ομοίως με το προκάτοχο του και η συγκεκριμένη τεχνολογία διαθέτει κάποια μειονεκτήματα. Βασικό πρόβλημα της τεχνολογίας είναι η δυναμική μεταβολή καθυστέρησης παράδοσης των πακέτων στον παραλήπτη. Ακόμα δεδομένου του κατακερματισμού δεν υπάρχει καμία εγγύηση πως θα φτάσουν όλα τα πακέτα στον τελικό παραλήπτη. Βασικό πλεονέκτημα είναι η εξυπηρέτηση μεγαλύτερου αριθμού τερματικών χωρίς την αύξηση των διαθέσιμων συνδέσεων. Ακόμα το γεγονός πως δεν απαιτείται το

δίκτυο ή μέρος αυτού να καταλαμβάνονται από ένα ζευγάρι αποστολέα και παραλήπτη για ορισμένο χρόνο κατά την επικοινωνία τους.

Για να δοθεί μια λύση στα βασικά μειονεκτήματα της τεχνολογίας αυτής υλοποιήθηκε το virtual circuit switching (VCS). Ως VC ορίζεται μια υπηρεσία επικοινωνίας βασισμένη στην εγκαθίδρυση συνδέσεων κατά την οποία τα δεδομένα παραδίδονται μέσω επικοινωνίας βασισμένης σε PS. Η τεχνολογία αυτή προσομοιάζει την CS δεδομένου πως τα πακέτα παραδίδονται με την σωστή σειρά και για αυτό υλοποιείται ή εγκαθίδρυση μιας σύνδεσης. Όπως και στα δίκτυα CS παρέχεται σταθερό bitrate και σταθερή καθυστέρηση παρόλα αυτά αυτό δεν γίνεται κατ' απόλυτο τρόπο.

#### **2.1.1.1 Αξιόπιστα (TCP) και μη αξιόπιστα (UDP) πρωτόκολλα επικοινωνίας**

Στα δίκτυα PS, τα οποία είναι και αυτά που χρησιμοποιούνται ευρέως στα σύγχρονα δίκτυα υψηλών ταχυτήτων υπάρχουν δύο διαφορετικές επιλογές πρωτοκόλλου μεταφοράς, για την σύνδεση και την επικοινωνία τερματικών εντός του δικτύου. Το πρώτο πρωτόκολλο μεταφοράς που χρησιμοποιείται είναι το Transmission Control Protocol (TCP) και το δεύτερο είναι το User Datagram Protocol (UDP). Οι βασικές διαφορές μεταξύ των δύο είναι πως το TCP διαθέτει πολλές λειτουργίες για τον έλεγχο παράδοσης, τον έλεγχο ροής και συμφόρησης, τη σωστή σειρά και την αποφυγή παραδόσεων πανομοιότυπων πακέτων, τα οποία μπορεί να προκύψουν για διάφορους λόγους μετά την αποστολή και πριν την παράδοση στην εφαρμογή του παραλήπτη. Ακόμη διαθέτει δυνατότητες εντοπισμού και διόρθωσης λαθών που προκύπτουν κατά την μετάδοση. Όλες αυτές οι λειτουργίες φυσικά επιβαρύνουν την ταχύτητα της επικοινωνίας καθώς εμπεριέχουν τόσο κόστος επεξεργασίας δεδομένων όσο και κόστος επιπρόσθετου φορτίου δικτύου καθώς ανταλλάσσεται επιπρόσθετος όγκος πληροφοριών μεταξύ του αποστολέα και του παραλήπτη.

Στον αντίποδα βρίσκεται το UDP σαν πρωτόκολλο επιπέδου μεταφοράς, το οποίο αν και δεν είναι ικανό να παρέχει την αξιόπιστη επικοινωνία μεταξύ τερματικών χρησιμοποιείται όταν κάτι τέτοιο δεν είναι απαραίτητο, ήτοι δεν υπάρχουν αυστηρές προδιαγραφές της

πραγματοποιούμενης επικοινωνίας, ως προς την ασφάλεια. Το βασικό πλεονέκτημα του είναι η ταχύτητα επικοινωνίας.

Τα παραπάνω πρωτόκολλα μεταφοράς χρησιμοποιούνται ευρέως από διάφορες εφαρμογές (πρωτόκολλα επιπέδου εφαρμογής). Έτσι, υπάρχουν εφαρμογές που εξ' ορισμού τους λειτουργούν με το ένα ή το άλλο πρωτόκολλο μεταφοράς.

### **2.1.2 Είδη δικτύων**

Η παρούσα παράγραφος έχει ως στόχο της να παραθέσει τα περισσότερο γνωστά είδη ενσύρματων και ασυρμάτων δικτύων.

Από ενσύρματα δίκτυα, μελετώνται τα LAN και WAN, ενώ έμφαση δίνεται περισσότερο στα ασύρματα δίκτυα, όπως WLAN, WiMAX, 3G networks, PAN, satellite networks, DVB/DAB and GAN.

#### **2.1.2.1 Local Area Network (LAN)**

Ιστορικά το LAN υπήρξε ο πρώτος ευρέως διαδεδομένος τρόπος δικτύωσης. Βασισμένο στον χαλκό επέτρεπε την επικοινωνία δύο ή περισσότερων τερματικών που συνήθως βρίσκονταν στον ίδιο χώρο. Κάθε δίκτυο LAN αποτελείται από κάποια βασικά συστατικά. Το πρώτο είναι το φυσικό μέσο. Αρχικά η δικτύωση πραγματοποιείτο μέσω ομοαξονικού καλωδίου. Σε αυτό συνδέονταν τα τερματικά μέσω αντίστοιχων καρτών δικτύωσης. Η εξέλιξη στον τομέα αυτό ήταν η χρήση καλωδίων Unshielded Twisted Pairs (UTP). Η χρήση τους εφαρμόστηκε από το StarLAN το 1984. Η μέγιστη ταχύτητα μεταφοράς ανερχόταν στο 1Mbps. Η χρήση των UTP καλωδίων εδραιώθηκε γρήγορα και με επιδίωξη μεγαλύτερων ταχυτήτων μεταφοράς δημιουργήθηκε το πρότυπο 10Base-T και η έννοια της δομημένης καλωδίωσης. Το καλώδιο 10Base-T αποτελείται από 4 αθωράκιστα

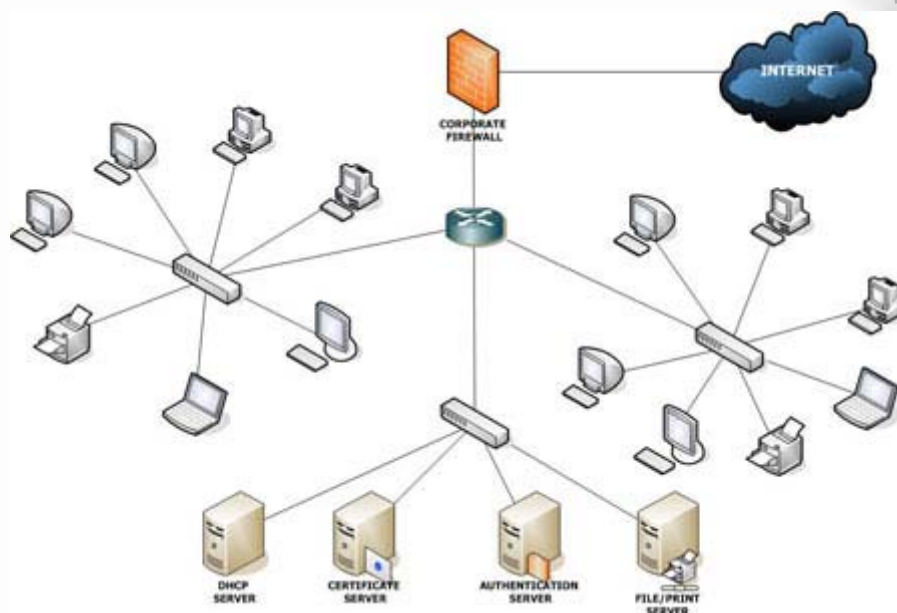
συνεστραμμένα ζεύγη χάλκινων καλωδίων. Η μέγιστη ταχύτητα μεταφοράς αυξήθηκε στα 10Mbps. Με συνεχόμενες εξελίξεις εμφανίστηκαν οι διάδοχοι του προτύπου πάντα με κύριο γνώμονα την αύξηση της ταχύτητας. Έτσι ακολούθησε το 100Base-T με μέγιστη ταχύτητα μεταφοράς τα 100Mbps και το 1000Base-T με μέγιστη ταχύτητα μεταφοράς το 1Gbps. Δύο σημαντικά δεδομένα κρίνεται σκόπιμο να αναφερθούν επιπρόσθετα των προαναφερθέντων. Το πρώτο είναι πως όλα τα μεταγενέστερα πρότυπα διέθεταν backwards compatibility σε θέματα ταχυτήτων. Αυτό είναι ένα πολύ σημαντικό γεγονός καθώς για την επέκταση ήδη υπάρχοντων δικτύων δεν αποτέλεσε προϋπόθεση η αλλαγή της καλωδίωσης. Ακόμη δυνατή ήταν η παράλληλη λειτουργία τερματικών σε διαφορετικές ταχύτητες σύνδεσης στο ίδιο δίκτυο. Το δεύτερο είναι ο βασικός περιορισμός που σύμφωνα με τις προδιαγραφές των τριών προτύπων η μέγιστη απόσταση λειτουργίας τους είναι τα 100m.

Το δεύτερο συστατικό σε ένα LAN μετά την εδραίωση της δομημένης καλωδίωσης είναι το backbone του δικτύου. Η πρώτη υλοποίηση για αυτό ήταν η χρήση HUB. Το HUB ήταν η συσκευή αυτή στην οποία συνδέονταν όλα τα τερματικά και επέτρεπε την μεταφορά μηνυμάτων. Ακόμη εκτός από τον ρόλο της μεταφοράς είχαν και μέρος του ρόλου της ανίχνευσης collision δεδομένου πως προωθούσαν ένα μήνυμα εμπλοκής σε όλους τους τερματικούς σταθμούς σε περίπτωση ανίχνευσης για να προβούν σε διακοπή αποστολών μηνυμάτων και επαναμετάδοση σε δεύτερο χρόνο. Διάδοχος του HUB υπήρξε το Switch το οποίο χρησιμοποιείται μέχρι και σήμερα. Ο διαχωρισμός των δύο έγκειται στο γεγονός πως τα switches είναι ικανά να δρομολογήσουν τα δεδομένα απευθείας από τον αποστολέα προς τον παραλήπτη μέσω unicast μεταδόσεων. Τα HUBs απλά πραγματοποιούσαν μεταδόσεις multicasting και ήταν λειτουργία του τερματικού να αναγνωρίσει αν τα δεδομένα αφορούν το συγκεκριμένο ή κάποιο άλλο τερματικό στο ίδιο δίκτυο. Τα σύγχρονα switches χωρίζονται σε δύο κατηγορίες. Στα managed και στα unmanaged. Τα πρώτα προσφέρουν επιπρόσθετες λειτουργίες όπως επιλογή ενεργοποίησης / απενεργοποίησης ενός εύρους πορτών, αποκλεισμός πρόσβασης βασισμένος στην MAC address του τερματικού, υλοποίηση πρωτοκόλλου Spanning tree, υποστήριξη SMTP, port monitoring, σύνδεση πορτών και δημιουργία VLAN. Από τα προηγούμενα κρίνεται σκόπιμη η επεξήγηση του port monitoring λειτουργία σύμφωνα με την οποία το switch έχει την

δυνατότητα να προωθεί την κίνηση από συγκεκριμένες πόρτες σε μια πόρτα που ορίζει ο διαχειριστής, της σύνδεσης πορτών κατά την οποία κάποιο τερματικό μπορεί να συνδεθεί σε δύο πόρτες του switch παράλληλα μέσω δύο καρτών δικτύου και το switch να τις θεωρήσει ως μία πόρτα με αυξημένο bandwidth και η δημιουργία Virtual LAN (VLAN) λειτουργία η οποία επιτρέπει την δημιουργία αυτόνομων υποδικτύων στα οποία εντάσσονται τα τερματικά που είναι συνδεδεμένα στο ίδιο switch. Δεδομένο αποτελεί πως το κόστος των managed είναι πολύ πιο υψηλό από τα unmanaged. Για αυτό το λόγο τα πρώτα χρησιμοποιούνται σε μεγάλα μεγέθους δίκτυα στα οποία οι παραπάνω λειτουργίες μπορούν να αξιοποιηθούν.

Άλλο ένα στοιχείο ενός LAN είναι οι γέφυρες, εξοπλισμός ο οποίος αναλαμβάνει να ενοποιεί υποδίκτυα έτσι ώστε να συμπεριφέρονται σαν ένα ενιαίο δίκτυο. Οι γέφυρες όπως και οι δρομολογητές βασίζονται στις ίδιες αρχές λειτουργίας του switch απλά δημιουργούν την σύνδεση στο επίπεδο σύζευξης δεδομένων του μοντέλου OSI. Επιπρόσθετα οι δρομολογητές δημιουργούν συνδέσεις στο επίπεδο δικτύου του μοντέλου OSI. Απόρροια του συγκεκριμένου είναι πως οι γέφυρες αναγνωρίζουν τα τερματικά μέσω MAC address ενώ οι δρομολογητές μέσω IP address. Λόγω αυτού του γεγονότος οι γέφυρες δεν μπορούν να προβούν σε διαχωρισμό δικτύων.

Η επιθυμία παροχής αυξημένων ταχυτήτων δεν έχει εξαλειφτεί ακόμα και σήμερα. Για αυτό το λόγω παρατηρείται μια στροφή από τον χαλκό ως φυσικό μέσο μετάδοσης σε άλλα μέσα. Αυτό που επικράτησε στην αλλαγή είναι η οπτική ίνα. Μέχρι πρότινος η χρήση οπτικών ινών ήταν πολύ περιορισμένη και το κόστος τους αυξημένο. Παρ' όλα αυτά τα βασικά χαρακτηριστικά των οπτικών ινών σε σύγκριση με τον χαλκό, δηλαδή το γεγονός πως δεν επηρεάζονται από ηλεκτρομαγνητικές παρεμβολές, παρατηρείται μικρότερη εξασθένιση σήματος σε συνδυασμό με την παροχή μεγαλύτερων ταχυτήτων μεταφοράς τις ανέδειξαν ως πλέον κατάλληλες για διαδοχή. Ήδη σε πολλές χώρες αναπτύσσονται με ταχείς ρυθμούς δίκτυα οπτικών ινών για λογαριασμό τηλεπικοινωνιακών παροχών. Όπως με κάθε αλλαγή στην τεχνολογία και βάσει των οικονομικών κλίμακας σε κάποια χρόνια τα δίκτυα οπτικών ινών προβλέπεται να είναι εύκολα προσβάσιμα από ιδιώτες και από μικρές επιχειρήσεις.



Εικόνα 2.1: Τυπική δομή ενός δικτύου LAN

### 2.1.2.2 Wide Area Network (WAN)

Τα δίκτυα WAN είναι η λογική επέκταση των δικτύων LAN ώστε να παρέχεται συνδεσιμότητα σε μεγαλύτερες αποστάσεις. Ουσιαστικά τα WAN εξασφαλίζουν την ένωση δικτύων διαφόρων τεχνολογιών σε ένα ενιαίο δίκτυο. Χαρακτηριστικό παράδειγμα του μεγαλύτερου διαθέσιμου WAN είναι το Internet. Οι συνηθέστεροι τρόποι δημιουργίας ενός WAN είναι μέσω της χρήσης μισθωμένων γραμμών, στις οποίες κάποιες φορές ανασταλτικός παράγοντας επιλογής τους αποτελεί η υψηλή τιμή διάθεσης τους, μέσω circuit switched δικτύων κάνοντας χρήση dialup συνδέσεων, μέσω PVC δικτύων και μέσω cell relay.

Ένας ιδιαίτερος τύπος WAN είναι τα ιδιωτικά. Αυτά συνήθως δημιουργούνται και διατηρούνται για να καλύψουν ανάγκες εταιριών ή οργανισμών με διάσπαρτες γεωγραφικά εγκαταστάσεις. Η δημιουργία ενός τέτοιου δικτύου μπορεί να επιτευχθεί με αρκετούς τρόπους. Δεδομένης της ιδιωτικής φύσης των προαναφερθέντων δικτύων, βασικό παράγοντα παίζει η διασφάλιση της ασφάλειας κατά την επικοινωνία των τερματικών. Δεδομένου του κόστους των μισθωμένων γραμμών η εναλλακτική που

επικρατεί για δημιουργία WAN είναι η χρήση του Virtual Private Network (VPN). Το VPN παρέχει αρκετές επιλογές όπως η αυθεντικοποίηση του τερματικού πριν την σύνδεση, η κρυπτογράφηση των δεδομένων ή ο καθορισμός μίας σταθερής διαδρομής κυκλωμάτων και δρομολογητών από το απομακρυσμένο τερματικό μέχρι το τοπικό δίκτυο, έχει παραλλαγές και διαφορετικούς τρόπους εφαρμογής.

Υπάρχουν διάφοροι τύποι VPN. Ένας πρώτος διαχωρισμός προκύπτει από τον πάροχο. Ένα VPN μπορεί να παρέχεται από την υποδομή της ίδιας της εταιρίας μέσω της επένδυσης της σε συγκεκριμένο εξοπλισμό για την λειτουργία του ή από κάποια τρίτη εταιρία. Στις δύο αυτές περιπτώσεις το τερματικό συνδέεται απευθείας στην εταιρία ή στον πάροχο ο οποίος αναλαμβάνει την υλοποίηση του VPN από τις εγκαταστάσεις του έως τις εγκαταστάσεις της εταιρίας. Ένας τύπος που χρησιμοποιήθηκε κατά κόρον στις δύο προηγούμενες περιπτώσεις ήταν το Virtual Private Dial-up Network. Σύμφωνα με αυτό το τερματικό συνδεόταν μέσω του τηλεφωνικού δικτύου (CS δικτύου) στον πάροχο και αυτός με την σειρά του στο δίκτυο της εταιρίας. Με την πάροδο του χρόνου και την εξέλιξη της τεχνολογίας διάφορα πρωτόκολλα εμφανίστηκαν για την δυνατότητα εκμετάλλευσης των PS δικτύων από το VPN. Η τεχνολογία που επικράτησε είναι το VPN tunneling σύμφωνα με το οποίο εγκαθιδρύεται μια συγκεκριμένη διαδρομή δεδομένων μέσα από προεπιλεγμένους δρομολογητές. Όπως στην προηγούμενη γενιά η σύνδεση του τερματικού μπορεί να πραγματοποιείται απευθείας στο δίκτυο της εταιρίας ή στην υποδομή τρίτου παρόχου. Τα επικρατέστερα πρωτόκολλα για αυτού του τύπου των VPNs είναι το Point to Point Tunneling Protocol (PPTP) το οποίο αναπτύχθηκε από κοινού από την Microsoft και την US Robotics και των Layer 2 Forwarding (L2F) και Layer 2 Tunneling Protocol (L2TP) από την Cisco. Εκτός από την δρομολόγηση φυσικά υποστηρίζεται και η κρυπτογράφηση των διακινούμενων δεδομένων.

### **2.1.2.3 Wireless Local Area Network (WLAN)**



Τα δίκτυα WLAN εμφανίστηκαν για να καλύψουν μια μεγάλη ανάγκη που εμφανίστηκε με την πρόοδο της τεχνολογίας. Αυτή η ανάγκη ήταν η παροχή δυνατότητας δικτύωσης σε τερματικά με παράλληλη διατήρηση της κινητικότητας τους. Με αυτή την τεχνολογία κινητά τερματικά όπως laptops, Personal Digital Assistants (PDA) και ακόμα και κινητά τηλέφωνα απέκτησαν την δυνατότητα πρόσβασης σε δίκτυα χρησιμοποιώντας ως φυσικό μέσο τον αέρα. Σε επίπεδο υλικού είναι απαραίτητη η ύπαρξη μιας ασύρματης κάρτας δικτύου ώστε να προστεθεί η δυνατότητα ασύρματης δικτύωσης σε κάποιο τερματικό. Αν και υπάρχουν τέτοιες κάρτες με διάφορους τρόπους σύνδεσης (USB, PCMCIA, PCI) η πλειοψηφία από τα σύγχρονα τερματικά όπως laptops και personal digital assistants (PDA) διαθέτουν ενσωματωμένες δυνατότητες ασύρματης δικτύωσης. Η συγκεκριμένη τεχνολογία έγινε ευρέως αποδεκτή και αναγνωρίσιμη σε τέτοιο σημείο ώστε η δυνατότητα να ενσωματωθεί και σε κάποιες κατηγορίες κινητών τηλεφώνων. Τα κραταιά πρωτόκολλα για την ασύρματη δικτύωση είναι τα 802.11a και 802.11b/g της IEEE. Η διαφορά των πρωτοκόλλων είναι οι συχνότητες λειτουργίας και οι υποστηριζόμενες ταχύτητες. Διάδοχος τους δε είναι το 802.11n στο οποίο, αν και βρίσκεται ακόμα σε draft έκδοση η οποία αναμένεται να οριστικοποιηθεί τον Δεκέμβριο του 2009, έχουν πραγματοποιηθεί οι ακόλουθες τροποποιήσεις. Αρχικά ενισχύει τόσο την μέγιστη ταχύτητα φτάνοντας στα 600 Mbps (προς το παρόν υποστηρίζονται μέχρι 270Mbps). Ακόμη εισάγεται η χρήση της τεχνολογίας Multiple Input Multiple Output (MIMO). Τέλος ενισχύει την ασφάλεια με την αύξηση των διαθέσιμων πρωτοκόλλων ασφαλείας. Στον ακόλουθο πίνακα παρουσιάζονται τα βασικά χαρακτηριστικά των πρωτοκόλλων. Το 802.11n δεν περιλαμβάνεται καθώς δεν έχει ακόμα οριστικοποιηθεί.

Standard	Frequency Band	Channels Supported	Channel width (MHz)	Mbps									
				11	5.5	2	1	-	-	-	-	-	-
802.11b	2.4 – 2.485 GHz	3	20	11	5.5	2	1	-	-	-	-	-	-
802.11g	2.4 – 2.485 GHz	3	20	54	48	36	24	18	12	9	6	2	1
802.11a	5.15 – 5.25 GHz, 5.25 – 5.35 GHz & 5.725 – 5.825 GHz	12	20	54	48	36	24	18	12	9	6	-	-

Πίνακας 2.1: Βασικά χαρακτηριστικά πρωτοκόλλων WLAN

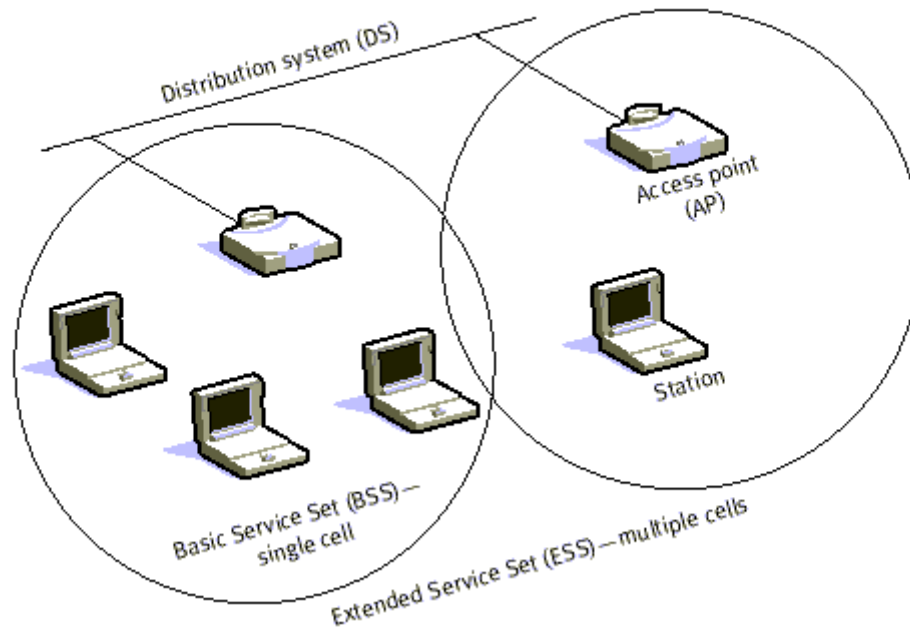
Ένα ασύρματο δίκτυο μπορεί να έχει δύο μορφές. Η πρώτη είναι η ad hoc λειτουργία κατά την οποία ένα δίκτυο δημιουργείται από τα τερματικά που βρίσκονται εντός ακτίνας. Σε αυτή την περίπτωση δεν χρειάζεται κάποιος επιπρόσθετος εξοπλισμός υποδομής. Συνήθως είναι δίκτυα που δημιουργούνται σε κάποιο χώρο για περιορισμένο χρόνο δεδομένου πως τα τερματικά δεν παραμένουν σε αυτό το χώρο μόνιμα. Η έλλειψη εξοπλισμού με ρόλο συντονιστή του δικτύου καλύπτεται από τα ίδια τα τερματικά.

Η δεύτερη λειτουργία είναι η λειτουργία υποδομής (infrastructure) και προϋποθέτει την ύπαρξη ενός επιπρόσθετου συστατικού σε επίπεδο υλικού, του Access Point (AP). Το AP έχει διπλή λειτουργία. Αρχικά παρέχει ένα κεντροποιημένο έλεγχο και συντονισμό του ασύρματου δικτύου καθώς όλα τα τερματικά συσχετίζονται με αυτό ώστε να επιτευχθεί η επικοινωνία μεταξύ τους ενώ παράλληλα παρέχει την δυνατότητα ενοποίησης του ασύρματου δικτύου με κάποιο ενσύρματο δίκτυο. Όπως είναι εμφανές η χρήση ενός AP δίνει την δυνατότητα διατήρησης ενός ασύρματου δικτύου σε ένα συγκεκριμένο χώρο χωρίς την απαραίτητη ύπαρξη κάποιου τερματικού.

Βασικό μειονέκτημα της τεχνολογίας WLAN αποτελεί η περιορισμένη ακτίνα κάλυψης η οποία είναι και δυναμικά μεταβαλλόμενη καθώς μέσω της χρήσης του αέρα ως φυσικό μέσο είναι ευπαθής σε ηλεκτρομαγνητικές παρεμβολές. Σε περίπτωση ανάγκης παροχής ασύρματης δικτύωσης σε μεγαλύτερη ακτίνα υπάρχουν δύο επιλογές. Η πρώτη είναι η λειτουργία δύο η περισσότερων APs σε συνδυασμό με ένα δίκτυο δομημένης καλωδίωσης. Δεδομένου πως όπως ήδη αναφέρθηκε ένα AP έχει την δυνατότητα ένωσης του με κάποιο ενσύρματο δίκτυο, δυο η περισσότερα AP μπορούν να δημιουργήσουν ένα πολυεπίπεδο ασύρματο δίκτυο. Σε αυτή την περίπτωση το κάθε AP μαζί με τους συσχετισμένους τερματικούς σταθμούς του ονομάζεται Basic Service Set (BSS) ενώ η ακτίνα κάλυψης του συγκεκριμένου ονομάζεται Basic Service Area (BSA). Το σύνολο των APs μαζί με το σύνολο των τερματικών σταθμών ονομάζεται Extended Service Set (ESS) και η υποδομή που παρέχει την ένωση των APs ονομάζεται Distribution System (DS).

Η ύπαρξη ενσύρματου δικτύου για την επίτευξη ένωσης δύο η περισσότερων APs δεν είναι απαραίτητη καθώς υπάρχει η εναλλακτική κατά την οποία γίνεται εκμετάλλευση του ίδιου φυσικού μέσου λειτουργίας του. Κάτι τέτοιο πραγματοποιείται με την χρήση ασύρματων γεφυρών (wireless bridges). Οι ασύρματες γέφυρες είναι ικανές να αναγνωρίζονται από τα APs και μέσω της χρήσης του Wireless Distribution System (WDS) να ενώνουν τα BSS μετατρέποντας τα σε ένα ESS.

Τέλος, αν το ενδιαφέρον σε ένα WLAN δεν είναι η δημιουργία ενός ESS αλλά απλά η αύξηση της ακτίνας κάλυψης ενός BSS μπορούν να χρησιμοποιηθούν ασύρματοι αναμεταδότες (repeaters). Αν και ένας repeater αποτελεί αυτόνομη συσκευή δεν αναγνωρίζεται από το AP. Η μόνη λειτουργία που εκτελεί ένας repeater είναι η αναμετάδοση των κυμάτων RF αφού πρώτα τα ενισχύσει. Δεν είναι σε θέση να αναγνωρίσει τα δεδομένα τα οποία διακινούνται στο δίκτυο.



Εικόνα 2.2: Τυπική δομή ESS

#### 2.1.2.4 WiMax

Τα δίκτυα Worldwide Interoperability for Microwave Access (ευρέως γνωστά ως WiMax) είναι δίκτυα τα οποία βασίζονται στην αρχή της ασύρματης δικτύωσης αλλά εμφανίστηκαν για να καλύψουν ένα από τα μεγαλύτερα μειονεκτήματα του WLAN. Αυτό δεν είναι άλλο από την ακτίνα κάλυψης. Τα WiMax δίκτυα βρίσκονται σε συνεχή εξέλιξη ως νέα τεχνολογία και δεδομένης της ύπαρξης πολλών προτύπων μπορούν να χρησιμοποιηθούν σε διαφορετικές περιπτώσεις. Μέχρι σήμερα υπάρχουν δύο συστήματα, τα WiMax και τα mobile WiMax τα οποία βασίζονται στο πρότυπο 802.16-2005 της IEEE. Τα εν λόγω δίκτυα όπως θα παρατεθεί και στην συνέχεια είναι εφάμιλλα των WLAN στον τρόπο λειτουργίας και των δικτύων κινητής σε θέματα υποδομής. Σημαντικά δεδομένα για το φυσικό μέσο αποτελούν πως το χρησιμοποιούμενο φάσμα συχνοτήτων μπορεί να είναι κατοχυρωμένο ανά πάροχο για την αποφυγή παρεμβολών ή όχι και πως δεν υπάρχει η απαίτηση οπτικής

επαφής του τερματικού με την υποδομή του δικτύου λόγω του μήκους κύματος των συχνοτήτων που χρησιμοποιούνται.

Βασικό συστατικό είναι η το κινητό τερματικό ή η κάρτα δικτύου για σταθερά τερματικά. Σε WiMax δίκτυα μπορούν να συμμετέχουν τόσο σταθερά τερματικά χωρίς την ανάγκη ύπαρξης καλωδίωσης, όπου σε αυτή την περίπτωση απαιτείται η ύπαρξη μιας ασύρματης κάρτας δικτύου για να τους επιτραπεί η δυνατότητα πρόσβασης στο δίκτυο. Κάποια κινητά τερματικά διαθέτουν ενσωματωμένη κάρτα δικτύου για WiMax δίκτυα αλλά η τεχνολογία κάνει την εμφάνιση της ακόμα και σε κινητά τηλέφωνα. Βασικό χαρακτηριστικό των καρτών ασύρματου δικτύου WiMax αποτελεί η δυνατότητα της χρήσης της τεχνολογίας MIMO. Με αυτόν τον τρόπο γίνεται ανεκτή μια καθυστέρηση λήψης της τάξεως των 10 msec. Αυτό αποτελεί πλεονέκτημα με σκοπό την βελτιστοποίηση της παρεχόμενης υπηρεσίας δεδομένης της έκτασης της ακτίνας που καλύπτει ένα τέτοιο δίκτυο.

Στην συνέχεια δεύτερο συστατικό αποτελεί η υποδομή δικτύου. Η διττή λειτουργία της δίνει την δυνατότητα εγκατάστασης κεραιών εσωτερικού αλλά και εξωτερικού χώρου. Τα base stations εξωτερικών χώρων μπορούν να καλύψουν μια ακτίνα από 500 μέτρα μέχρι 2 χιλιόμετρα σύμφωνα με τις γεωγραφικές ιδιαιτερότητες της περιοχής που καλύπτουν. Ακόμα υπάρχει η δυνατότητα της απευθείας επικοινωνίας ενός BS με κάποιο γειτονικό του σε διαφορετικές συχνότητες και μέσω μικροκυμματικής ζεύξης, αρκεί να υπάρχει οπτική επαφή των δύο (line of sight). Μέσω αυτής της δυνατότητας, η οποία αναφέρεται ως point to multipoint backhaul, μπορεί να δημιουργηθεί εύκολα ένα σύστημα κεραιών το οποίο καλύπτει μια γεωγραφική περιοχή ανάλογη με αυτή μιας πόλης. Για την mobile έκδοση του πρωτοκόλλου στην υποδομή περιλαμβάνονται και MSCs για την δυνατότητα διαχείρισης και δρομολόγησης των κλήσεων από τα κινητά τερματικά.

Σημαντικό γεγονός για το εν λόγω πρωτόκολλο αποτελεί η επιλογή της Intel, εταιρίας παραγωγής επεξεργαστών που κατέχει το μεγαλύτερο μερίδιο της αγοράς και πιο συγκεκριμένα το 77% στους προσωπικούς υπολογιστές, να παρέχει εγγενή υποστήριξη της τεχνολογίας από τους επεξεργαστές Centrino 2 οι οποίοι χρησιμοποιούνται σε φορητούς υπολογιστές.

### 2.1.2.5 Δίκτυα 3G

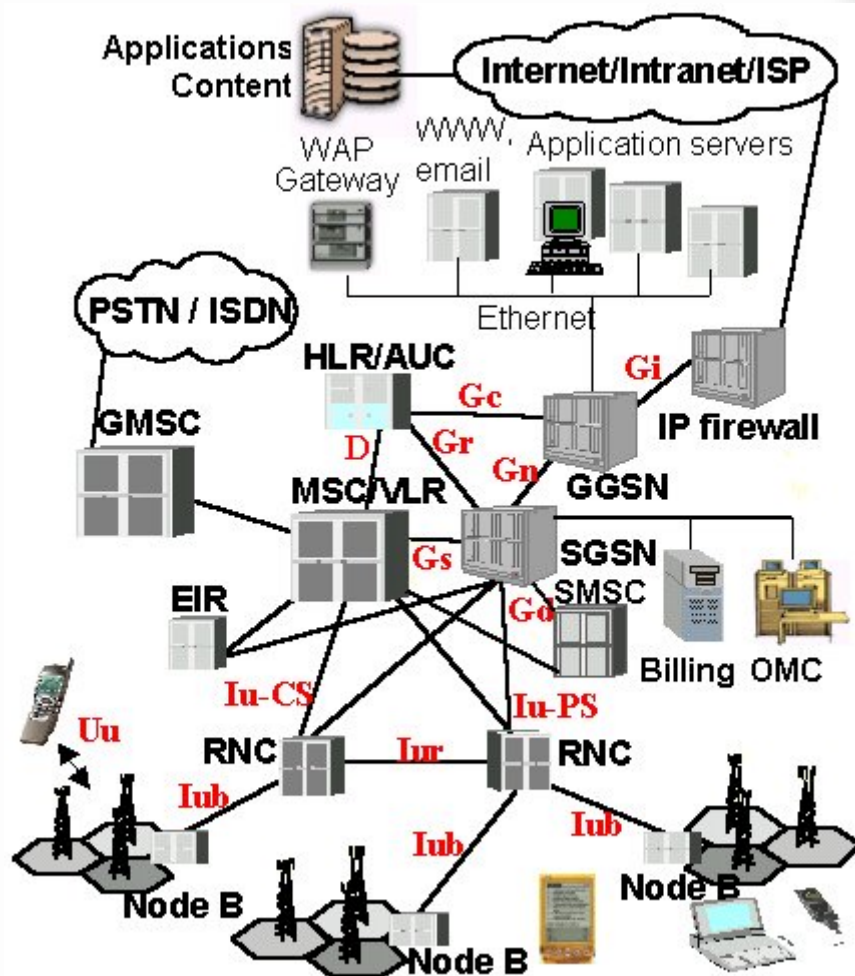
Τα δίκτυα τρίτης γενιάς (3G) έχουν ως κυριότερο εκπρόσωπό τους το Universal Mobile Telecommunications System (UMTS) αποτελούν φυσική εξέλιξη των δικτύων δεύτερης γενιάς ή Global System for Mobile communications (GSM) κινητών επικοινωνιών. Τα δίκτυα κινητών επικοινωνιών μέχρι την δεύτερη γενιά επικεντρώνονταν στις υπηρεσίες παροχής φωνής μέσω ψηφιακών καναλιών. Μια πρώτη αλλαγή στο συγκεκριμένο γεγονός πραγματοποιήθηκε με την ανάπτυξη των δικτύων 2.5 γενιάς τα οποία στην ήδη υπάρχουσα τεχνολογία πρόσθεσαν μέσω του General Packet Radio Service (GPRS) στην Ευρώπη και του Enhanced Data rates for GSM Evolution (EDGE) στις Ηνωμένες πολιτείες της Αμερικής την δυνατότητα μεταφοράς δεδομένων. Τα δίκτυα 3G αντιθέτως παρέχουν εγγενή υποστήριξη μεταφοράς δεδομένων μέσω του HSPA. Ακόμη μια διαφοροποίηση είναι η χρήση διαφορετικών συχνοτήτων για την πρόσβαση στο δίκτυο. Τα GSM δίκτυα χρησιμοποιούσαν τα 900 και τα 1800 MHz ενώ τα 3G δίκτυα τα 1900 MHz.

Βασικό συστατικό για ένα δίκτυο 3G είναι το κινητό τερματικό. Το κινητό τερματικό το οποίο υποστηρίζει την εν λόγω τεχνολογία είναι ικανό να συνδεθεί σε κάποιο δίκτυο τρίτης γενιάς παρόλα αυτά όλα τα τερματικά είναι backwards compatible με τα δίκτυα GSM. Αυτό συμβαίνει καθώς ένα δίκτυο 3G μπορεί να λειτουργήσει παράλληλα και σε συνεργασία με ένα δίκτυο GSM. Ως κινητό τερματικό μπορεί βέβαια να οριστεί εκτός από το τηλέφωνο που χρησιμοποιείται κατά κόρων σε ένα τέτοιο δίκτυο και το οποιοδήποτε τερματικό διαθέτει τον κατάλληλο εξοπλισμό ώστε να έχει πρόσβαση στο δίκτυο, όπως USB modem. Ακόμη άξιο αναφοράς αποτελεί το γεγονός πως η πλειοψηφία των τερματικών δύναται να αλλάζουν λειτουργία σύμφωνα με το διαθέσιμο δίκτυο μέσω μεταπομπών και χωρίς να χρειαστεί κάποια περεταίρω αλλαγή.

Το δεύτερο τμήμα του είναι ο RNC (Ελεγκτής Ασύρματων Δικτύων) ο οποίος ελέγχει τους συνδεδεμένους σε αυτόν BSs. Στις κύριες αρμοδιότητές του υπάγονται η διαχείριση των πόρων, τμήματα διαχείρισης της κίνησης, ο σχεδιασμός των πακέτων και ο έλεγχος

παράδοσης αυτών, οι λειτουργίες ασφάλειας, ο εξωτερικός έλεγχος της ισχύος των βρόχων και άλλα.

Τέλος, Το Node B υπάγεται στο UTRAN τμήμα του δικτύου UMTS και αντιστοιχεί στο σταθμό πομποδέκτη βάσης (BTS) του GSM. Είναι υπεύθυνος για την κρυπτογράφηση στη ραδιοζεύξη με τον κινητό σταθμό ενώ παράλληλα παρέχει τις απαραίτητες λειτουργίες για τη διαδικασία της σηματοδosis που εφαρμόζεται για την υποστήριξη της ραδιο-διεπαφής του WCDMA. Κάθε Node B ελέγχεται από έναν RNC που ονομάζεται CRNC (Controlling RNC), ο οποίος με τη σειρά του ελέγχει το φόρτο και τη συμφόρηση της κυψέλης, ενώ παράλληλα εκτελεί τον έλεγχο της πρόσβασης και την εκχώρηση κωδικών για τις νέες συνεδρίες και συνδέσεις που εγκαθίστανται.



Εικόνα 2.3: Τυπική δομή δικτύου 3G

### 2.1.2.6 Personal Area Network (PAN)

Τα δίκτυα PAN είναι δίκτυα πολύ μικρής εμβέλειας. Ως ένα PAN μπορεί να θεωρηθεί δύο τερματικά συνδεδεμένα μεταξύ τους με τέτοιο τρόπο ώστε να επιτυγχάνεται η δικτύωση. Χαρακτηριστικό παράδειγμα αποτελεί η δημιουργία δικτύου PAN μέσω χρήσης καλωδίου Universal Serial Bus (USB) ή η χρήση καλωδίου Firewire. Τυπικά η απόσταση λειτουργίας τους περιορίζεται σε λίγα μέτρα. Φυσική μετεξέλιξη των PANs είναι τα Wireless PAN. Με την εκμετάλλευση τεχνολογιών όπως οι υπέρυθρες ακτίνες ή το πρωτόκολλο επικοινωνίας Bluetooth εξαλείφτηκε η ανάγκη ενσύρματης σύζευξης των τερματικών. Το Bluetooth, μια από τις κραταιές τεχνολογίες δημιουργίας WPANs επιτρέπει την σύνδεση τερματικών σε



ένα δίκτυο περιορισμένης εμβέλειας και ταχύτητας μεταφοράς δεδομένων, περί το 1 Mbps. Το δίκτυο αυτό λειτουργεί βασισμένο στην λειτουργία του Ad Hoc, η οποία ήδη έχει αναφερθεί. Ένα τερματικό αναλαμβάνει τον ρόλο του συντονιστή και τα υπόλοιπα επικοινωνούν με αυτό. Ένα PAN μέσω της τεχνολογίας Bluetooth μπορεί να υποστηρίξει μέχρι 255 συσκευές από τις οποίες μόνο οι 8 μπορούν να είναι ενεργές ταυτόχρονα. Δυνατότητα του τερματικού – συντονιστή είναι να θέτει ένα τερματικό σε λειτουργία η σε αδράνεια.

Όσο αφορά το τερματικό Bluetooth η ακτίνα κάλυψης του χαρακτηρίζεται από την κλάση του, η οποία προσδιορίζει ταυτόχρονα και την ενεργειακή απαίτηση του. Στον ακόλουθο πίνακα εμφανίζονται αναλυτικά τα στοιχεία.

	Μέγιστη Ενέργεια	Απόσταση (εκτιμάται)
Class 1	100 mW (20 dBm)	100μ.
Class 2	2.5 mW (4 dBm)	10μ.
Class 3	1 mW (0 dBm)	1μ.

Πίνακας 2.2: Στοιχεία bluetooth

Η αρχική προδιαγραφή του πρωτοκόλλου Bluetooth εξελίχθηκε στο Bluetooth v.2. Βασικές βελτιώσεις πραγματοποιήθηκαν στην ταχύτητα μεταφοράς δεδομένων, η οποία αυξήθηκε στα 2 Mbps, στη χαμηλότερη κατανάλωση ενέργειας και στη μείωση της πολυπλοκότητας στις πολλαπλές συνδέσεις μεταξύ σταθμών λόγω της αύξησης του ρυθμού μεταγωγής. Περαιτέρω βελτίωση αναμένεται να πραγματοποιηθεί στα προαναφερθέντα με την παρουσίαση της τρίτης έκδοσης του πρωτοκόλλου στην οποία θα γίνεται χρήση διαφορετικών συχνοτήτων επικοινωνίας με αποτέλεσμα την επίτευξη υψηλών ταχυτήτων μεταφοράς δεδομένων, μέχρι τα 480 Mbps.

Η παροχή ασύρματης δυνατότητας δικτύωσης τέτοιου τύπου βρήκε υποστήριξη από την θεωρία του ubiquitous computing κατά την οποία η τεχνολογία και η δικτύωση ενσωματώνεται και σε συσκευές πέραν των κλασικών τερματικών.

### 2.1.2.7 Satellite Networks

Σε κάποιες περιπτώσεις πρέπει να επιτευχθεί η ενοποίηση δικτύων που βρίσκονται σε διαφορετικές ηπείρους ή ακόμη και η σύνδεση ενός τερματικού σε ένα δίκτυο από κάποια τοποθεσία που δεν υπάρχει κάποια υποδομή που να το επιτρέπει. Σε αυτές τις περιπτώσεις μια εναλλακτική αποτελούν τα δορυφορικά δίκτυα.

Πρώτο βασικό χαρακτηριστικό ενός τέτοιου δικτύου αποτελεί η κάρτα δικτύου που επιτρέπει την πρόσβαση σε ένα τέτοιο δίκτυο. Αυτή η κάρτα μπορεί να είναι της μορφής PCI για χρήση με ένα σταθερό τερματικό ή αυτόνομη συσκευή στην οποία το τερματικό συνδέεται μέσω USB ή Ethernet. Ιδιαίτερη αναφορά κρίνεται σκόπιμο να γίνει στην κεραία του εξοπλισμού από την πλευρά του τερματικού δεδομένου ότι υπάρχουν αρκετές παραλλαγές. Δύο είναι οι βασικοί άξονες στον τομέα των κεραιών. Ο πρώτος είναι η αμφίδρομη ή όχι επικοινωνία του τερματικού με το δορυφορικό δίκτυο και ο δεύτερος είναι η κινητικότητα του τερματικού. Σύμφωνα με τον πρώτο άξονα αν η επικοινωνία δεν είναι αμφίδρομη τότε χρειάζεται ένα δορυφορικό πιάτο και ένα Low Noise Block (LNB) για την αναγνώριση του σήματος που εκπέμπεται από τον δορυφόρο και αφορά το downlink της σύνδεσης, τούτέστιν την αποστολή δεδομένων από τον δορυφόρο προς το τερματικό. Σε αυτό το μοντέλο ένας εναλλακτικός δίκτυο χρησιμοποιείται για να καλυφθεί η απώλεια του uplink, όπως μια σύνδεση PSTN. Στην αντίθετη περίπτωση η χρήση ενός LNB και ενός Block Up Converter (BUC) είναι απαραίτητη. Ο δεύτερος άξονας είναι η κινητικότητα του τερματικού. Δεδομένου πως το φυσικό μέσο για άλλη μια φορά είναι ο αέρας παρέχεται ένας βαθμός κινητικότητας του τερματικού. Φυσικά κάτι τέτοιο προϋποθέτει διαφορετικό τύπο κεραίας δεδομένου πως πρέπει να αναπροσαρμόζεται δυναμικά στον δορυφόρο σύμφωνα με την κίνηση του τερματικού. Αυτό μπορεί να γίνει είτε με ένα ηλεκτρονικά

περιστρεφόμενο σε δύο άξονες βραχίονα ο οποίος στηρίζει το δορυφορικό πιάτο είτε με ειδικά κατασκευασμένες κεραίες.

Δεύτερο χαρακτηριστικό είναι το δίκτυο των δορυφόρων που θα χρησιμοποιηθούν από το τερματικό. Υπάρχουν αρκετά δίκτυα τα οποία υποστηρίζουν υψηλές ταχύτητες μεταφοράς δεδομένων κάνοντας χρήση της τεχνολογίας Very Small Aperture Terminal (VSAT) η οποία παρέχει αμφίδρομη επικοινωνία του τερματικού με τον δορυφόρο. Κάποιες από τις πιο γνωστές υπηρεσίες είναι οι ακόλουθες. Το Astra2connect το οποίο χρησιμοποιεί το δίκτυο δορυφόρων Astra με σύνολο δύο γεωστατικούς δορυφόρους για την παροχή της υπηρεσίας για την παροχή κάλυψης στην Ευρώπη, το Inmarsat Broadband Global Area Network (BGAN) το οποίο παρέχει παγκόσμια κάλυψη με ένα δίκτυο τριών δορυφόρων. Αδελφό σύστημα δε θεωρείται και το fleet broadband το οποίο χρησιμοποιεί την ίδια υποδομή και παρέχει τις ίδιες υπηρεσίες αλλά χρησιμοποιείται από την Ναυτιλία. Ακόμη το Iridium Openport, το οποίο αποτελείται από ένα σύστημα εξήντα έξη δορυφόρων χαμηλής τροχιάς για την παροχή παγκόσμιας κάλυψης. Ακόμη υπάρχει η υπηρεσία StarBand η οποία χρησιμοποιεί δύο δορυφόρους, τον GE-4 και τον Telstar 7 για την παροχή κάλυψης στην Αμερική. Τέλος μέσω του HellasSat 2 παρέχεται κάλυψη στην ευρύτερη περιοχή της μεσογείου.

Άξιο αναφοράς κρίνεται πως σε όλες τις διαθέσιμες λύσεις παρέχεται η υπηρεσία σε δύο μορφές. Με και χωρίς εγγυημένο εύρος ζώνης.

#### **2.1.2.8 Ψηφιακά δίκτυα ευρείας εκπομπής (DVB / DAB)**

Ένας τύπος δικτύων με μεγάλο ενδιαφέρον λόγω του μελλοντικού ρόλου τους είναι τα Digital Video Broadcast (DVB) και Digital Audio Broadcast (DAB) δίκτυα. Αρχικά η βασική διαφοροποίηση των συγκεκριμένων δικτύων με τα προηγούμενα προκύπτει από το γεγονός πως χρησιμοποιούν συνήθως simplex (χωρίς αυτό να αποκλείει την χρήση duplex) και broadcast μετάδοση. Προκάτοχος των εν λόγω δικτύων ήταν τα δίκτυα ευρείας αναλογικής εκπομπής τα οποία χρησιμοποιήθηκαν για την παροχή μαζικών υπηρεσιών

επικοινωνίας (ενημέρωση, ψυχαγωγία κ.α.). Σαν δομή δικτύου δεν παρουσιάζεται κάτι σύνθετο δεδομένου ότι αποτελείται από ένα σύστημα πομπών και δεκτών οπότε κρίνεται σκόπιμη η περεταίρω ανάλυση των τεχνολογιών.

Αρχικά τα εν λόγω δίκτυα αλλάζουν τον τρόπο μετάδοσης στο φυσικό στρώμα κάνοντας χρήση μιας και μόνο συχνότητας (Single Frequency). Με αυτόν τον τρόπο απλουστεύεται η ραδιοκάλυψη καθώς μια συχνότητα ανατίθεται σε έναν πάροχο για την καθολική κάλυψη της περιοχής στην οποία δρα. Ακόμη απαλείφονται οι παρεμβολές μεταξύ σημάτων. Για την εξασφάλιση του τελευταίου συνήθως χρησιμοποιείται τεχνολογία OFDM. Βασική προϋπόθεση για την αδιάλειπτη λειτουργία ενός τέτοιου δικτύου αποτελεί ο καθολικός και τέλειος συγχρονισμός εκπομπής των πομπών. Κάτι τέτοιο επιτυγχάνεται συνήθως με την χρήση Global Positioning System (GPS).

Τα DAB συστήματα είναι σε θέση να παρέχουν ψηφιακό ήχο υψηλής ποιότητας μαζί με ένα σύνολο υπηρεσιών προστιθέμενης αξίας όπως υπηρεσίες κειμένων ενημέρωσης και ηλεκτρονικό οδηγό προγράμματος. Βασικό πρόβλημα που προέκυψε κατά την υλοποίηση τέτοιων συστημάτων ήταν πως για την εξοικονόμηση φάσματος εκπομπής ήταν άμεσα αλληλένδετα με τους χαμηλούς ρυθμούς μετάδοσης σε συνδυασμό με την χρήση της MPEG2 κωδικοποίησης είχαν ως αποτέλεσμα την χαμηλή ποιότητα της παρεχόμενης υπηρεσίας. Διέξοδος σε αυτό το πρόβλημα δόθηκε από την εξέλιξη του προτύπου DAB, το DAB+ στο οποίο η κωδικοποίηση του ήχου πραγματοποιείται με το κωδικοποιητή / αποκωδικοποιητή Advanced Audio Coding (AAC). Ο συγκεκριμένος είναι σε θέση να παρέχει καλύτερη συμπίεση από τον προκατόχο του. Τέλος μια άλλη έκδοση του είναι το DAB/IP το οποίο παρέχει τις ίδιες υπηρεσίες αλλά βασίζεται στο πρωτόκολλο IP.

Τα DVB συστήματα είναι σε θέση να παρέχουν ευρεία διανομή ψηφιακών πολυμέσων παράλληλα με υπηρεσίες προστιθέμενης αξίας ακόμα και υπό την μορφή συνδρομητικών υπηρεσιών. Η ανάπτυξη του συγκεκριμένου ξεκίνησε το 1993 και σήμερα υπάρχουν ήδη ένα σύνολο από πρότυπα όπως το DVB-Terrestrial (DVB-T), το DVB-Cable (DVB-C), το DVB-Satellite (DVB-S) και τέλος το DVB-Handheld (DVB-H) για να καλύψουν μέσω τεχνικών τροποποιήσεων τόσο τους διάφορους τρόπους μετάδοσης όσο και τους τύπους τερματικών. Το DVB σαν πρότυπο είναι σε θέση να μεταδώσει τόσο πολυμεσικό υλικό

τυπικής ποιότητας (SD) όσο και υψηλής ποιότητας (HD) μέσω κωδικοποίησης MPEG-2 ή MPEG-4 AVC. Βασικά χαρακτηριστικά του DVB είναι η δυνατότητα ιεραρχικής μετάδοσης χαρακτηρίζοντας τις ροές δεδομένων ως υψηλής και χαμηλής προτεραιότητας και η δυνατότητα για πρόσω διόρθωση σφαλμάτων (Forward Error Correction – FEC). Κατά την υψηλή προτεραιότητα έχει χαμηλό ρυθμό μετάδοσης και μεγαλύτερη ανθεκτικότητα ενώ η χαμηλή προτεραιότητα έχει υψηλό ρυθμό μετάδοσης και μικρότερη ανθεκτικότητα.

Αν και υποκατηγορία το DVB-H είναι αρκετά σημαντικό καθώς παρέχει την υπηρεσία DVB σε φορητά τερματικά. Οι βασικές απαιτήσεις από το συγκεκριμένο πρότυπο είναι η παροχή της υπηρεσίας με αυξημένη ισχύ εκπομπής ώστε να υποστηρίζονται τερματικά σε εσωτερικούς χώρους, να βελτιωθεί ο αλγόριθμος FEC για την αντιμετώπιση καταστάσεων φτωχής λήψης, να υποστηρίζονται δυνατότητες μεταπομπής και να δοθεί έμφαση στην εξοικονόμηση ενέργειας του τερματικού καθώς αυτό αποτελεί ένα πολύ σημαντικό παράγοντα στα φορητά τερματικά. Τέλος υιοθετεί το μοντέλο IP Datacasting (IPDC). Βάσει αυτού υποστηρίζονται ένα σύνολο από συμβατές κωδικοποιήσεις εικόνας και ήχου.

#### 2.1.2.9 Global Area Networks (GAN)

Τα δίκτυα GAN είναι δίκτυα τα οποία εκμεταλλευόμενα πολλαπλές τεχνολογίες δικτύου παρέχουν συνεχόμενη συνδεσιμότητα του τερματικού το οποίο συνδέεται σε αυτά παράλληλα με την παροχή εγγυημένων υπηρεσιών. Μέχρι και σήμερα τέτοιου είδους δίκτυα βρίσκονται σε θεωρητικό επίπεδο και για αυτό τον λόγο δεν μπορούν να αναφερθούν δεδομένα αντίστοιχα με τα προηγούμενα δίκτυα.

Η IEEE αναπτύσσει την προδιαγραφή 802.20, γνωστή και ως Mobile Broadband Wireless Access (MBWA) για δίκτυα του συγκεκριμένου τύπου. Κάποιες από τις βασικές απαιτήσεις που προκύπτουν είναι οι ακόλουθες. Η παροχή κάλυψης μέσω roaming και μεταπομπών σε ταχύτητες άνω του 1 Mbps, η αναγνώριση των τερματικών μέσω καινούργιου τύπου Medium Access Control (MAC) διευθύνσεων, η υποστήριξη πλήρους κινητικότητας του τερματικού σε ταχύτητες έως 250 Km/H, η υλοποίηση μέσω της τεχνολογίας PS και παροχή

μικρών καθυστερήσεων δικτύου και η χρήση συχνοτήτων κάτω των 3.5GHz με την κατοχύρωση τους από τον πάροχο.

### 2.1.3 Συγκριτικά στοιχεία δικτύων

Κατόπιν της εκτενούς αναφοράς στα κυριότερα δίκτυα υψηλών ταχυτήτων κρίνεται σκόπιμη η σύγκριση των χαρακτηριστικών τους. Αυτό πραγματοποιείται στον ακόλουθο πίνακα.

	Maximum Distance	Maximum Speed	Physical Layer
PAN	10m	200Mbps	Wired
LAN	100m	1Gbps	Wired
WAN	-	-	Mixed
WPAN	100m	2Mbps	Wireless
WLAN	100m (per base)	125Mbps	Wireless
WiMax	10Km	70Mbps	Wireless
3G	-	0,384Mbps	Wireless
Satellite	-	2Mbps	Wireless
GAN	-	-	Mixed

Πίνακας 2.1: Συγκριτικά στοιχεία διαφόρων τεχνολογιών

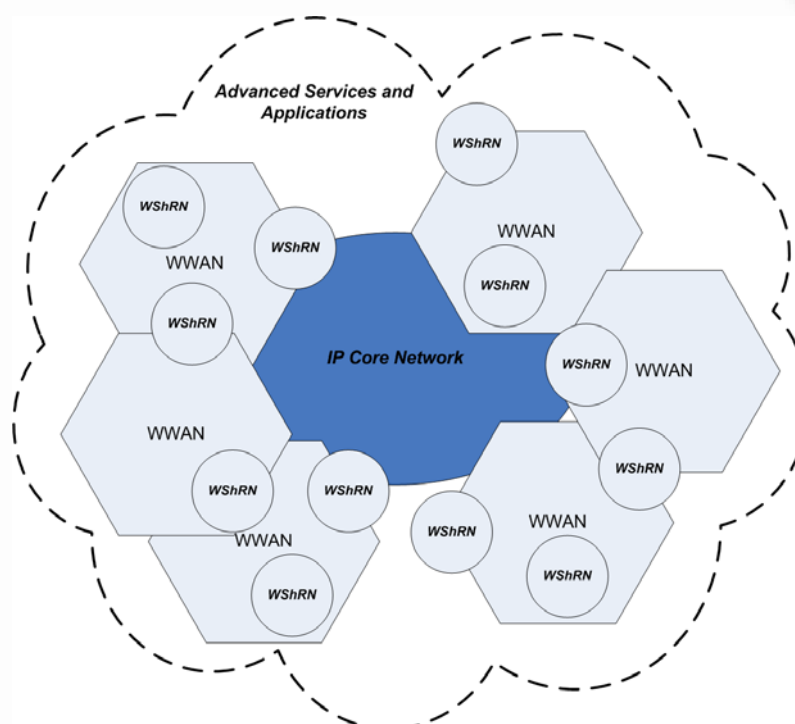
## 2.2. Χαρακτηριστικά συστημάτων B3G

Καθώς η παγκοσμιοποίηση επιφέρει τη ανάγκη καθολικής αντιμετώπισης των θεμάτων που αναφέρονται στις επικοινωνίες, οι ολοένα αυξανόμενες ανάγκες των χρηστών που δημιουργούνται από το σύστημα των επικοινωνιών καθαυτό επιτάσσουν την πραγματοποίηση τεράστιων επενδύσεων στον τομέα της έρευνας, που στόχο έχει την πραγμάτωση του οράματος της παροχής στο χρήστη συνεχούς και αψευγάδιαστης συνδεσιμότητας, οπουδήποτε και οποτεδήποτε, με τρόπο οικονομικά συμφέροντα.

Με την εισροή των 3G συστημάτων και χωρίς να έχουν παραγκωνιστεί τα ήδη υπάρχοντα συστήματα 2G και 2.5G, ο σημερινός κόσμος των τηλεπικοινωνιών χαρακτηρίζεται από τη συνύπαρξη τεχνολογιών (Radio Access Technologies – RATs), οι οποίες μπορούν κάλλιστα να ταξινομηθούν σε 2 μεγάλες οικογένειες, ήτοι:

- Την οικογένεια ασυρμάτων τεχνολογιών ευρείας περιοχής (Wireless Wide Area Networks - WWANs), που συμπεριλαμβάνει τις κινητές επικοινωνίες 2G/2.5G/3G, την οικογένεια πρωτοκόλλων IEEE 802.16, το πρωτόκολλο WiMAX και τις τεχνολογίες εκπομπής (broadcasting technologies), και
- Την οικογένεια ασυρμάτων τεχνολογιών περιορισμένης κλίμακας (Wireless Short Range Networks - WShRN), που περιλαμβάνει ασύρματα τοπικά / προσωπικά δίκτυα (Wireless Local Area Networks – WLANs / Wireless Personal Area Networks - WPANs), όπως και τα ασύρματα δίκτυα αισθητήρων (Wireless Sensor Networks) [11][14][15].

Αυτή η κατάσταση απεικονίζεται στην παρακάτω εικόνα.



Εικόνα 2.3: Το όραμα B3G

Μια τέτοια συνύπαρξη πληθώρας τεχνολογιών σε συνδυασμό με τις ολοένα αυξανόμενες προσδοκίες των χρηστών για παγκοσμίας εμβέλειας δυνατότητες στις τηλεπικοινωνίες, έχει φέρει στο προσκήνιο προσπάθειες διασυνεργασίας μεταξύ των ανωτέρω (μέχρι πρότινος ανταγωνιστικών) τεχνολογιών και κοινής λειτουργίας τους άνωθεν μια κοινής (καθολικής) υποδομής ασύρματης πρόσβασης (wireless access infrastructure), η οποία εκφράζει το λεγόμενο όραμα B3G. Η πραγμάτωση ενός τέτοιου οράματος θα έλυνε κυριολεκτικά τα χέρια των παρόχων δικτύου (Network Operators – NOs), λόγω των εναλλακτικών λύσεων που υπόσχεται, σε περιπτώσεις που ένας NO αδυνατεί να εξυπηρετήσει τους χρήστες της περιφέρειάς του. Με αυτό το σκεπτικό η διασυνεργασία μεταξύ των διαφόρων RATs μοιάζει να αποτελεί προαπαιτούμενο, για τη θεμελίωση του οράματος B3G.

Η έννοια των συνεργαζόμενων δικτύων (cooperative networks), ιδανική στο άκουσμά της, έχει αναπτυχθεί με στόχο την αύξηση της αποδοτικότητας των παρεχομένων υπηρεσιών και την εκμετάλλευση των πλεονεκτημάτων διαφορετικής φύσης RATs, με παράλληλο εκμηδενισμό των μειονεκτημάτων τους. Κεντρική ιδέα αποτελεί το γεγονός της συνύπαρξης



τεχνολογιών, όπως GSM, GPRS, UMTS, WLAN, WiMAX, DVB, WSN, ως συστατικών στοιχείων μιας ετερογενούς υποδομής ασύρματης πρόσβασης. Η συνύπαρξη αυτή δε, πραγματοποιείται με τρόπο συμπληρωματικό και όχι ανταγωνιστικό. Σε μια τέτοια υποδομή, ένας NO δύναται να κατέχει άδειες λειτουργίας για περισσότερα από ένα RATs. Ταυτόχρονα, έχει τη δυνατότητα να συνεργάζεται με άλλους NOs, ώστε να βασίζεται σε εναλλακτικές λύσεις, σε περίπτωση που ο ίδιος αδυνατεί να καλύψει την απαιτούμενη χωρητικότητα στο δίκτυό του ή τα απαιτούμενα επίπεδα ποιότητας υπηρεσίας (QoS), με τρόπο οικονομικά αποδεκτό.

Πώς καθίστανται όμως αυτές οι εναλλακτικές λύσεις εκμεταλλεύσιμες; Με άλλα λόγια, ποιοι είναι οι μηχανισμοί διαχείρισης τέτοιων συστημάτων, σε ποιες αρχές στηρίζονται και τι υπηρεσίες μπορούν να προσφέρουν; Αυτό είναι το αντικείμενο της επόμενης παραγράφου.

### 2.3. Η διαχείριση δικτύων B3G

Σε απάντηση των παραπάνω, οι εναλλακτικές λύσεις που επικαλούνται οι πάροχοι δικτύων B3G καθίστανται εκμεταλλεύσιμες μέσω κάποιου συστήματος διαχείρισης (management system) που προσαρτάται σε ένα RAT, με τη βοήθεια του οποίου οι χρήστες κατευθύνονται στο καταλληλότερο RAT σε διαφορετικά χωροχρονικά πλαίσια, βάσει των απαιτήσεών τους και κάποιων κριτηρίων αποδοτικότητας των δικτύων. Ένα σύστημα διαχείρισης με τα προαναφερθέντα χαρακτηριστικά εφαρμόζεται μεν σε κάθε τεχνολογία χωριστά, αλλά οι πλατφόρμες των διαφορετικών RATs δύναται να συνεργάζονται. Αυτό αποτελεί μια καθ'όλα ρεαλιστική προσέγγιση, μέσω της οποίας ο κάθε NO διατηρεί απόρρητες τις πληροφορίες σχετικά με τη δομή του δικτύου του, παρά την όποια συνεργασία με τους μέχρι πρότινος ανταγωνιστές του. Πρέπει δε να σημειωθεί πως η έννοια των συνεργαζόμενων δικτύων υποδηλώνει την ύπαρξη μιας κεντρικής υποδομής που βασίζεται στο πρωτόκολλο IPv4 (Internet Protocol Version 4), δυνάμενη ωστόσο να εφαρμοστεί

συνεργαζόμενη με το πρωτόκολλο IPv6, το οποίο θα μείωνε τις απαιτούμενες τεχνολογίες δικτύου και την επακόλουθη πολυπλοκότητα.

Σε γενικές γραμμές, η ιδέα της διάθεσης μιας πληθώρας συνεργαζόμενων RATs στο βωμό του κοινού στόχου για παροχή υψηλής ποιότητας υπηρεσιών, φαντάζει εξαιρετικά ελκυστική για τους NOs. Αυτό δικαιολογείται εφόσον, όποτε συναντούν δυσκολίες στο δίκτυό τους, μπορούν να βασίζονται σε εναλλακτικές λύσεις, μέσω της δυνατότητας εξυπηρέτησης των πελατών τους από συνεργαζόμενους παρόχους και μάλιστα με χρήση του ίδιου ή και άλλου RAT. Έτσι καθίσταται βέβαιη η μη υποβάθμιση της ποιότητας των παρεχομένων υπηρεσιών.

Μολαταύτα, όσο ελκυστική και να φαίνεται μια τέτοιου είδους συνεργασία, ο απώτερος στόχος της παροχής αψεγάδιαστης συνδεσιμότητας (ακόμα και εν κινήσει) είναι ακόμα μακρινός και ο δρόμος που οδηγεί προς αυτόν δύσβατος. Ο πρωταρχικός λόγος για αυτό συνίσταται σε πιθανές αντιδράσεις που δύναται να εγείρει ένα επιχειρησιακό μοντέλο που στηρίζεται στη συνεργασία μέχρι πρότινος ανταγωνιστών. Ένας δεύτερος, παράπλευρος λόγος έγκειται στο ότι η συνεργασία, ως περιγράφηκε, απαιτεί την εκ των προτέρων (a priori) εγκατάσταση του απαραίτητου λογισμικού για τη λειτουργία των RATs σε τερματικά και στοιχεία δικτύου, πράγμα το οποίο προϋποθέτει συνεχείς, πιθανόν ριψοκίνδυνες και σίγουρα δαπανηρές επενδύσεις, οποτεδήποτε εισάγονται στις παγκόσμιες επικοινωνίες, νέες τεχνολογίες.

Για να παρακαμφθεί η επιρροή τέτοιων μειονεκτημάτων στην τεχνολογική ανάπτυξη, υποχρεωτικά η επιστήμη στρέφεται στην κατεύθυνση δικτύων και τερματικών ευέλικτων, ικανών να προσαρμόζονται στις διαρκώς μεταβαλλόμενες συνθήκες του περιβάλλοντος. Μόνο με αυτόν τον τρόπο θα επιτευχθεί το πολυπόθητο όραμα της αδιάλειπτης συνδεσιμότητας. Αυτές οι ικανότητες αντικατοπτρίζονται στα λεγόμενα «δυναμικά αναδιαρθρούμενα δίκτυα» (reconfigurable networks) και ακόμα περισσότερο στα «γνωσιακά δίκτυα» (cognitive networks - CN). Αυτές οι κατηγορίες δικτύων αποτελούν και το αντικείμενο των ακολούθων 2 παραγράφων. Ένα B3G σύστημα είναι ένα δίκτυο το οποίο βασίζεται σε ήδη υπαρκτά δίκτυα. Βασική ιδέα που διακατέχει αυτά τα συστήματα είναι η συνεργασία αυτών των δικτύων με σκοπό την αδιάλειπτη παροχή υπηρεσιών σε

κάθε τερματικό που συνδέεται σε αυτό. Βασική αρχή για την θεωρητική ανάπτυξη τους αποτελεί η φιλοσοφία “always best connected”. Σύμφωνα με αυτή κάθε τερματικό θα είναι σε θέση να εκμεταλλευτεί τις δυνατότητες τις κάθε επιμέρους τεχνολογίας που παρέχουν τα δίκτυα που απαρτίζουν το B3G σύστημα έτσι ώστε να εξασφαλίζει τους πόρους και τα χαρακτηριστικά για την αδιάλειπτη παροχή των υπηρεσιών που επιθυμεί ανά πάσα στιγμή. Κάποια από τα πιο βασικά χαρακτηριστικά της ποικιλίας που παρέχεται περιέχει διαφοροποιήσεις σε εύρος ζώνης, ακτίνα κάλυψης, καθυστέρηση, επίπεδο ασφάλειας, απαιτούμενη ισχύς και κόστος υπηρεσίας.

Έχοντας αναλύσει τις σημαντικότερες από τις τεχνολογίες οι οποίες θα αποτελέσουν τα μέρη της σύνθετης B3G ραδιο-υποδομής, στην συνέχεια θα αναλυθούν τα χαρακτηριστικά και οι απαιτήσεις που τίθενται στα συστήματα αυτά, τόσο από την πλευρά της υποδομής του δικτύου όσο και από την πλευρά του εξοπλισμού των τερματικών σταθμών.

#### **2.4. Η έννοια της δυναμικής αναδιάρθρωσης**

Η ικανότητα δυναμικής αναδιάρθρωσης (Reconfigurability) αποτελεί εξέλιξη της έννοιας του Software Defined Radio (SDR) και συνιστά μια έννοια που σκοπό έχει να διευκολύνει τη διασυνεργασία μεταξύ τεχνολογιών, ξένων μεταξύ τους και έτσι να προσφέρει εναλλακτικές επιλογές ασύρματης πρόσβασης σε συγκεκριμένες περιοχές εξυπηρέτησης (service areas). Αυτό το επιτυγχάνει προσφέροντας σε τερματικές συσκευές, αλλά και σε στοιχεία δικτύου, τη δυνατότητα να αναπροσαρμόζονται στις εκάστοτε συνθήκες, με την παροχή μηχανισμών που καθιστούν πραγματοποιήσιμη την αλλαγή παραμέτρων των πρωτοκόλλων μιας τεχνολογίας και της εν γένει διάρθρωσής της, όποτε οι εξωτερικές συνθήκες το επιβάλλουν. Η αναδιάρθρωση δύναται να πραγματοποιηθεί όχι μόνο με τα προεγκατεστημένα κομμάτια λογισμικού σε κάθε τερματικό, αλλά και με δυναμικό (online) κατέβασμα (download) του απαιτούμενου λογισμικού, όπως και ασφαλή εγκατάστασή του, με στόχο τη λειτουργία της νέας τεχνολογίας. Η ιδέα αυτή αποτελεί μια ειδοποιό διαφορά στα συστήματα B3G εν συγκρίσει με την παραδοσιακή στατικότητα που χαρακτήριζε μέχρι σήμερα τα τερματικά και τα στοιχεία ενός παραδοσιακού δικτύου ασυρμάτων

επικοινωνιών. Με άλλα λόγια, δεν αποτελεί σενάριο επιστημονικής φαντασίας ένα τμήμα δικτύου το οποίο αναδιαρθρώνεται, χρησιμοποιώντας διαφορετική τεχνολογία, από τη μέχρι πρότινος χρησιμοποιούμενη, επιλέγοντας κάθε στιγμή την καταλληλότερη.

### **Είδος ασύρματης επικοινωνίας**

### **Βασικά Χαρακτηριστικά Τερματικών και Στοιχείων Δικτύου**

*Κλασικές ασύρματες επικοινωνίες*

Ένα RAT / τερματικό

Ένα RAT / στοιχείο δικτύου

Στατικός καθορισμός παραμέτρων  
δικτύου

*Συνεργαζόμενα Δίκτυα B3G*

Τα τερματικά επιλέγουν ανάμεσα σε  
πολλά RATs. Εναλλακτικά υπάρχει η  
δυνατότητα ταυτόχρονης χρήσης πολλών  
RATs.

Ένα RAT ανά στοιχείο δικτύου (βασικό  
χαρακτηριστικό)

Προεγκατάσταση (offline) του λογισμικού  
και του υλικού που απαιτείται για τη  
λειτουργία εναλλακτικών RATs

*Δυναμικά αναδιαρθρούμενα δίκτυα*

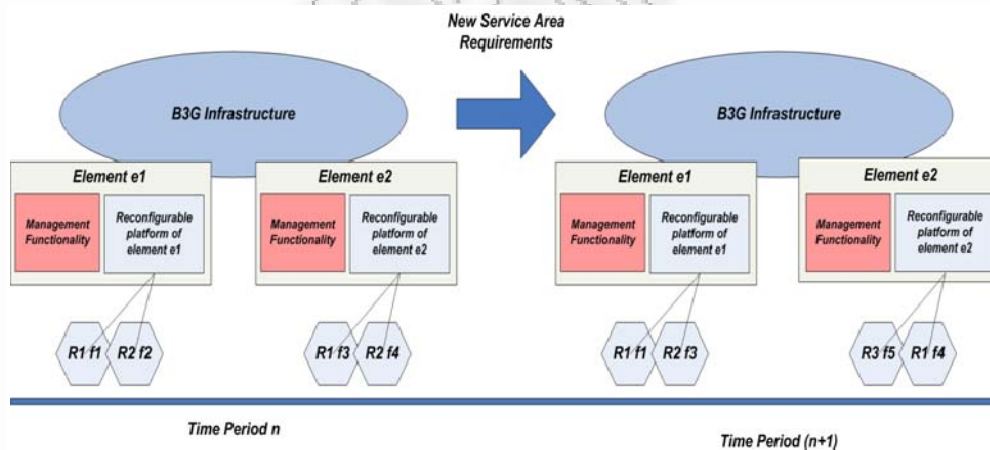
Τα τερματικά επιλέγουν ανάμεσα σε  
πολλά RATs. Εναλλακτικά υπάρχει η  
δυνατότητα ταυτόχρονης χρήσης πολλών  
RATs.

Δυναμική (online) εγκατάσταση και  
καθορισμός παραμέτρων του λογισμικού

που απαιτείται για τη χρήση του  
επιλεγμένου RAT, κατά τη λειτουργία του  
τερματικού και των στοιχείων δικτύου.

Εικόνα 2-4: Βασικές έννοιες ασυρμάτων επικοινωνιών

Η αναδιάρθρωση εν γένει συνιστά διαδικασία που επηρεάζει πολλά στρώματα του OSI (cross-layer process). Ωστόσο, επικεντρώνοντας το ενδιαφέρον μας στα κατώτερα των στρωμάτων, μόνο και μόνο για να δοθεί μια σαφής εικόνα της αναδιάρθρωσης, θα λέγαμε ότι μια πλατφόρμα που προσφέρει τη δυνατότητα δυναμικής αναδιάρθρωσης, παρέχει στους πομποδέκτες της τη δυνατότητα (επί παραδείγματι) να επιλέγουν την εκάστοτε καταλληλότερη τεχνολογία και συχνότητα προς λειτουργία. Αυτή η δυνατότητα θα αποτελέσει και βασικό κομμάτι της διατριβής και προς κατανόησή της παρατίθεται και το ακόλουθο σχήμα (Εικόνα 2-5).



Εικόνα 2.5: Πλατφόρμα δυναμικά αναδιαρθρούμενη

Παρατηρούμε ότι κάθε στοιχείο (element) μιας υποδομής B3G, δύναται να αναπροσαρμόζει (βάσει νέων συνθηκών και απαιτήσεων) τη συχνότητα λειτουργίας του, το RAT του, ή και τα δύο των ανωτέρω. Αυτό επιτυγχάνεται με τη χρήση έξυπνων μηχανισμών διαχείρισης, σαν αυτούς που θα παρουσιαστούν στα επόμενα κεφάλαια.

## 2.5 Τα cognitive δίκτυα επικοινωνιών

Λαμβάνοντας υπόψη την αυξημένη πολυπλοκότητα που χαρακτηρίζει τα B3G συστήματα (εξαιτίας τόσο των διαρκώς και περισσότερο καινοτόμων παρεχομένων υπηρεσιών, όσο και της συνύπαρξης και συνεργασίας συστημάτων διαφορετικής φύσης), καθίσταται δύσκολη η εξελικτική πορεία των ασυρμάτων επικοινωνιών. Αυτό συμβαίνει διότι με την παρούσα λειτουργική αρχιτεκτονική των ασυρμάτων επικοινωνιών, η επικοινωνία μεταξύ των διαφόρων οντοτήτων κάθε δικτύου περιορίζεται σημαντικά από τη διαστρωματωμένη αρχιτεκτονική των πρωτοκόλλων, πράγμα το οποίο δεν επιτρέπει σε ένα στοιχείο δικτύου να διαθέτει γνώση της κατάστασης στην οποία βρίσκονται άλλα στοιχεία. Με αυτή τη λογική, οποιαδήποτε ανταλλαγή πληροφορίας μεταξύ στοιχείων δικτύου είναι εξαιρετικά χρονοβόρα, οπότε καθίσταται εξαιρετικά δύσκολη και αργή η προσαρμογή του όλου δικτύου σε τυχόν αλλαγές που επιτάσσονται από εξωτερικά αίτια.

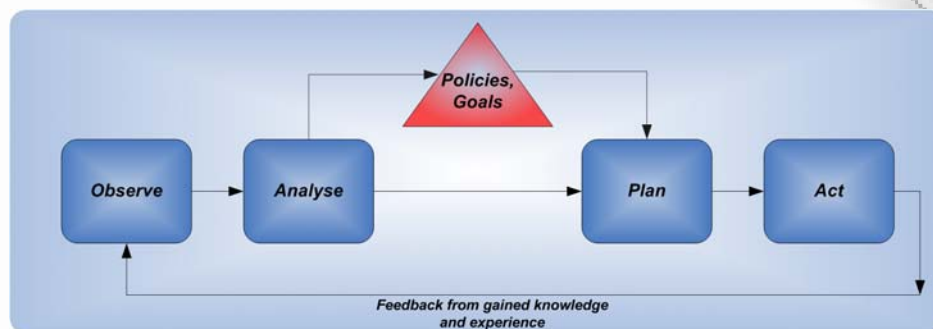
Στον αντίποδα αυτών και με στόχο να χρησιμοποιηθούν συμπληρωματικά ως προς την όλη ασύρματη B3G υποδομή, βρίσκονται τα λεγόμενα γνωσιακά δίκτυα επικοινωνιών (cognitive networks), τα οποία αναμένεται ότι θα διευκολύνουν σημαντικά την επίτευξη του στόχου της συνδεσιμότητας οποτεδήποτε, οπουδήποτε και με οποιονδήποτε τρόπο.

Σε γενικές γραμμές, “cognitive” μπορεί να ονομαστεί ένα σύστημα το οποίο διαθέτει τη δυνατότητα να γνωρίζει το βέλτιστο τρόπο με τον οποίο θα πρέπει να ανταποκριθεί στις επιταγές του εξωτερικού του περιβάλλοντος. Η εν λόγω γνώση προέρχεται από τη συνεχή ανταλλαγή πληροφοριών μεταξύ του συστήματος και του περιβάλλοντός του, μετατρέπεται σταδιακά σε εμπειρία και υποστηρίζεται από την ύπαρξη μηχανισμών λήψης αποφάσεων που δύνανται να μαθαίνουν τι απόφαση πρέπει να λάβουν. Στην περίπτωση των ασυρμάτων δικτύων, ο cognitive χαρακτήρας συνίσταται στη δυνατότητα γνώσης του βέλτιστου τρόπου με τον οποίο τα δίκτυα πρέπει να προσαρμοστούν στις διαρκώς μεταβαλλόμενες εξωτερικές συνθήκες. Θα λέγαμε ότι τα cognitive δίκτυα, με τον τρόπο λειτουργίας τους, ελαχιστοποιούν (ακόμα περισσότερο από την περίπτωση των reconfigurable δικτύων) την εκ των προτέρων αναγκαία εγκατάσταση οποιουδήποτε λογισμικού σε τερματικά και στοιχεία δικτύου. Αντίθετα, διαθέτοντας απλά τους

κατάλληλους μηχανισμούς που διατηρούν πληροφορία από παρελθούσες αλληλεπιδράσεις του δικτύου με το περιβάλλον, δύνανται να προσαρμόζουν καταλλήλως λειτουργικές τους παραμέτρους και άρα και τη συμπεριφορά τους, ανταποκρινόμενα έτσι σε εξωτερικές διεγέρσεις. Σε συγκεκριμένες περιπτώσεις μάλιστα, η προσαρμογή πραγματοποιείται και εκ των προτέρων, λόγω της γνώσης του πότε είναι πιθανό να χρειαστεί.

Η προσαρμογή των cognitive δικτύων στις μεταβαλλόμενες απαιτήσεις του περιβάλλοντος πραγματοποιείται μέσω αναδιάρθρωσης των ιδίων λειτουργικών τους παραμέτρων, αναδιάρθρωση η οποία δύνανται να επηρεάσει όλα τα στρώματα λειτουργίας ενός πρωτοκόλλου, όπως το φυσικό (PHY), το MAC (Medium Access Control), το LLC (Logical Link Control), τα στρώματα δικτύου, μεταφοράς, κλπ. Η αναδιάρθρωση στα στρώματα PHY/MAC περιλαμβάνει την επιλογή των καταλληλότερων RAT(s) και φάσματος για λειτουργία (εδώ υπάρχουν οι μηχανισμοί αναδιάρθρωσης που αναφέρθηκαν ανωτέρω, οι οποίοι δείχνουν και τη συμπληρωματικότητα των cognitive ως προς τα reconfigurable δίκτυα). Στο επίπεδο δικτύου τα cognitive δίκτυα παρέχουν μηχανισμούς για την κατανομή της κίνησης στα επιλεχθέντα RATs, με τρόπο που εξασφαλίζει την ικανοποίηση δεδομένων κριτηρίων. Στο επίπεδο εφαρμογών, διατίθενται μηχανισμοί για την παροχή των υψηλότερων δυνατών επιπέδων QoS στους χρήστες.

Με αυτή τη λογική, προβλέπεται ότι οι δικτυακές δυνατότητες που διαθέτουν τα cognitive δίκτυα θα συμβάλουν στη μείωση των εξόδων υποδομής (Capital Expenditure - CAPEX) και παράλληλα τον έλεγχο των λειτουργικών εξόδων (Operational Expenditure - OPEX), κατά την εγκατάσταση και θέση σε λειτουργία υποδομών B3G. Προς τούτο, διενεργείται έρευνα σε μεγάλη κλίμακα πάνω σε ποικίλα θέματα που αγκαλιάζουν την τεχνολογία των cognitive δικτύων, τόσο υπό τη σκέπη ερευνητικών προγραμμάτων, όσο και υπό άλλους οργανισμούς παγκόσμιας εμβέλειας [17][25]. Τέτοιες εργασίες έχουν στόχο τόσο τη θεμελίωση του πλαισίου στο οποίο θα βασιστεί η εμπορική είσοδος των γνωσιακών δικτύων, όσο και την αναγνώριση των σχετικών με αυτά περιοχών δουλειάς. Η εικόνα 2-6 απεικονίζει γραφικά την αρχή λειτουργίας των cognitive δικτύων.



Εικόνα 2.6: Αρχή λειτουργίας cognitive δικτύων

Παρατηρούμε ότι το δίκτυο αδιαλείπτως παρακολουθεί (observes / monitors) το περιβάλλον, με στόχο να ανακαλύψει τυχόν αλλαγές που δύνανται να επηρεάσουν τη λειτουργία του. Οι παρατηρήσεις υπόκεινται σε ανάλυση (analyse) και αποτελούν τη βάση για τη λήψη της απόφασης αν απαιτείται κάποιας μορφής αναδιάρθρωση. Μόλις ληφθεί η σχετική απόφαση (plan), το δίκτυο δρα αναλόγως (act). Η διαδικασία αυτή επαναλαμβάνεται υπό τη μορφή βρόχου, ο οποίος καθοδηγείται (ελέγχεται) από ένα σύνολο από πολιτικές / στρατηγικές (policies) και στόχους (goals) αποφάσεων. Η όλη επαναλαμβανόμενη διαδικασία στηρίζεται σε μηχανικές μεθόδους εκμάθησης (machine learning).

## 2.6 Προκλήσεις σχετικές με τη χρήση του φάσματος στα δίκτυα B3G

Με τον όρο «φάσμα» εννοούμε οποιοδήποτε εύρος στην κλίμακα των συχνοτήτων. Συχνά μάλιστα ο όρος υποδηλώνει το συγκεκριμένο εύρος που είναι απαραίτητο για την επιτυχή σύνδεση μεταξύ ενός πομπού και ενός δέκτη. Και αυτό διότι από την εποχή της πρώτης επιτυχούς μετάδοσης και λήψης ενός ραδιοφωνικού σήματος που πραγματοποιήθηκε από τον Guglielmo Marconi, το «ραδιο-φάσμα» αποδείχτηκε το περισσότερο θεμελιώδες μέγεθος από το οποίο εξαρτάται οποιαδήποτε ραδιοεπικοινωνία. Με το πέρασμα του χρόνου μάλιστα, η χρήση του αυξήθηκε και εξακολουθεί να αυξάνεται με γοργούς ρυθμούς, με αποτέλεσμα στη σύγχρονη εποχή, το φάσμα να τίθεται σε μια πληθώρα από διαφορετικές χρήσεις σε στεριά, θάλασσα, αέρα και διάστημα. Αυτές οι χρήσεις από τη μία πλευρά επιφέρουν τεράστια οφέλη στην κοινωνία μας, αλλά ταυτόχρονα καθιστούν



ζωτικής σημασίας τη χρησιμοποίηση των κατάλληλων μηχανισμών διαχείρισης του πολύτιμου αυτού αγαθού. Στην πραγματικότητα και οι πάροχοι των υπηρεσιών που χρησιμοποιούν το φάσμα ενδιαφέρονται για τη σωστή διαχείρισή του, καθότι επιθυμούν να μεγιστοποιήσουν την αποδοτικότητα στη χρήση του φάσματος που διαθέτουν, ώστε να εξυπηρετήσουν ταυτόχρονα όσο το δυνατόν περισσότερους χρήστες και άρα να αυξήσουν τα έσοδά τους.

Προς το παρόν το φάσμα κατανέμεται σε εθνικό επίπεδο με τη μέθοδο της παροχής στατικών / μόνιμων αδειών για χρήση συγκεκριμένου εύρους του σε κάθε RAT, ενώ υπάρχουν περαιτέρω υποδιαιρέσεις του, ώστε κάθε πάροχος να έχει υπό την κατοχή του ένα μόνο κομμάτι που χρειάζεται για την παροχή συγκεκριμένων υπηρεσιών, μέχρις ότου λήξει η άδεια που κατέχει. Αυτά τα κομμάτια φάσματος έχουν προκαθορισμένο και σταθερό εύρος και απέχουν μεταξύ τους απόσταση (στην κλίμακα των συχνοτήτων) ικανή ώστε να αποφεύγονται οι ανεπιθύμητες παρεμβολές. Η απόσταση αυτή απαντάται στη βιβλιογραφία και με τον όρο «εύρος ζώνης – φρουρός». Αυτή η μέθοδος ανάθεσης φάσματος αποτελεί την ιδανική λύση και για εύκολη κατασκευή συσκευών που λειτουργούν σε συγκεκριμένες συχνότητες, οπότε σε γενικές γραμμές έχει αποδειχθεί η πλέον αποτελεσματική εδώ και χρόνια.

Ωστόσο, ουδέν καλόν αμιγές κακού! Η μέθοδος στατικής ανάθεσης φάσματος διαθέτει αρκετά μειονεκτήματα. Κατ' αρχάς οι βιομηχανίες ασυρμάτων συσκευών και επικοινωνιών συγκλίνουν συνεχώς μεταξύ τους και οι προσφερόμενες από αυτές υπηρεσίες και εφαρμογές ελάχιστα διαφέρουν μεταξύ τους. Έτσι, οι μέχρι πρότινος υπάρχοντες μηχανισμοί διαίρεσης του φάσματος αναλόγως της υπηρεσίας παύουν να είναι αποτελεσματικοί. Ακόμα μεγαλύτερη σύγκλιση αναμένεται να επέλθει τα επόμενα χρόνια, με τη συνένωση πολλών RATs διαφορετικής φύσεως σε κοινά δίκτυα – κορμούς, όπως επιτάσσουν οι σύγχρονες επικοινωνίες. Συγκεκριμένα, η είσοδος των cognitive δικτύων ήδη προβλέπει συγκεκριμένα σενάρια ευέλικτης χρήσης του διαθέσιμου φάσματος, όπως θα αναφερθεί στο κεφάλαιο 6. Εκτός των άλλων, η στατική παροχή αδειών σε συγκεκριμένα RATs παρεμποδίζει την ανάπτυξη νέων τεχνολογιών, καθώς επιβραδύνει τον κύκλο κάθε τεχνολογίας στην αγορά.

Επιπλέον, αξιοσημείωτο ζήτημα αποτελεί το γεγονός ότι τα περισσότερα δίκτυα επικοινωνιών υπόκεινται σε χρονικές και τοπικές μεταβολές της απαιτούμενης κίνησης, με αποτέλεσμα να μεταβάλλεται ανάλογα και το ποσοστό του φάσματος που χρησιμοποιείται. Έτσι, ένα κομμάτι του φάσματος που χρησιμοποιείται για μια υπηρεσία παραμένει αχρησιμοποίητο σε συγκεκριμένες χρονικές στιγμές και σε ορισμένες περιοχές, ενώ αντίστοιχα άλλες υπηρεσίες στερούνται του απαιτούμενου φάσματος στις ίδιες χρονικές στιγμές και περιοχές. Δεδομένης μάλιστα της οικονομικής αξίας που έχει αποκτήσει το φάσμα και της σημασίας της αποδοτικής χρήσης του, πρέπει ξεκάθαρα να αποφεύγεται οποιαδήποτε σπατάλη του.

Από τα παραπάνω γίνεται σαφής η ύπαρξη κινήτρων για τη μελέτη αποδοτικότερων σχημάτων ανάθεσης φάσματος και η πίεση προς την έρευνα για μεγαλύτερη αποδοτικότητα στη χρήση του φάσματος και μεγαλύτερη ευελιξία. Αυτά βέβαια απαιτούν νέες τεχνολογίες αναφορικά με κεραιές, εναέριες διεπαφές (air interfaces) και τοπολογίες δικτύου. Σε γενικές γραμμές, πιστεύεται ότι θα μπορούσε να βελτιωθεί η επίδοση των δικτύων, να αυξηθούν τα επίπεδα ποιότητας και να μειωθεί το κόστος προς το χρήστη, αν δινόταν η δυνατότητα στους NOs, να αποφασίσουν αυτοί για το πως θα αξιοποιήσουν το φάσμα. Με άλλα λόγια θα ήταν αποδοτικότερο το να διαλέξουν οι ίδιοι οι NOs το πώς θα αναθέσουν το φάσμα στα συστήματά τους, ώστε να αποφευχθεί το να υπάρχουν τμήματα φάσματος τα οποία μένουν αχρησιμοποίητα, ενώ θα μπορούσαν να αξιοποιηθούν σε συστήματα τα οποία παρουσιάζουν υψηλή χρησιμοποίηση. Η δυνατότητα δυναμικής ανάθεσης του φάσματος που αποτελεί και το γενικό γνωστικό αντικείμενο μέσα στο οποίο κινείται η διατριβή, αποτελεί σήμερα έναν από τους πλέον σύγχρονους τομείς έρευνας στον τομέα των ασυρμάτων επικοινωνιών. Οι μέθοδοι, οι προϋποθέσεις, οι απαιτήσεις και τα αποτελέσματα μιας τέτοιας προσέγγισης εγείρουν το ενδιαφέρον της διεθνούς επιστημονικής κοινότητας, με ρυθμό αυξανόμενο μάλιστα, ειδικά με τη συνεχή έλευση περισσότερο εξελιγμένων συστημάτων.

## 2.7 DSNPM

Μέχρι στιγμής έχει παρουσιαστεί το πόσο σημαντικό και πολύπλοκο γεγονός αποτελεί η χρήση ετερογενών δικτύων και κατ' επέκταση η διαχείριση του συνολικού φάσματος στα B3G δίκτυα. Για αυτό κρίνεται απαραίτητη η αναφορά στο Dynamic Self-organizing Network Planning and Management (DSNPM) δεδομένου ότι είναι ο μηχανισμός ο οποίος έχει ως στόχο την λήψη μεσοπρόθεσμων και μακροπρόθεσμων αποφάσεων μέσω τις συλλογής δεδομένων περί τις κατάστασης του δικτύου, οι οποίες αφορούν την δυναμική αναδιάρθρωση η οποία αναφέρθηκε νωρίτερα. Οι αποφάσεις αυτές προκύπτουν βάσει ανάλυσης των συλλεγόμενων δεδομένων τα οποία αποτελούν και τα δεδομένα εισόδου. Μετά την ανάλυση τους προκύπτει γνώση η οποία αξιοποιείται για την παραμετροποίηση του δικτύου, γεγονός το οποίο με την σειρά του θα παράγει καινούργια δεδομένα εισόδου.

Για να παρουσιαστεί πιο λεπτομερώς το DSNPM, κρίνεται σκόπιμη η αναφορά στις απαιτήσεις που καλείται να καλύψει. Κάποιες από αυτές είναι η παραμετροποίηση όσο αφορά την υποστήριξη διαφόρων κλάσεων τελικών χρηστών, η αποδοτικότητα στην διαχείριση πολλαπλών και δυναμικά μεταβαλλόμενων καταστάσεων όσο αφορά την υποδομή του δικτύου, η βελτιστοποίηση της σύνδεσης των τερματικών όσο αφορά το QoS αλλά και το κόστος, η περιαγωγή των χρηστών εντός του δικτύου δίνοντας έμφαση στην απρόσληπτη και απροβλημάτιστη σύνδεση, η συνεργασία των διαφορετικών RATs και η επεκτασιμότητα ώστε να μπορεί να υπάρχει άμεση ανταπόκριση στις αλλαγές του περιβάλλοντος.

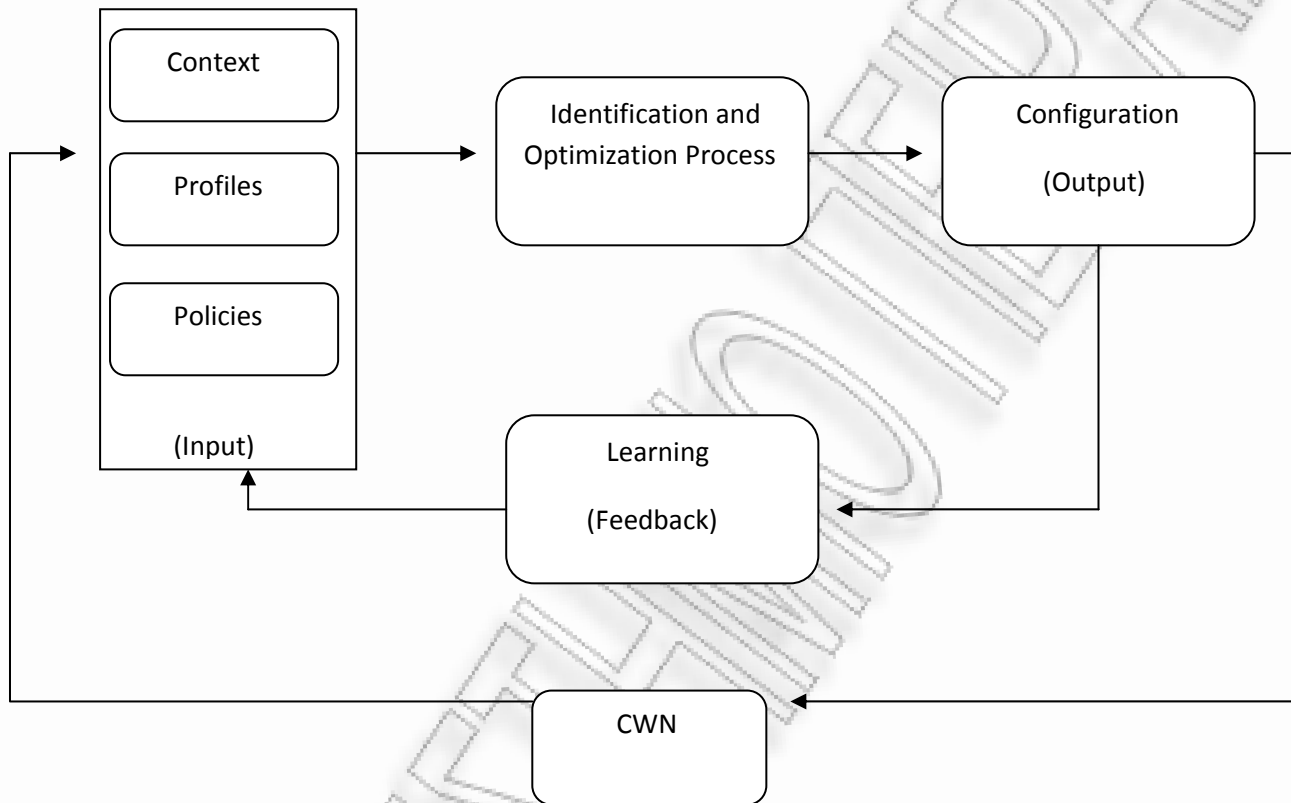
Για να καλυφθούν οι ανωτέρω απαιτήσεις παρέχονται ένα σύνολο λειτουργιών οι οποίες υποστηρίζουν την απόκτηση των παραμέτρων του περιβάλλοντος έτσι ώστε να υποστηρίζεται επίγνωση του περιβάλλοντος, η διαχείριση προφίλ ώστε να υποστηρίζεται η εξατομίκευση, οι πολιτικές για την παροχή κανόνων που είναι απαραίτητοι για την βέλτιστη συνδεσιμότητα, η άρση αποφάσεων που αφορούν την βέλτιστη συνδεσιμότητα, η συνεργασία μεταξύ των διαφορετικών τεχνολογιών για την παροχή διάχυτης υπολογιστικής

και η απόκτηση της γνώσης βασιζόμενη στην διαδικασία της εκπαίδευσης, κάτι πολύ σημαντικό για την επέκταση και την πολυπλοκότητα.

Αναλυτικά ο τρόπος λειτουργίας του περιγράφεται ακολούθως. Το DSNPM πραγματοποιεί την συλλογή πληροφοριών μέσω της λειτουργίας των παραμέτρων του περιβάλλοντος που αφορούν τα στοιχεία του CWN όπως και την κατάσταση του περιβάλλοντος. Επιπρόσθετα μέσω των λειτουργιών παρακολούθησης παρέχονται πληροφορίες για κάθε τερματικό και για συγκεκριμένη χρονική περίοδο όσο αφορά τις απαιτήσεις δικτύου, τις συνθήκες φορητότητας και το απαιτούμενο QoS επίπεδο. Μέσω αυτών των πληροφοριών αναγνωρίζονται τυχόν προβληματικές καταστάσεις λειτουργίας του δικτύου αλλά ενημερώνονται και οι δείκτες απόδοσης δικτύου. Ακόμα τα προφίλ των τερματικών παρέχουν πληροφόρηση σύμφωνα τις δυνατότητες, προτιμήσεις, απαιτήσεις και περιορισμούς των τερματικών αλλά και τις δυνατότητες του δικτύου. Αφού καταγραφεί τι εφαρμογές χρησιμοποιούνται από τα τερματικά, κάτι το οποίο αποτελεί έναν από τους βασικούς παράγοντες για την παραμετροποίηση του τρόπου σύνδεσης του τερματικού στο δίκτυο ανακτούνται οι πολιτικές δεδομένου ότι πέραν των απαιτήσεων των τερματικών, όποια αναδιάρθρωση του δικτύου οφείλει να είναι βιώσιμη και σύμφωνη με τις αρχές λειτουργίας του παρόχου.

Έχοντας όλα τα παραπάνω ως δεδομένα εισόδου ξεκινάει η διαδικασία βελτιστοποίησης του δικτύου. Μια από τις πρακτικές προσέγγισης για την επίτευξη του στόχου αυτού είναι ο εντοπισμός της αναδιάρθρωσης η οποία θα αποφέρει την βελτιστοποίηση δίνοντας προτεραιότητα στην ικανοποίηση του τελικού χρήστη παράλληλα με το κόστος της αναδιάρθρωσης αλλά και το κόστος του QoS.

Από την εκτέλεση των αλγορίθμων που περιλαμβάνει το DSNPM προκύπτει γνώση και εμπειρία όσο αφορά το δίκτυο. Μέσω αυτής της γνώσης το περιβάλλον μπορεί να χαρακτηριστεί με ένα μοναδικό τρόπο. Ομοίως χαρακτηρίζεται και η προτεινόμενη αναδιάρθρωση. Με την διατήρηση ενός ιστορικού προτεινόμενων αναδιαρθρώσεων παρέχεται η δυνατότητα της σύγκρισης της οιαδήποτε τρέχουσας κατάστασης με αυτές στο ιστορικό ώστε όχι μόνο να δοθεί μια λύση αλλά να υπάρχει η εκ των προτέρων επιβεβαίωση ότι η συγκεκριμένη λύση θα δώσει τα επιθυμητά αποτελέσματα.



Εικόνα 2.7: Οπτική αναπαράσταση του DSNPM

## 2.8 Αξιολόγηση δικτύων B3G

Με γνώμονα τα παραπάνω, εύλογα τίθεται το ακόλουθο ερώτημα:

«Πώς μπορούμε να γνωρίζουμε ότι ένα σύνθετο περιβάλλον επικοινωνιών (απαρτιζόμενο από πολλές τεχνολογίες) χρήζει κάποιου είδους προσαρμογής στις εξωτερικές συνθήκες»;

Με άλλα λόγια, οι διαχειριστές των δικτύων επιβάλλεται να δύνανται να κρίνουν πότε ένα τμήμα δικτύου τους αντιμετωπίζει κάποιο πρόβλημα. Και στην περίπτωση απλών τεχνολογιών RATs (πχ ένα δίκτυο UMTS, ένα δίκτυο GSM, κλπ.), κάτι τέτοιο είναι σαφώς

ευκολότερο. Αντίθετα, στην περίπτωση σύνθετων τοπολογιών που αποτελούνται από πληθώρα ετερογενών τεχνολογιών, η συγκεκριμένη εργασία καθίσταται δυσκολότερη.

Για αυτούς τους λόγους, συνηθίζεται η παρακολούθηση των τιμών κάποιων παραμέτρων, οι οποίες καλούνται Key Performance Indicators (KPIs). Οι παράμετροι αυτές είναι ενδεικτικές και σε καμία περίπτωση δεν αποτελούν σαφές δείγμα της ορθής ή μη λειτουργίας ενός τμήματος δικτύου. Ωστόσο, μπορούν να συντελέσουν σημαντικά στη σωστή λήψη αποφάσεων περί της αναγκαιότητας αλλαγής (αναδιάρθρωσης) κάποιων εκ των λειτουργικών χαρακτηριστικών (συχνότητες, τεχνολογίες ισχείς, κλπ), ώστε να διορθωθούν τυχόν προβλήματα και να καταστεί δυνατή η βέλτιστη εξυπηρέτηση των χρηστών.

Κατ' αναλογία με τα παραπάνω, το επόμενο κεφάλαιο αναπτύσσει την έννοια της οντολογίας, η οποία αποτελεί περιγραφή με προκαθορισμένη μορφή του τρόπου με τον οποίο συσχετίζονται διάφορες οντότητες, βοηθώντας έτσι στην επικοινωνία και ανταλλαγή πληροφοριών μεταξύ ετερογενών συστατικών (τεχνολογιών) και άρα αποτελώντας θεμελιώδες στοιχείο της λειτουργίας συστημάτων (δικτύων) B3G.

## 3. Οντολογίες

### 3.1 Εισαγωγή – Τι είναι «οντολογία»

Η λέξη "οντολογία" είναι ένας όρος δανεισμένος από τη φιλοσοφία, η οποία αναφέρεται στην επιστήμη της περιγραφής των διαφόρων ειδών οντοτήτων στον κόσμο και του τρόπου που αυτές συσχετίζονται. Συχνά, συγχέεται με την επιστημολογία που αφορά τη γνώση, ενώ έχει δημιουργήσει πολλές διαμάχες στις συζητήσεις για την τεχνητή νοημοσύνη.

Στα πλαίσια της κατανεμημένης γνώσης, οντολογία είναι ο προσδιορισμός μιας σύλληψης, δηλαδή, μια περιγραφή των εννοιών και των σχέσεων που μπορούν να υπάρξουν για έναν πράκτορα λογισμικού ή μια κοινότητα τέτοιων πρακτόρων. Είναι μια αναγνώσιμη από τον υπολογιστή περιγραφή της γνώσης των πόρων στο δίκτυο. Αυτός ο ορισμός είναι σύμφωνος με τη χρήση της οντολογίας ως σύνολο ορισμών εννοιών, αλλά γενικότερος. Λέμε ότι ένας πράκτορας λογισμικού "δεσμεύεται" σε μια οντολογία εάν οι ενέργειές του είναι σύμφωνες με τους ορισμούς στην οντολογία. Πρακτικά, μια κοινή οντολογία καθορίζει το λεξιλόγιο με το οποίο ερωτήσεις και ισχυρισμοί ανταλλάσσονται μεταξύ των πρακτόρων λογισμικού. Οι οντολογικές δεσμεύσεις είναι συμφωνίες χρησιμοποίησης του κοινού λεξιλογίου κατά τρόπο κατανοητό και συνεπή. Οι πράκτορες λογισμικού που μοιράζονται ένα λεξιλόγιο δεν χρειάζονται να μοιράζονται μια βάση γνώσεων και καθένας από αυτούς δεν απαιτείται να απαντήσει σε όλες τις ερωτήσεις που μπορούν να διατυπωθούν στο κοινό λεξιλόγιο. Με τις οντολογίες οι πράκτορες λογισμικού γίνονται ευφυείς επειδή μπορούν να χρησιμοποιήσουν τη γνώση που περιλαμβάνεται στην οντολογία στο στάδιο της διαπραγμάτευσης και της λήψης αποφάσεων.

Στην επιστήμη των πληροφοριών, μια οντολογία καθορίζει τους όρους που χρησιμοποιούνται για να περιγράψουν και να αναπαραστήσουν έναν τομέα της γνώσης, με

σκοπό τη διευκόλυνση της διανομής και της επαναχρησιμοποίησης της γνώσης αυτής. Μια οντολογία είναι το προϊόν μιας προσπάθειας να διατυπωθεί ένα εξαντλητικό και αυστηρό εννοιολογικό σχήμα για μια περιοχή. Είναι μία ιεραρχική δομή δεδομένων που περιέχει όλες τις σχετικές οντότητες, τις μεταξύ τους σχέσεις και κανόνες εξαγωγής συμπερασμάτων μέσα σε εκείνη την περιοχή. Οι οντολογίες χρησιμοποιούνται συνήθως στην τεχνητή νοημοσύνη και στην αναπαράσταση γνώσης. Τα προγράμματα υπολογιστών μπορούν να χρησιμοποιήσουν μια οντολογία για ποικίλους λόγους συμπεριλαμβανομένου του επαγωγικού συλλογισμού, της ταξινόμησης, των ποικίλων τεχνικών επίλυσης προβλημάτων, καθώς επίσης και για να διευκολύνουν την επικοινωνία και τη διανομή των πληροφοριών μεταξύ των διαφορετικών συστημάτων. Χρησιμοποιούνται από τους ανθρώπους, τις βάσεις δεδομένων, και τις εφαρμογές που πρέπει να μοιραστούν τις πληροφορίες περιοχών (μια περιοχή είναι ακριβώς μία συγκεκριμένη θεματική περιοχή ή τομέας της γνώσης, όπως η ιατρική, η κατασκευή εργαλείων, η ακίνητη περιουσία, η γεωγραφία, η οικονομική διαχείριση, κ.λπ.). Οι οντολογίες περιλαμβάνουν χρησιμοποιούμενους από υπολογιστές ορισμούς βασικών εννοιών μίας θεματικής περιοχής και τις σχέσεις μεταξύ τους. Κωδικοποιούν τη γνώση σε μια περιοχή, καθώς και τη γνώση από περισσότερες από μία περιοχές. Έτσι, καθιστούν την γνώση αυτή επαναχρησιμοποιήσιμη.

Η οντολογική ανάλυση αποσαφηνίζει τη δομή της γνώσης. Δεδομένης μιας περιοχής, η οντολογία της διαμορφώνει τον πυρήνα οποιουδήποτε συστήματος αναπαράστασης γνώσης για αυτή την περιοχή. Χωρίς τις οντολογίες, ή τις συλλήψεις που κρύβονται κάτω από τη γνώση, δεν μπορεί να υπάρξει ένα λεξιλόγιο για την αναπαράσταση της γνώσης.

## 3.2 Παραδείγματα οντολογιών

### 3.2.1 Οντολογία και φιλοσοφία

Η οντολογία ως κλάδος της φιλοσοφίας είναι η επιστήμη που εξετάζει την υπόσταση των διαφόρων ειδών και δομών των αντικειμένων, των ιδιοτήτων τους καθώς και των σχέσεων



τους σε κάθε τομέα της πραγματικότητας. Ως όρος, προερχόμενος από την ελληνική γλώσσα, δημιουργήθηκε το 1613 ανεξάρτητα από δύο φιλοσόφους, τους Rudolf Gockel {Lexicon philosophicum} και Jacob Lorhard {Theatrum philosophicum}, ενώ στη φιλοσοφία συνήθως αποτελεί τη μελέτη της υπόστασης των διάφορων ειδών και δομών των αντικειμένων, των ιδιοτήτων τους καθώς και των σχέσεων τους σε κάθε τομέα της πραγματικότητας. Η έννοια της οντολογίας χρησιμοποιείται συχνά από τους φιλοσόφους ως συνώνυμο των μεταφυσικών, ενός όρου που χρησιμοποιήθηκε από τους πρώτους μαθητές του Αριστοτέλη για να αναφερθούν σε αυτό που ο ίδιος ο Αριστοτέλης αποκαλούσε Πρώτη Φιλοσοφία. Η οντολογία ασχολείται με το τι υπάρχει ή καλύτερα την έννοια της ύπαρξης, ενώ ορισμένες φορές κατά μια γενικότερη έννοια πραγματεύεται τη μελέτη του τι μπορεί να υπάρχει. Στη φιλοσοφία έχει τουλάχιστον τέσσερις καθιερωμένες έννοιες.

Την οντολογία υπό την καθαρή φιλοσοφική έννοια (που ασχολείται με το γιατί υπάρχει κάτι από το να μην υπάρχει τίποτα), την οντολογία ως τομέας γνώσεως που ασχολείται με τα φιλοσοφικά προβλήματα των οντοτήτων, την οντολογία ως κλάδο που πραγματεύεται τη μελέτη του αποτελέσματος της οντολογίας ως τομέα γνώσεως και, τέλος, την εφαρμοσμένη επιστημονική οντολογία που επικεντρώνεται στη μελέτη ενός συγκεκριμένου επιστημονικού τομέα γνώσεως που εξετάζει τι υπάρχει στην πραγματικότητα. Ο τελευταίος τομέας χωρίζεται στην οντολογία ως θεωρητικό κλάδο που παρέχει μια περιγραφή των πραγμάτων που υποτίθεται ότι υφίστανται σύμφωνα με μια ιδιαίτερη θεωρία, άλλα που δεν είναι αναγκαίο να είναι αληθινά και στην οντολογία, ως υφιστάμενο κλάδο που ασχολείται με τον πραγματικό κόσμο όλων των υπαρχουσών οντοτήτων που προσδιορίζονται από μια τελείως πλήρη εφαρμοσμένη οντολογική θεωρία.

### **3.2.2 Οντολογία και πληροφορική**

Η οντολογία στην πληροφορική έχει μια πάρα πολύ πρακτική σημασία. Το λεξικό του Webster ορίζει την οντολογία στην πληροφορική ως:

«μια κατηγοριοποίηση όλων των σημαντικών κατηγοριών αντικείμενων ή εννοιών που υπάρχουν σε κάποιο τομέα γνώσης συμπεριλαμβάνοντας και τις αναμεταξύ τους σχέσεις. Όταν είναι πλήρης η οντολογία αποτελεί την κατηγοριοποίηση όλων των εννοιών σε κάποιο τομέα γνώσης, συμπεριλαμβανομένων των αντικειμένων και των ιδιοτήτων τους, καθώς και των λειτουργιών που απαιτούνται για να τα καθορίσουν πλήρως. Μια απλουστευμένη οντολογία μπορεί να περιέχει μόνο μια ιεραρχική ταξινόμηση που παρουσιάζει τις σχέσεις μεταξύ των εννοιών που περιέχει. Μια οντολογία μπορεί να απεικονιστεί ως μια θεωρητική γραφική παράσταση με τους κόμβους και τα τόξα που αντιπροσωπεύουν τα αντικείμενα και τις σχέσεις».

Σύμφωνα με τον ορισμό του Webster η οντολογία είναι μια συστηματική κατανομή, κατηγοριοποίηση των εννοιών, ιεραρχική ταξινόμηση κτλ., ενώ σύμφωνα με τον πλέον διαδεδομένο ορισμό του Gruber η οντολογία είναι ένας ρητός και τυπικός ορισμός μιας εννοιολογικής σύλληψης. Έτσι λοιπόν, στον τομέα της Τεχνητής Νοημοσύνης αλλά και της (βιο)πληροφορικής, η οντολογία παίζει έναν πολύ σημαντικό ρόλο εξυπηρετώντας τις τεχνολογικές ανάγκες εκμετάλλευσης της πληροφορίας. Συνεπώς, συχνά αναφερόμαστε σε μια ορολογία δανεισμένη από τη φιλοσοφία που δεν αποτελεί μια θεωρία όπως στη φιλοσοφία, άλλα έναν ορισμό μιας εννοιολογικής σύλληψης που αποσκοπεί στην αντιπροσώπευση της πληροφορίας. Αυτό όμως που γίνεται συνεχώς όχι μόνο επιθυμητό, άλλα και απολύτως αναγκαίο κατά τον σχεδιασμό της οντολογίας, είναι η ανάγκη εκμετάλλευσης βασικών αρχών προσαρμοσμένων από την εφαρμοσμένη επιστημονική οντολογία. Αυτές οι φιλοσοφικές αρχές θα μας βοηθήσουν στην οργάνωση και εκμετάλλευση της γνώσης.

### **3.2.3 Οντολογία και γενετική**

Λαμβάνοντας υπόψη την ιστορία της έρευνας στον τομέα της γενετικής των τελευταίων 50 ετών, η συμβολή της φαινοτυπικής ανάλυσης των γενετικών μεταλλάξεων έπαιξε ρόλο ζωτικής σημασίας στην κατανόηση της ομαλής λειτουργίας των γονιδίων καθώς οι

φαινοτυπικές αναλύσεις μιας μετάλλαξης, αλλά και των συνδυασμών τους, προσδιόρισαν όχι μόνο τον βιολογικό σκοπό των μεμονωμένων γονιδίων αλλά και τη λειτουργία σύνθετων βιολογικών διαδικασιών. Εντούτοις, έχουν υπάρξει, με ελάχιστες εξαιρέσεις, πολύ λίγες οργανωμένες προσπάθειες συστηματοποίησης της περιγραφής των φαινοτυπικών πληροφοριών, ακόμη και σε πολύ καλά μελετημένους οργανισμούς, όπως π.χ. τη δροσόφylla.

Οι οντολογίες αποτελούν σημαντικό μέσο για την απεικόνιση Βιολογικών πληροφοριών από τον καιρό του Linnaeus και με τη σχετικά πρόσφατη εμφάνιση της οντολογίας των γονιδίων (Gene Ontology GO) το 2000. Αυτές οι τεχνικές αυστηρής διευκρίνισης των σημασιολογικών σχέσεων μεταξύ ορισμών έχουν γίνει πρότυπα που χρησιμοποιούνται για τη υποστήριξη της αντιπροσώπευσης της γνώσης στον τομέα της γενετικής. Οι ιεραρχικές οντολογίες φυλάσσουν τις πληροφορίες για τη δομή μιας ιδιαίτερης γνωστικής περιοχής με ποικίλους βαθμούς λεπτομέρειας, επιτρέποντας κατά συνέπεια, την ενσωμάτωση των εννοιών και των περιγραφών σε διαφορετικά επίπεδα διάκρισης. Αυτή η προσέγγιση αποτελεί τη βάση νέων μεθόδων υπολογιστικής ανάλυσης Βιολογικών πληροφοριών. Μια δικαιολογημένη ερώτηση είναι γιατί να χρησιμοποιήσουμε μια οντολογία και όχι μια σειρά από αποσυνδεδεμένους, τυποποιημένους ορισμούς, όπως αυτοί παρέχονται από ένα ελεγχόμενο λεξιλόγιο; Τα πλεονεκτήματα της χρήσης οντολογιών έχουν υποστηριχτεί εκτενώς, αλλά ο κύριος λόγος είναι ότι οι οντολογίες προσπαθούν να συλλάβουν την ακριβή έννοια των όρων, με τρόπο που μπορεί να είναι υπολογιστικά εκμεταλλεύσιμος και συμπεριλαμβάνει την αυτοματοποιημένη λογική συναγωγή συμπερασμάτων, την τυπική συλλογιστική, τον έλεγχο συνέπειας κτλ. Από τη σκοπιά της φαινοτυπικής γνώσης, ο σημαντικότερος παράγοντας είναι η ανάγκη συνδυασμού πληροφοριών από διαφορετικές φαινοτυπικές επισημειώσεις ή από διαφορετικά πρωτόκολλα (Βιολογικές δοκιμασίες), καθώς επίσης και τις σχέσεις μεταξύ τους.

### 3.3 Μέθοδοι ανάπτυξης

#### 3.3.1 Γενική μεθοδολογία ανάπτυξης οντολογίας

Οι οντολογίες εκφράζονται συνήθως σε μια βασισμένη στη λογική γλώσσα, έτσι ώστε λεπτομερείς, ακριβείς, συνεπείς και σημαντικές διακρίσεις να μπορούν να γίνουν μεταξύ των κλάσεων, των ιδιοτήτων, και των σχέσεων. Μερικά εργαλεία οντολογίας μπορούν να εκτελέσουν αυτοματοποιημένο συλλογισμό χρησιμοποιώντας τις οντολογίες, και να παρέχουν έτσι προηγμένες υπηρεσίες σε ευφυείς εφαρμογές όπως: σημασιολογική αναζήτηση και ανάκτηση, πράκτορες λογισμικού, υποστήριξη αποφάσεων, ομιλία και φυσική κατανόηση γλώσσας, διαχείριση γνώσης, ευφυείς βάσεις δεδομένων, και ηλεκτρονικό εμπόριο.

Οι οντολογίες μπορούν να αποδειχθούν πολύ χρήσιμες για μια κοινότητα ως τρόπος δόμησης και ορισμού της έννοιας των μετα-δεδομένων που συλλέγονται αυτήν την περίοδο και τυποποιούνται. Χρησιμοποιώντας τις οντολογίες, οι αυριανές εφαρμογές μπορούν να είναι “ευφυείς”, υπό την έννοια ότι μπορούν να λειτουργήσουν ακριβέστερα στο ανθρώπινο εννοιολογικό επίπεδο.

Αν θέλαμε να βρούμε μερικά ακόμα πεδία εφαρμογής των οντολογιών, ερχόμενοι στα δίκτυα επικοινωνιών και υπολογιστών, θα λέγαμε ότι οι οντολογίες μπορούν να ενισχύσουν τη λειτουργία του Web από πολλές απόψεις. Μπορούν να χρησιμοποιηθούν για να βελτιώσουν την ακρίβεια των Web αναζητήσεων - το πρόγραμμα αναζήτησης μπορεί να ψάχνει μόνο εκείνες τις σελίδες που αναφέρονται σε μια ακριβή έννοια αντί όλων των άλλων εννοιών των διφορούμενων λέξεων κλειδιών. Οι πιο προηγμένες εφαρμογές μπορούν να χρησιμοποιήσουν τις οντολογίες για να συσχετίσουν τις πληροφορίες μιας σελίδας με τις σχετικές δομές γνώσης και κανόνες εξαγωγής συμπερασμάτων. Επιπλέον, καθιστούν πολύ ευκολότερη την ανάπτυξη προγραμμάτων που μπορούν να αντιμετωπίσουν περίπλοκες ερωτήσεις, των οποίων οι απαντήσεις δεν βρίσκονται σε μία ενιαία ιστοσελίδα.

Ακόμη, οι οντολογίες μπορούν να χρησιμοποιηθούν για να υποστηρίξουν συστήματα τεχνητής νοημοσύνης (AI), με την παροχή μιας βαθύτερης και σθεναρότερης αντιπροσώπευσης της περιοχής στην οποία κάποιος επιθυμεί να εξάγει συλλογισμούς και να λύσει προβλήματα. Σε αυτές εφαρμόζεται, αυστηρά, ο παραγωγικός συλλογισμός και όχι λογική ασαφής ή βασισμένη σε πιθανότητες, ή οποιοσδήποτε συλλογισμός που προσπαθεί να μιμηθεί τη συνείδηση.

Μια οντολογία διαφέρει από ένα σχήμα XML διότι είναι μια αναπαράσταση γνώσης, όχι ένα σχήμα μηνυμάτων. Τα περισσότερα βασισμένα στη βιομηχανία πρότυπα Ιστού αποτελούνται από έναν συνδυασμό σχημάτων μηνυμάτων και προδιαγραφών πρωτοκόλλου. Σε αυτά τα σχήματα έχει δοθεί μια λειτουργική σημασιολογία αλλά η προδιαγραφή δεν έχει ως σκοπό να υποστηρίξει το συλλογισμό έξω από τα πλαίσια αυτής της λειτουργίας.

Οι οντολογίες είναι κρίσιμες για τις εφαρμογές που θέλουν να αναζητήσουν ή να συγχωνεύσουν πληροφορίες από διαφορετικές (γνωστικές) περιοχές. Μια οντολογία που είναι συνδεδεμένη με μια ιδιαίτερη περιοχή ονομάζεται domain ontology. Η αποκαλούμενη foundation ontology ή upper ontology είναι ένα είδος οντολογίας που προσπαθεί να περιγράψει γενικές οντότητες και δεν είναι τόσο συγκεκριμένη, οπότε είναι γενικά πιο εφαρμόσιμη. Περιέχει ένα βασικό γλωσσάριο, με τους όρους του οποίου οτιδήποτε άλλο σε μια ευρεία περιοχή μπορεί και πρέπει να περιγραφεί.

Αναφορικά με τα χαρακτηριστικά μιας οντολογίας, ορίζεται ένα λεξιλόγιο (vocabulary) από έννοιες που περιγράφουν τον κόσμο μας. Εν συνεχεία, καθορίζονται κλάσεις και υποκλάσεις, καθεμιά από τις οποίες αναπαριστά μια έννοια και έχει ιδιαίτερα χαρακτηριστικά, αλλά και οντότητες που ανήκουν σε αυτή, με συγκεκριμένη ιεραρχία. Οι κλάσεις αυτές συσχετίζονται μεταξύ τους και μάλιστα οντότητες της μιας κλάσης συσχετίζονται με οντότητες κάποιας άλλης κλάσης. Παράλληλα, παράγονται τα λεγόμενα αξιώματα, ήτοι κανόνες που διέπουν τις έννοιες και τις σχέσεις μεταξύ τους. Επιπλέον, μπορεί να περιέχονται σε μια οντολογία ιδιότητες, περιορισμοί τιμών, δηλώσεις ασυμβατότητας και προσδιορισμός λογικών σχέσεων μεταξύ των αντικειμένων.

Έτσι, σε γενικές γραμμές, μια οντολογία καθορίζει ένα κοινό λεξιλόγιο και μια κοινή κατανόηση για ένα θέμα. Διασφαλίζει την επικοινωνία μεταξύ κάποιων οντοτήτων που καθοδηγούν την ανταλλαγή «μετα-δεδομένων», τα οποία στην πληροφορική καλούνται «πράκτορες λογισμικού» (agents). Τα δεδομένα αυτά αναπαριστούν σαφώς τη σημασιολογία των δεδομένων με τρόπο κατανοητό από τους υπολογιστές. Βέβαια, θα έλεγε κανείς έτσι ότι οι οντολογίες διαθέτουν πολλά κοινά στοιχεία με τις βάσεις δεδομένων. Αυτό σε ένα βαθμό είναι αλήθεια, όμως μια βάση δεδομένων είναι ένα σύνολο από πίνακες και τις μεταξύ τους σχέσεις. Αντίθετα, μια οντολογία περιέχει συντακτικά και σημασιολογικά πλουσιότερη πληροφορία από μια βάση δεδομένων. Επιπλέον, η πληροφορία που περιγράφεται από μια οντολογία περιέχει ημιδομημένο κείμενο σε φυσική γλώσσα και όχι πληροφορία ενός σχεσιακού πίνακα. Τέλος, σε αντίθεση με μια βάση δεδομένων, μια οντολογία πρέπει να είναι δικτυακής αρχιτεκτονικής, διότι χρησιμοποιείται για το διαμοιρασμό της πληροφορίας.

Για την ανάπτυξη μιας οντολογίας αρχικά πρέπει να βρεθεί ο σκοπός που θα εξυπηρετήσει. Εν συνεχεία, πρέπει να συλληφθούν οι έννοιες της οντολογίας και οι μεταξύ τους σχέσεις, αλλά και οι όροι με τους οποίους θα αναφέρονται οι έννοιες και οι σχέσεις. Κατόπιν θα οριστούν διάφορα αξιώματα και στιγμιότυπα των εννοιών και όλα τα παραπάνω θα κωδικοποιηθούν σε μια συγκεκριμένη γλώσσα. Τέλος, θα αξιολογηθούν και θα τεκμηριωθούν.

Πολυάριθμες είναι οι περιοχές εφαρμογών μιας οντολογίας, όπως τα συστήματα διαχείρισης γνώσης, η μοντελοποίηση της γνώσης, η μετάφραση της πληροφορίας, το ηλεκτρονικό εμπόριο, η ανάκτηση γνώσης και πληροφορίας, η επεξεργασία και η κατανόηση μιας φυσικής γλώσσας, οι υπηρεσίες που προσφέρει ο σημασιολογικός ιστός, πώς μπαίνει η σημασιολογία στις ιστοσελίδες και γενικά πώς μεταδίδεται η εκεί ευρισκόμενη πληροφορία.

### 3.3.2 Παράδειγμα - η γλώσσα οντολογίας ιστού OWL

#### 3.3.2.1 Γενικά στοιχεία

Η OWL δημιουργήθηκε για να ικανοποιήσει την ανάγκη για μια γλώσσα οντολογίας Ιστού. Η OWL έχει σχεδιαστεί για τις εφαρμογές που πρέπει να επεξεργαστούν το περιεχόμενο των πληροφοριών αντί απλά να παρουσιάσουν τις πληροφορίες στους ανθρώπους [14]. Σκοπός της OWL είναι να παρασχεθεί ένα τυποποιημένο σχήμα που είναι συμβατό με την αρχιτεκτονική του World Wide Web. Η τυποποίηση των οντολογιών σε γλώσσα OWL θα κάνει τα δεδομένα στο Web περισσότερο επεξεργάσιμα από μηχανές και επαναχρησιμοποιήσιμα στις εφαρμογές. Έτσι, η επεκτασιμότητα, η μετατρεψιμότητα και η διαλειτουργικότητα κατέχουν υψηλή προτεραιότητα στο σχεδιασμό της γλώσσας. Συγχρόνως, η OWL προσπαθεί να επιτύχει μια καλή ανταλλαγή μεταξύ της εξελισιμότητας και της εκφραστικής δύναμης.

Μπορεί να χρησιμοποιηθεί για να αντιπροσωπεύσει σαφώς την έννοια όρων σε λεξιλόγια και τις σχέσεις μεταξύ αυτών των όρων. Αυτή η αναπαράσταση των όρων και των αλληλεξαρτήσεών τους καλείται οντολογία. Υπερβαίνει τις γλώσσες XML, XML Schema, RDF και RDF Schema αφού υποστηρίζει μεγαλύτερη διασαφήνιση του περιεχομένου του Ιστού από τις μηχανές, παρέχοντας πρόσθετο λεξιλόγιο μαζί με μια επίσημη σημασιολογία. Πιο συγκεκριμένα:

- Η XML παρέχει μια σύνταξη επιφάνειας για τα δομημένα έγγραφα, αλλά δεν επιβάλλει κανέναν σημασιολογικό περιορισμό στην έννοια αυτών των εγγράφων.
- Το XML Schema είναι μια γλώσσα για τον περιορισμό της δομής των εγγράφων XML και επεκτείνει επίσης την XML με τα datatypes.
- Η RDF είναι ένα μοντέλο δεδομένων για τα αντικείμενα και τις μεταξύ τους σχέσεις, παρέχει μια απλή σημασιολογία για αυτό το μοντέλο και αυτά τα μοντέλα μπορούν να αναπαρασταθούν σε μια σύνταξη XML.

- Το RDF Schema <http://www.w3.org/TR/2002/WD-rdf-schema-20021112/> είναι ένα λεξιλόγιο για την περιγραφή των ιδιοτήτων και των κλάσεων RDF αντικειμένων, με μια σημασιολογία για τις γενικευμένες ιεραρχίες τέτοιων ιδιοτήτων και κλάσεων.
- Η OWL προσθέτει περισσότερο λεξιλόγιο για την περιγραφή των ιδιοτήτων και των κλάσεων, όπως για παράδειγμα: σχέσεις μεταξύ των κλάσεων (π.χ. disjointness), αριθμός στοιχείων συνόλου (π.χ. "ακριβώς ένας"), λογικοί συνδυασμοί κλάσεων (π.χ. ένωση, τομή, συμπλήρωμα), ισότητα, πλουσιότεροι τύποι ιδιοτήτων, χαρακτηριστικά ιδιοτήτων (π.χ. συμμετρία, μοναδικότητα, μεταβατικότητα), και απαριθμημένες κλάσεις.

Ένα πλεονέκτημα των OWL οντολογιών θα είναι η διαθεσιμότητα των εργαλείων που μπορούν να εξάγουν συλλογισμούς για αυτές. Τα εργαλεία θα παράσχουν γενική υποστήριξη που δεν είναι συγκεκριμένη για μία ιδιαίτερη περιοχή, το οποίο θα συνέβαινε εάν επρόκειτο να δημιουργηθεί ένα σύστημα για την εξαγωγή συλλογισμών για ένα συγκεκριμένο βιομηχανικό πρότυπο XML σχήματος. Η οικοδόμηση ενός υγιούς και χρήσιμου συστήματος συλλογισμού δεν είναι μια απλή προσπάθεια. Η κατασκευή μιας οντολογίας είναι πολύ πιο βολική.

Η γλώσσα οντολογίας Ιστού OWL είναι μια σημασιολογική γλώσσα σήμανσης για την έκδοση και τη διανομή των οντολογιών στο World Wide Web. Αναπτύσσεται ως επέκταση του λεξιλογίου RDF, προέρχεται από τη γλώσσα οντολογίας Ιστού DAML+OIL και είναι γραμμένη σε XML ώστε να μπορεί εύκολα να είναι ανεξάρτητη του λειτουργικού συστήματος και των γλωσσών εφαρμογής ενός υπολογιστή. Επιπλέον, ο σημασιολογικός ιστός (Semantic Web), χρησιμοποιώντας οντολογίες βασισμένες στην XML κι όχι παραδοσιακές δομές δέντρων, θα καταστήσει ευκολότερη για τις εφαρμογές υπηρεσιών Web την επεξεργασία στοιχείων από πολλαπλές πηγές. Η γλώσσα παρέχει έναν τυποποιημένο τρόπο να καθοριστούν οι βασισμένες στο Web οντολογίες έτσι ώστε τα στοιχεία να μπορούν να περιγραφούν ως αυτό που είναι, αντί ως έγγραφο σε μια δομή δέντρων ή άλλη αφαίρεση βάσεων δεδομένων.

Μια OWL οντολογία μπορεί να περιλάβει τις περιγραφές των κλάσεων, των ιδιοτήτων και των στιγμιοτύπων τους. Λαμβάνοντας υπόψη μια τέτοια οντολογία, η επίσημη OWL



σημασιολογία διευκρινίζει πώς να παραγάγει τα λογικά συμπεράσματα της, δηλαδή γεγονότα όχι κυριολεκτικά παρόντα στην οντολογία, αλλά συνεπαγόμενα από τη σημασιολογία. Αυτές οι συνεπαγωγές μπορούν να βασιστούν σε ένα ενιαίο έγγραφο ή σε πολλαπλά κατανεμημένα έγγραφα που έχουν συνδυαστεί χρησιμοποιώντας καθορισμένους OWL μηχανισμούς.

Η OWL είναι ένα συστατικό του Semantic Web που στοχεύει να καταστήσει τους Web πόρους ευκολότερα προσιτούς στις αυτοματοποιημένες διαδικασίες, με την προσθήκη πληροφοριών για τους πόρους που περιγράφουν ή παρέχουν Web περιεχόμενο. Δεδομένου ότι το Semantic Web είναι κατανεμημένο, η OWL πρέπει να επιτρέψει να συγκεντρώνονται πληροφορίες από κατανεμημένες πηγές. Αυτό γίνεται εν μέρει επιτρέποντας στις οντολογίες να είναι συσχετισμένες, συμπεριλαμβάνοντας ρητά την εισαγωγή πληροφοριών από άλλες οντολογίες.

Επιπλέον, η OWL κάνει μια ανοικτή παγκόσμια υπόθεση. Δηλαδή, οι περιγραφές των πόρων δεν είναι περιορισμένες σε ένα ενιαίο αρχείο ή πεδίο. Ενώ η κλάση  $C_1$  μπορεί να καθοριστεί αρχικά στην οντολογία  $O_1$ , μπορεί να επεκταθεί σε άλλες οντολογίες. Οι συνέπειες αυτών των πρόσθετων προτάσεων για τη  $C_1$  είναι μονοτονικές. Οι νέες πληροφορίες δεν μπορούν να αποσύρουν τις προηγούμενες πληροφορίες. Οι νέες πληροφορίες μπορούν να είναι αντιφατικές, αλλά τα γεγονότα και οι συνεπαγωγές μπορούν μόνο να προστεθούν, ποτέ να διαγραφούν. Η δυνατότητα τέτοιων αντιφάσεων είναι κάτι που ο σχεδιαστής μιας οντολογίας πρέπει να λάβει υπόψη.

Οι πρακτικές εφαρμογές της OWL περιλαμβάνουν τα Web portals, όπου μπορεί να χρησιμοποιηθεί για να δημιουργήσει τους κανόνες κατηγοριοποίησης προκειμένου να βελτιωθεί η αναζήτηση, τις συλλογές πολυμέσων, όπου μπορεί να χρησιμοποιηθεί για να επιτρέψει βασισμένες σε περιεχόμενο αναζητήσεις των μέσων και τις υπηρεσίες Ιστού (Web services), όπου μπορεί να χρησιμοποιηθεί για την ανακάλυψη και τη σύνθεση υπηρεσιών Ιστού καθώς επίσης και για τη διαχείριση δικαιωμάτων και τον έλεγχο πρόσβασης.

### 3.3.2.2 Οι υπο-γλώσσες της OWL

Η OWL έχει τρεις όλο και περισσότερο-εκφραστικές υπο-γλώσσες: OWL Lite, OWL DL, και OWL Full [14][15].

- Η *OWL Lite* υποστηρίζει εκείνους τους χρήστες που χρειάζονται κυρίως μια ιεραρχία ταξινόμησης και απλούς περιορισμούς. Χρησιμοποιεί μόνο μερικά από τα γλωσσικά χαρακτηριστικά της OWL και έχει περισσότερους περιορισμούς στη χρήση των χαρακτηριστικών αυτών από την OWL DL ή την OWL Full. Παραδείγματος χάριν, στην OWL Lite οι κλάσεις μπορούν μόνο να οριστούν σε σχέση με κατονομασμένες (named) υπερκλάσεις (οι υπερκλάσεις δε μπορεί να είναι αυθαίρετες εκφράσεις), και μόνο ορισμένα είδη περιορισμών κλάσεων μπορούν να χρησιμοποιηθούν. Η ισοδυναμία μεταξύ κλάσεων και σχέσεων υποκλάσεων επιτρέπεται μόνο μεταξύ named κλάσεων, και όχι μεταξύ αυθαίρετων εκφράσεων κλάσης. Όμοια, οι περιορισμοί στην OWL Lite χρησιμοποιούν μόνο named κλάσεις. Η OWL Lite έχει επίσης μια περιορισμένη έννοια του αριθμού στοιχείων συνόλου - τα μόνα cardinalities που επιτρέπονται για να δηλωθούν ρητά είναι 0 ή 1. Η OWL Lite έχει χαμηλότερη πολυπλοκότητα από την OWL DL. Η OWL Lite έχει την ιδιότητα να μπορεί να γίνεται αντικείμενο αποδοτικής επεξεργασίας από αλγόριθμους εξαγωγής συμπερασμάτων (inference algorithms).
- Η *OWL DL* υποστηρίζει εκείνους τους χρήστες που θέλουν τη μέγιστη εκφραστικότητα διατηρώντας την υπολογιστική πληρότητα (όλα τα συμπεράσματα είναι εγγυημένα να είναι υπολογίσιμα) και decidability (όλοι οι υπολογισμοί θα τελειώσουν σε πεπερασμένο χρόνο). Η OWL DL περιλαμβάνει όλα τα OWL γλωσσικά κατασκευάσματα, αλλά μπορούν να χρησιμοποιηθούν μόνο κάτω από ορισμένους περιορισμούς (παραδείγματος χάριν, ενώ μια κλάση μπορεί να είναι υποκλάση πολλών κλάσεων, μια κλάση δεν μπορεί να είναι αντικείμενο instance μιας άλλης κλάσης). Η OWL DL ονομάζεται έτσι λόγω της αντιστοιχίας της με την περιγραφική λογική, έναν τομέα της έρευνας που έχει μελετήσει τη λογική που αποτελεί τη βάση της OWL. Η OWL DL και η OWL Full χρησιμοποιούν το ίδιο λεξιλόγιο αν και η OWL

DL υπόκειται σε μερικούς περιορισμούς. Κατά προσέγγιση, η OWL DL απαιτεί χωρισμό τύπων (μια κλάση δεν μπορεί να είναι επίσης άτομο ή ιδιότητα, μια ιδιότητα δεν μπορεί να είναι επίσης άτομο ή κλάση). Αυτό υπονοεί ότι οι περιορισμοί δεν μπορούν να εφαρμοστούν στα γλωσσικά στοιχεία της OWL DL (κάτι που επιτρέπεται στην OWL Full). Επιπλέον, η OWL DL απαιτεί οι ιδιότητες να είναι είτε ObjectProperties είτε DatatypeProperties: Οι DatatypeProperties είναι σχέσεις μεταξύ instances κλάσεων και RDF κυριολεκτικών (literals) και XML Schema τύπων δεδομένων, ενώ οι ObjectProperties είναι σχέσεις μεταξύ instances δύο κλάσεων. Η OWL DL έχει την ιδιότητα να μπορεί να γίνεται αντικείμενο αποδοτικής επεξεργασίας από έναν description logic reasoner.

- Η *OWL Full* απευθύνεται στους χρήστες που θέλουν τη μέγιστη εκφραστικότητα και τη συντακτική ελευθερία της RDF χωρίς υπολογιστικές εγγυήσεις. Παραδείγματος χάριν, στην *OWL Full* μια κλάση μπορεί να αντιμετωπιστεί ταυτόχρονα ως συλλογή ατόμων και ως άτομο σαν αποτέλεσμα ιδιοτήτων. Η *OWL Full* επιτρέπει σε μια οντολογία να αυξήσει την έννοια του προκαθορισμένου (RDF ή OWL) λεξιλογίου. Είναι απίθανο ότι οποιοδήποτε λογισμικό συλλογισμού θα είναι σε θέση να υποστηρίξει τον πλήρη συλλογισμό για κάθε χαρακτηριστικό γνώρισμα της *OWL Full*.

Κάθε μια από αυτές τις υπο-γλώσσες είναι μια επέκταση του απλούστερου προκατόχου της, όσον αφορά αυτό που μπορεί να εκφραστεί έγκυρα αλλά και αυτό που μπορεί να προκύψει εγκύρως ως συμπέρασμα. Ισχύουν οι παρακάτω σχέσεις:

- Κάθε νόμιμη οντολογία *OWL Lite* είναι μια νόμιμη οντολογία *OWL DL*.
- Κάθε νόμιμη οντολογία *OWL DL* είναι μια νόμιμη οντολογία *OWL Full*.
- Κάθε έγκυρο συμπέρασμα *OWL Lite* είναι ένα έγκυρο συμπέρασμα *OWL DL*.
- Κάθε έγκυρο συμπέρασμα *OWL DL* είναι ένα έγκυρο συμπέρασμα *OWL Full*.

Η επιλογή της κατάλληλης υπο-γλώσσας της OWL για την ανάπτυξη της οντολογίας εξαρτάται από τις ανάγκες της. Η επιλογή μεταξύ της OWL Lite και της OWL DL εξαρτάται από το βαθμό στον οποίο οι χρήστες απαιτούν τα περισσότερο-εκφραστικά κατασκευάσματα που παρέχονται από την OWL DL. Η επιλογή μεταξύ της OWL DL και της OWL Full εξαρτάται κυρίως από το βαθμό στον οποίο οι χρήστες απαιτούν τις ευκολίες μετα-διαμόρφωσης του σχήματος RDF (π.χ. καθορισμός κλάσεων άλλων κλάσεων, ή σύνδεση των ιδιοτήτων με τις κλάσεις). Κατά χρησιμοποίηση της OWL Full σε σύγκριση με την OWL DL, η υποστήριξη συλλογισμού είναι λιγότερο προβλέψιμη αφού πλήρεις OWL Full εφαρμογές δεν υπάρχουν.

Η OWL Full μπορεί να αντιμετωπισθεί ως επέκταση της RDF, ενώ η OWL Lite και η OWL DL μπορούν να αντιμετωπισθούν ως επεκτάσεις μιας περιορισμένης άποψης της RDF. Κάθε OWL έγγραφο (Lite, DL, Full) είναι ένα έγγραφο RDF, και κάθε έγγραφο RDF είναι ένα OWL Full έγγραφο, αλλά μόνο μερικά έγγραφα RDF θα είναι ένα νόμιμο έγγραφο OWL Lite ή OWL DL. Λόγω αυτού, κάποια προσοχή πρέπει να ληφθεί όταν θέλει ένας χρήστης να μετατρέψει ένα έγγραφο RDF σε OWL. Όταν η εκφραστικότητα της OWL DL ή της OWL Lite κρίνεται κατάλληλη, μερικές προφυλάξεις πρέπει να ληφθούν για να εξασφαλίσουν ότι το αρχικό έγγραφο RDF συμμορφώνεται με τους πρόσθετους περιορισμούς που επιβάλλονται από την OWL DL και την OWL Lite. Μεταξύ των άλλων, κάθε URI που χρησιμοποιείται όπως ένα όνομα κλάσης πρέπει να δηλωθεί σαφώς ότι είναι του τύπου `owl:Class` (και ομοίως για τις ιδιότητες), κάθε άτομο πρέπει να δηλωθεί ότι ανήκει τουλάχιστον σε μια κλάση (ακόμα και μόνο στην `owl:Thing`), τα URI που χρησιμοποιούνται για τις κλάσεις, ιδιότητες και άτομα πρέπει να είναι διαχωρισμένα.

Περισσότερες λεπτομέρειες σχετικά με τη σύνταξη της OWL υπάρχουν στις διευθύνσεις: <http://www.w3.org/TR/owl-ref/> και <http://www.w3.org/TR/owl-guide/>.

Το επόμενο κεφάλαιο περιγράφει το πρότυπο P1900 της IEEE με τη χρήση μιας οντολογίας, η οποία έχει ως στόχο την αναγνώριση των διαφόρων αδυναμιών του εν λόγω προτύπου και πιθανώς την εξεύρεση λύσεων / καλύτερων προτάσεων. Για το σκοπό αυτό χρησιμοποιείται το εργαλείο PROTÉGÉ, για τη δημιουργία της αντίστοιχης οντολογίας.

## 4. Μια οντολογία για δίκτυα B3G: το πρότυπο IEEE P1900

### 4.1 Εισαγωγή

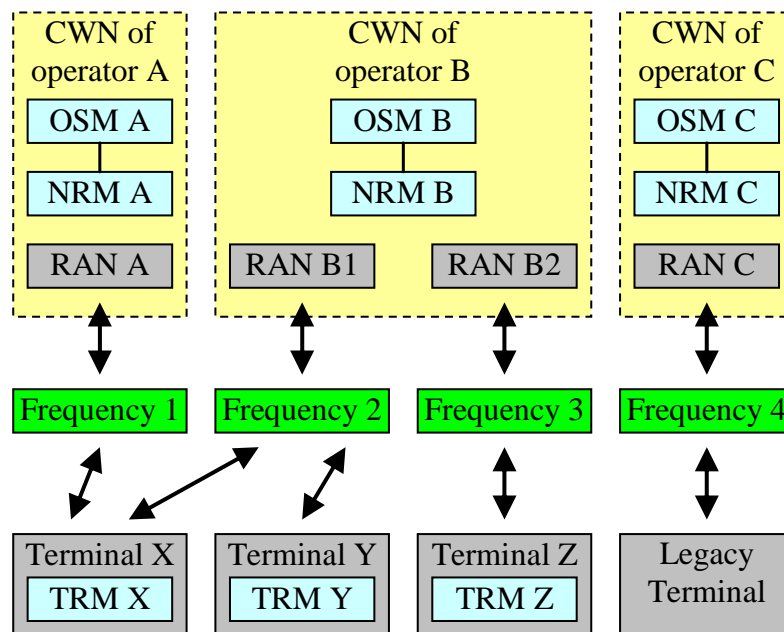
Ένα από αυτά τα που απασχολούν τους μηχανικούς δικτύων είναι η αρχιτεκτονική λειτουργίας στοιχείων δικτύου και τερματικών σε ένα B3G περιβάλλον. Πρέπει εδώ να σημειωθεί ότι κοινό τόπο αποτελεί το ότι η συμπαρουσία διαφορετικής φύσης τεχνολογιών, των οποίων επιζητούμε τη συνεργασία, απαιτεί ευφυείς μεθόδους για την αποτελεσματική διαχείριση των διαθέσιμων πόρων. Μάλιστα, οι πάροχοι δείχνουν ιδιαίτερο ενδιαφέρον προς αυτή την κατεύθυνση, δεδομένου ότι συνδέεται άμεσα με το πώς αυτοί εκμεταλλεύονται το διαθέσιμο φάσμα τους, για την αγορά του οποίου έχουν επενδύσει τεράστια ποσά. Συγκεκριμένα, αναφορικά με το φάσμα, απαιτούνται περισσότερο ευέλικτες πολιτικές από τις ρυθμιστικές αρχές, ώστε να απλοποιηθεί η διαχείριση αδειοδοτούμενου (licensed) και μη αδειοδοτούμενου (unlicensed) φάσματος και να υποστηρίξει πλέον συνεργασίες και αλλαγές στη χρήση του. Σήμερα ήδη παρατηρείται και κάποια πρόοδος προς αυτή την κατεύθυνση.

Διάφοροι παγκόσμιοι οργανισμοί πιστοποίησης αποσκοπούν στο να δημιουργήσουν και να αναπτύξουν τα λεγόμενα «πρότυπα» (standards), ήτοι σύνολα οντολογιών που αναφέρονται σε συγκεκριμένα θέματα. Ο παγκόσμιος οργανισμός ηλεκτρολόγων και ηλεκτρονικών μηχανικών (Institute of Electrical and Electronic Engineers) κινείται προς αυτή την κατεύθυνση, έχοντας ήδη εργαστεί για τη δημιουργία των σχετικών προτύπων, συνόλων δηλαδή οντολογιών. Αυτό αποτελεί και το αντικείμενο της παραγράφου αυτής.

Το πρότυπο P1900 [16] περιγράφει μια αρχιτεκτονική που αποτελείται από (α) ένα τμήμα διαχείρισης δικτύου και (β) ένα τμήμα διαχείρισης συσκευών και (γ) τις πληροφορίες που ανταλλάσσονται ανάμεσα στα (α) και (β), με στόχο την κατανεμημένη λήψη αποφάσεων που θα οδηγήσουν σε μια βελτιωμένη χρήση των πόρων του δικτύου (συμπεριλαμβανομένου του φάσματος). Το εν λόγω πρότυπο αναφέρεται σε ένα ετερογενές περιβάλλον δυναμικά αναδιαρθρούμενων στοιχείων δικτύου και τερματικών και αποσκοπεί, μέσω της αναδιάρθρωσης, στην βελτίωση των παρεχομένων επιπέδων ποιότητας υπηρεσίας (QoS levels). Στο επίπεδο που βρίσκεται αυτή τη στιγμή, το πρότυπο έχει ορίσει τη βασική αρχιτεκτονική (βασικό κομμάτι οντολογίας) που καλείται functional and system architecture και κάποιες βασικές διαδικασίες για τη λήψη αποφάσεων. Σε ένα επόμενο στάδιο θα ακολουθήσει ο λεπτομερής ορισμός των αντιστοίχων πρωτοκόλλων που περιγράφουν την ανταλλαγή πληροφορίας ανάμεσα στα διάφορα τμήματα της αρχιτεκτονικής.

#### 4.2 Περιβάλλον (context) 1900.4

Το πρότυπο IEEE P1900.4 θεωρεί ένα πλήρως ετερογενές ασύρματο περιβάλλον που φαίνεται στην παρακάτω εικόνα. Το περιβάλλον αυτό μπορεί να εμπεριέχει αρκετούς παρόχους, οι οποίοι διαθέτουν άδειες λειτουργίας περισσότερων της μίας περιοχής δικτύου (Radio Access Network – RANs). Αυτά τα RANs μπορεί να χρησιμοποιούν διάφορες διεπαφές για την επικοινωνία με τα τερματικά.



OSM – Operator Spectrum Manager  
 NRM – Network Reconfiguration Manager  
 TRM – Terminal Reconfiguration Manager  
 CWN – Composite Wireless Network  
 RAN – Radio Access Network

Εικόνα 4.1: Το ετερογενές περιβάλλον του προτύπου IEEE 1900.4

Επίσης, στα πλαίσια του εν λόγω προτύπου είναι και η ενασχόληση με προχωρημένες τεχνικές διαχείρισης φάσματος [16][17][18]. Ένα παράδειγμα τέτοιας τεχνικής είναι ο χαρακτηρισμός “spectrum assignment” σαν μία φέρουσα συχνότητα, ένα εύρος ζώνης σήματος και η τεχνολογία (radio interface) που θα χρησιμοποιηθεί, όπου επιτρέπεται η δυναμική διάθεση φάσματος σε RANs. Άλλο σχετικό παράδειγμα είναι αυτό κατά το οποίο η ανάθεση σε RANs είναι στατική, αλλά κάποια RANs επιτρέπεται να λειτουργούν ταυτόχρονα σε περισσότερες από μία (στατικές) αναθέσεις (spectrum assignments).

Τα RANs υποτίθεται ότι είναι είτε συμβατικά είτε δυναμικά αναδιαρθρούμενα (reconfigurable). Η αναδιάρθρωση μπορεί να απαιτείται επί παραδείγματι για την προσαρμογή σε ένα νέο spectrum assignment ή για τη λειτουργία των RANs ως δευτερεύοντα συστήματα.

Τα τερματικά, αντίστοιχα, που εξετάζονται στο IEEE 1900.4 πρότυπο μπορούν επίσης να είναι είτε συμβατικά είτε δυναμικά αναδιαρθρούμενα. Τα δυναμικά αναδιαρθρούμενα τερματικά μπορεί να διαθέτουν δυνατότητες να λειτουργούν ταυτόχρονα με πολλές τεχνολογίες, έχοντας δηλαδή παράλληλες ταυτόχρονες συνδέσεις (multi-homing) ή και όχι.

Ο αντικειμενικός στόχος του προτύπου IEEE 1900.4 είναι ο ορισμός ενός συστήματος διαχείρισης που αποφασίζει για ένα σύνολο ενεργειών που αποσκοπούν στη βελτιστοποίηση της χρήσης φάσματος και των παρεχομένων επιπέδων ποιότητας υπηρεσίας σε ένα ετερογενές περιβάλλον, όπως το προαναφερθέν. Συγκεκριμένα, το πρότυπο περιγράφει τις διαχειριστικές μονάδες - οντότητες (entities) και τις διεπαφές (interfaces) ενός τέτοιου συστήματος. Οι δύο κύριες οντότητες διαχείρισης είναι η «Network Reconfiguration Manager» (NRM) και η οντότητα «Terminal Reconfiguration Manager» (TRM). Η NRM είναι η μονάδα λήψης αποφάσεων από την πλευρά του δικτύου, υπεύθυνη για την αναδιάρθρωση των RANs, ενώ η TRM είναι η μονάδα λήψης αποφάσεων από την πλευρά του τερματικού, υπεύθυνη για την αναδιάρθρωση του τερματικού. Μία επιπλέον οντότητα είναι η οντότητα που χρησιμοποιείται για τη διαχείριση του φάσματος πολλών RANs από μέρους ενός παρόχου, ήτοι η οντότητα «Operator Spectrum Manager» (OSM).

Η σύνθεση του προαναφερθέντος ετερογενούς περιβάλλοντος με τις διαχειριστικές οντότητες του προτύπου 1900.4 δημιουργεί ένα «Σύνθετο Ασύρματο Δίκτυο» (Composite Wireless Network - CWN). Αυτή η σύνθεση διαθέτει πολλές καινοτόμες δυνατότητες και την ευελιξία να βελτιστοποιεί τη χρήση των πόρων του δικτύου και άρα τα παρεχόμενα επίπεδα ποιότητας υπηρεσίας.

### 4.3 Σενάρια - Use Cases

Το πρότυπο της IEEE 1900.4 ορίζει τριών ειδών use cases:

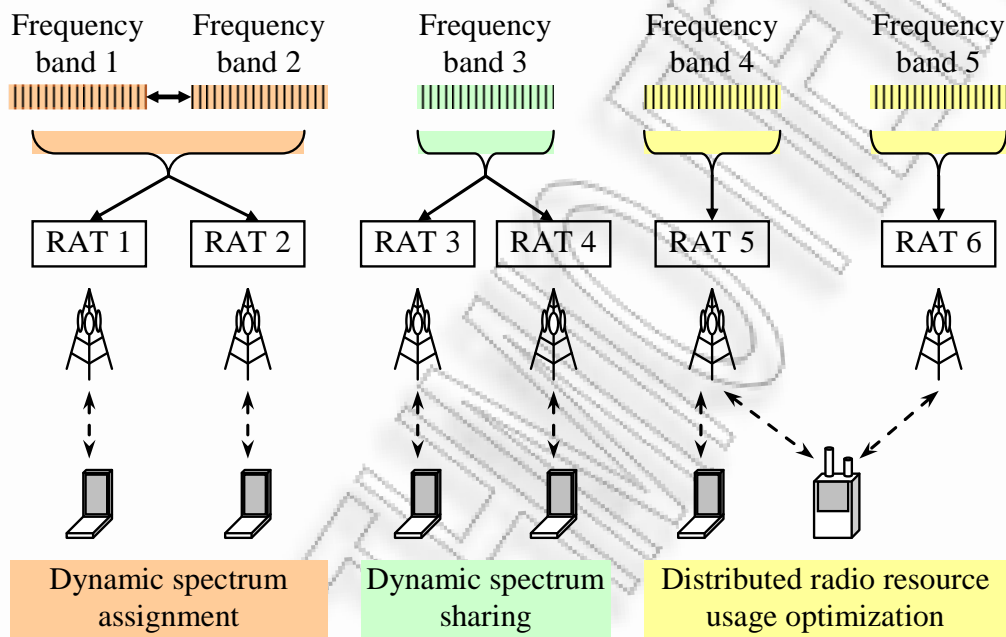
Δυναμική ανάθεση φάσματος (dynamic spectrum assignment)



Δυναμική κοινή χρήση φάσματος (dynamic spectrum sharing) και

Κατανεμημένη βελτιστοποίηση χρήσης πόρων (distributed radio resource optimisation).

Αυτά τα cases απεικονίζονται στην παρακάτω εικόνα.



Εικόνα 4.2: Use cases του προτύπου IEEE 1900.4

Στην περίπτωση του use case της δυναμικής ανάθεσης φάσματος (**dynamic spectrum assignment**), οι ζώνες συχνοτήτων ανατίθενται δυναμικά σε RANs ώστε να βελτιστοποιούν τη διαχείριση πόρων και την παροχή επιπέδων ποιότητας υπηρεσίας. Η OSM δημιουργεί (υπό τη μορφή γεννήτριας) κάποιες πολιτικές ανάθεσης φάσματος που εκφράζουν το ρυθμιστικό πλαίσιο, αλλά και τους στόχους του παρόχου για τη βελτιστοποίηση της χρήσης φάσματος. Κατόπιν, η OSM μεταφέρει αυτές τις πολιτικές στο αντίστοιχο NRM, ενώ κάθε NRM αναλύει αυτές τις πολιτικές και τις διαθέσιμες πληροφορίες περιβάλλοντος (context) και λαμβάνει δυναμικά αποφάσεις κατανομής, βάσει αυτών των στοιχείων. Μετά τη λήψη των αντίστοιχων αποφάσεων, κάθε οντότητα NRM ζητά τις αντίστοιχες διαρθρώσεις από τα

RANs. Μετά την αναδιάρθρωση του κάθε RAN, τα τερματικά χρειάζεται επίσης να αναδιαρθρωθούν.

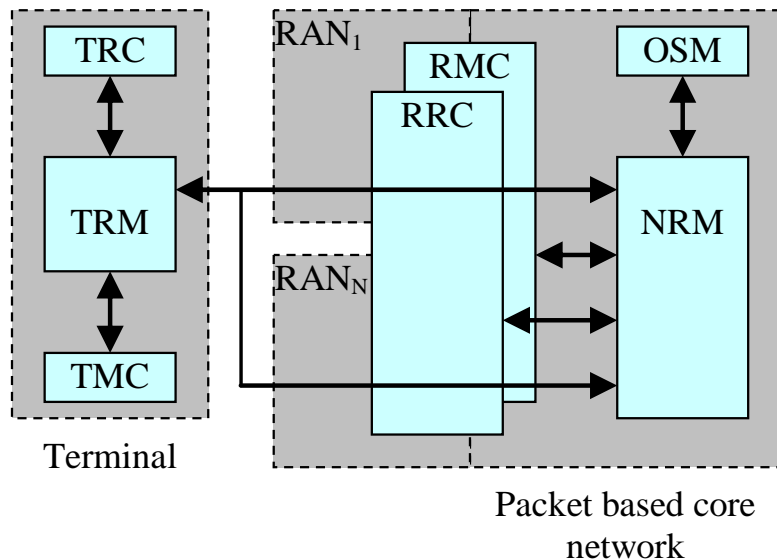
Στην περίπτωση της δυναμικής, κοινής χρήσης φάσματος (**dynamic spectrum sharing**), οι ζώνες φάσματος είναι στατικά ανατεθειμένες σε RANs. Ωστόσο, μια συγκεκριμένη ζώνη φάσματος μπορεί να χρησιμοποιείται από κοινού με αρκετά RANs ώστε να βελτιστοποιείται η χρήση φάσματος και η παροχή επιπέδων ποιότητας υπηρεσίας. Οι NRMs αναλύουν τις διαθέσιμες πληροφορίες του περιβάλλοντος (context information) και λαμβάνουν αποφάσεις σχετικά με την πρόσβαση σε μια ζώνη συχνοτήτων. Εν συνεχεία αυτών των αποφάσεων, οι NRMs ζητούν τις αντίστοιχες αναδιαρθρώσεις από τα RANs τους. Επίσης, οι NRMs δημιουργούν πολιτικές επιλογής αλγορίθμων διαχείρισης πόρων και κατόπιν τις στέλνουν στις οντότητες TRMs. Οι TRMs κατόπιν αναλύουν αυτές τις πολιτικές και, θεωρώντας και τις πληροφορίες του περιβάλλοντος, λαμβάνουν δυναμικά αποφάσεις σχετικά με το αν τα τερματικά πρέπει να αποκτήσουν πρόσβαση σε νέες ζώνες συχνοτήτων. Αυτές οι αποφάσεις λαμβάνονται στο πλαίσιο των πολιτικών που μεταβιβάζονται στα τερματικά από τις οντότητες NRM. Ακολουθώντας αυτές τις αποφάσεις, οι TRMs ζητούν τις αντίστοιχες αναδιαρθρώσεις των τερματικών, όπου είναι απαραίτητο. Πρέπει επίσης να σημειωθεί ότι η δυναμική, κοινή χρήση φάσματος περιλαμβάνει και τα σενάρια πρωτεύουσας έναντι δευτερεύουσας χρήσης φάσματος.

Τέλος, στην περίπτωση της κατανεμημένης βελτιστοποίησης χρήσης πόρων (**distributed radio resource usage optimization**), οι ζώνες φάσματος που έχουν ανατεθεί σε RANs είναι στατικές. Επιπλέον, δεν εξετάζεται η αναδιάρθρωση των RANs, αλλά αντίθετα το συγκεκριμένο θέμα περιορίζεται σε δυναμικά αναδιαρθρούμενα τερματικά (με ή χωρίς τη δυνατότητα multi-homing). Οι NRMs αναλύουν τις πληροφορίες περιβάλλοντος, δημιουργούν (γεννούν) πολιτικές επιλογής πόρων και τις αποστέλλουν στις οντότητες TRMs. Οι TRMs αναλύουν τις πολιτικές και τις πληροφορίες αυτές και λαμβάνουν αποφάσεις δυναμικά για τις αναδιαρθρώσεις των τερματικών με στόχο τη βελτιστοποίηση της χρήσης των πόρων. Και πάλι, αυτές οι αποφάσεις λαμβάνονται στο πλαίσιο των πολιτικών που μεταβιβάζονται στα τερματικά από τις οντότητες NRM. Ακολουθώντας αυτές

τις αποφάσεις, οι TRMs ζητούν τις αντίστοιχες αναδιαρθρώσεις των τερματικών, όπου είναι απαραίτητο.

#### 4.4 Αρχιτεκτονική

Η αρχιτεκτονική των διαφόρων συστημάτων διαχείρισης πόρων, όπως αυτή προτείνεται από το πρότυπο IEEE 1900.4 φαίνεται στην παρακάτω εικόνα. Το πρότυπο περιλαμβάνει 4 διαχειριστικές οντότητες από την πλευρά του δικτύου, 3 διαχειριστικές οντότητες από την πλευρά του τερματικού και 6 διεπαφές (interfaces) ανάμεσα στις ανωτέρω οντότητες.



TRC – Terminal Reconfiguration Controller  
 TRM – Terminal Reconfiguration Manager  
 TMC – Terminal Measurement Collector  
 RMC – RAN Measurement Collector  
 RRC – RAN Reconfiguration Controller  
 OSM – Operator Spectrum Manager  
 NRM – Network Reconfiguration Manager  
 RAN – radio access network

Εικόνα 4.3: Προτεινόμενη αρχιτεκτονική του προτύπου IEEE 1900.4

Οι οντότητες διαχείρισης από την πλευρά του δικτύου είναι οι ακόλουθες:

Operator Spectrum Manager (OSM);

RAN Measurement Collector (RMC);

Network Reconfiguration Manager (NRM); and

RAN Reconfiguration Controller (RRC).

**Η OSM οντότητα** καθιστά ικανό τον πάροχο να ελέγχει τις αποφάσεις της NRM για τη δυναμική ανάθεση του φάσματος. Για αυτό το σκοπό, το πρότυπο ορίζει τις λεγόμενες πολιτικές ανάθεσης φάσματος (*spectrum assignment policies*). Οι πολιτικές αυτές εκφράζουν το ρυθμιστικό πλαίσιο, ορίζοντας κανόνες για τη διαχείριση των ζωνών φάσματος που έχει διαθέσιμες ένας πάροχος. Επιπλέον, αυτές οι πολιτικές εκφράζουν τους αντικειμενικούς στόχους του παρόχου που σχετίζονται με τη διαχείριση του φάσματος. Οι πολιτικές ανάθεσης φάσματος στέλνονται από την OSM στη NRM οντότητα.

Η οντότητα **RAN Measurement Collector (RMC)** συλλέγει τις πληροφορίες περιβάλλοντος (*context information*) του RAN και τις μεταβιβάζει στη NRM. Αυτές οι πληροφορίες περιβάλλοντος περιλαμβάνουν (μεταξύ άλλων) τα ακόλουθα στοιχεία:

Αντικειμενικοί στόχοι σχετικά με τη βελτιστοποίηση χρήσης των πόρων για ένα RAN

Δυνατότητες ενός RAN

Μετρήσεις σε ένα RAN

Δυνατότητες μεταφοράς ενός RAN.

Η οντότητα **Network Reconfiguration Manager (NRM)** είναι η οντότητα που διαχειρίζεται τα ασύρματα δίκτυα και τα τερματικά για την κατανομημένη βελτιστοποίηση της διαχείρισης των πόρων και την παροχή βελτιωμένων επιπέδων ποιότητας υπηρεσίας.

Στην πλευρά του δικτύου, η NRM λαμβάνει αποφάσεις αναδιάρθρωσης του RAN και στέλνει αιτήσεις αναδιάρθρωσης στην οντότητα RAN Reconfiguration Controller (RRC). Με άλλα λόγια, η NRM αποφασίζει απευθείας για την αναδιάρθρωση των RANs που σχετίζεται με τη δυναμική ανάθεση φάσματος και με τη δυναμική, κοινή χρήση φάσματος.

Για τη διαχείριση της αναδιάρθρωσης των τερματικών από την οντότητα NRM, το πρότυπο ορίζει τις λεγόμενες πολιτικές επιλογής πόρων (*radio resource selection policies*). Αυτές οι πολιτικές στέλνονται από τη NRM στις TRMs οντότητες υπό τη δικαιοδοσία της και δημιουργούν το πλαίσιο, μέσα στο οποίο οι TRMs θα κάνουν λήψη διαφόρων αποφάσεων αναδιάρθρωσης.

Για τη διασφάλιση της σταθερής λειτουργίας των ετερογενών δικτύων, οι πολιτικές επιλογής πόρων μπορούν να συμπεριλαμβάνουν και το μέγιστο δυνατό χρονικό διάστημα που απαιτείται για την αναδιάρθρωση. Η αναδιάρθρωση ενός τερματικού πρέπει να πραγματοποιείται εντός αυτού το χρόνου, από τη στιγμή που το τερματικό λαμβάνει αυτές τις πολιτικές.

Η οντότητα **RAN Reconfiguration Controller (RRC)** είναι εκείνη η οποία ελέγχει την αναδιάρθρωση των RANs βάσει των αιτήσεων της οντότητας NRM.

Για την υποστήριξη της επιθυμητής κλιμακούμενης λειτουργίας (*scalable operation*), οι οντότητες RMC, NRM και RRC δημιουργούνται και λειτουργούν με κατανομημένο τρόπο.

Οι οντότητες διαχείρισης από την πλευρά του τερματικού είναι οι ακόλουθες:

Terminal Measurement Collector (TMC)

Terminal Reconfiguration Manager (TRM)

## Terminal Reconfiguration Controller (TRC)

Κάθε τερματικό έχει μία οντότητα TMC, μία TRM, καθώς και μία TRC.

Η οντότητα **Terminal Measurement Collector (TMC)** είναι η οντότητα που συλλέγει πληροφορίες του περιβάλλοντος του τερματικού και τις στέλνει στην TRM.

Η πληροφορία *Terminal context information* που ορίστηκε στο πρότυπο 1900.4 συμπεριλαμβάνει τα ακόλουθα στοιχεία:

Προτιμήσεις χρηστών,

Απαιτούμενα επίπεδα ποιότητας υπηρεσίας,

Δυνατότητες τερματικού,

Μετρήσεις τερματικού και

Πληροφορίες τοποθεσίας τερματικού.

Η οντότητα **Terminal Reconfiguration Manager (TRM)** είναι η οντότητα που διαχειρίζεται το τερματικό για την κατανομημένη βελτιστοποίηση των πόρων μεταξύ τερματικού και δικτύου, καθώς και την παροχή βελτιωμένων επιπέδων ποιότητας υπηρεσίας. Αυτή η βελτιστοποίηση πραγματοποιείται στο πλαίσιο που ορίζεται από την NRM και εκφράζεται από τις πολιτικές επιλογής πόρων, με τρόπο σύμφωνο με τις προτιμήσεις των χρηστών.

Η οντότητα **Terminal Reconfiguration Controller (TRC)** είναι η οντότητα που ελέγχει την αναδιάρθρωση του τερματικού βάσει των αιτήσεων που δέχεται από την TRM.

Αντιστοίχως, τα 6 interfaces που προσδιορίζονται στο συγκεκριμένο πρότυπο είναι τα ακόλουθα:

**Interface μεταξύ NRM και TRM**, το οποίο χρησιμοποιείται για να μεταφέρει τις ακόλουθες πληροφορίες:

Από NRM σε TRM:

Πολιτικές επιλογής πόρων,

Πληροφορίες περιβάλλοντος RAN και

Πληροφορίες περιβάλλοντος τερματικών (του ενός σε σχέση με άλλα).

Από TRM σε NRM:

Πληροφορίες περιβάλλοντος τερματικών σε σχέση με άλλα τερματικά της TRM.

Δύο επιλογές υπάρχουν για τη φυσική υλοποίηση αυτού του interface. Η μία από αυτές είναι η λεγόμενη σηματοδότηση εκτός ζώνης (out-band signaling), η οποία χρησιμοποιεί ένα συγκεκριμένο RAN. Η δεύτερη επιλογή είναι η σηματοδότηση εντός ζώνης (in-band signaling), η οποία χρησιμοποιεί RANs με ενεργές συνδέσεις με τα τερματικά. Το πρότυπο υποστηρίζει και τις δύο επιλογές, αλλά και την επιλογή συνδυασμού των δύο προαναφερθέντων μεθόδων.

**Interface μεταξύ TRM και TRC** το οποίο χρησιμοποιείται για να μεταφέρει τις ακόλουθες πληροφορίες:

Από TRM σε TRC:

Αιτήσεις αναδιάρθρωσης τερματικών.

Από TRC σε TRM:

Ανταποκρίσεις στις αιτήσεις αναδιάρθρωσης τερματικών.

**Interface μεταξύ TRM και TMC** το οποίο χρησιμοποιείται για να μεταφέρει τις ακόλουθες πληροφορίες:

Από TRM σε TMC:

Αιτήσεις πληροφοριών περιβάλλοντος τερματικών.

Από TMC σε TRM:

Ανταποκρίσεις στις αιτήσεις πληροφοριών περιβάλλοντος τερματικών

Οι TRC και TMC παρέχουν τυποποιημένα και ανεξάρτητα του μέσου interfaces για την υποβολή αιτήσεων από την TRM για αναδιάρθρωση τερματικών και τη λήψη πληροφοριών περιβάλλοντος των τερματικών. Αυτό διασφαλίζει ότι τα συστήματα που βασίζονται στο πρότυπο 1900.4 μπορούν να λειτουργήσουν με τερματικά που υποστηρίζουν διάφορες τεχνολογίες radio interfaces.

**Interface μεταξύ NRM και RRC**, το οποίο χρησιμοποιείται για να μεταφέρει τις ακόλουθες πληροφορίες:

Από NRM σε RRC:

Αιτήσεις αναδιάρθρωσης RAN.

Από RRC σε NRM:

Ανταποκρίσεις σε αιτήσεις αναδιάρθρωσης RAN.

**Interface μεταξύ NRM και RMC**, το οποίο χρησιμοποιείται για να μεταφέρει τις ακόλουθες πληροφορίες:

Από NRM σε RMC:

Αιτήσεις πληροφορίας περιβάλλοντος του RAN.

Από RMC σε NRM:

Ανταποκρίσεις στις αιτήσεις πληροφορίας περιβάλλοντος του RAN..



RRC and RMC provide media-independent standard interfaces for NRM to request RAN reconfiguration and to obtain RAN context information. This enables 1900.4 to support reconfiguration of various access points and base stations and to obtain context information from RANs using different radio interfaces.

**Interface μεταξύ NRM και OSM**, το οποίο χρησιμοποιείται για να μεταφέρει τις ακόλουθες πληροφορίες:

Από OSM σε NRM:

Πολιτικές ανάθεσης φάσματος.

Από NRM σε OSM:

Πληροφορίες σε αποφάσεις ανάθεσης φάσματος.

Αυτό το interface που ορίζεται στο 1900.4 δίνει στον πάροχο μια σημαντικού βαθμού δυνατότητα ελέγχου στη λειτουργία του NRM.

Επιπροσθέτως των interfaces, το πρότυπο IEEE 1900.4 ορίζει 3 «service access points» (SAPs):

rCFG\_TR\_SAP - transport SAP.

rCFG\_MEDIA\_SAP - reconfiguration and measurement SAP.

rCFG\_MNG\_SAP - management SAP.

Κάθε οντότητα του προτύπου 1900.4 θα μπορούσε να έχει ένα ή περισσότερα SAPs.

Το **Transport SAP** παρέει υπηρεσία επιπέδου μεταφοράς για τα μηνύματα που ανταλλάσσονται μεταξύ των οντοτήτων του προτύπου IEEE 1900.4. Συγκεκριμένα, γενικεύει τους μηχανισμούς επιπέδου μεταφοράς των οντοτήτων του προτύπου, παρέχοντας ένα

σύνολο από βασικούς μηχανισμούς και αντιστοιχώντας τους σε πρωτόκολλα επιπέδου μεταφοράς.

Επί παραδείγματι, αυτό το SAP χρησιμοποιείται για την ανταλλαγή πολιτικών επιλογής πόρων και πληροφοριών κατάστασης του περιβάλλοντος ανάμεσα σε NRM και TRM. Εάν υπάρχουν αρκετές οντότητες NRMs και υπάρχει ένα interface ανάμεσά τους, το SAP αυτό χρησιμοποιείται για την ανταλλαγή πληροφοριών κατάστασης του περιβάλλοντος, επιλογές ανάθεσης φάσματος, αποφάσεις αναδιάρθρωσης και πολιτικές επιλογής πόρων ανάμεσα στα NRMs.

Το **Reconfiguration and Measurement SAP** παρέχει υπηρεσίες αναδιάρθρωσης και μετρήσεων για τα υπό διαχείριση RANs και για τα τερματικά. Παρέχει ένα σύνολο από βασικούς μηχανισμούς για τις οντότητες του προτύπου για να συλλέξουν πληροφορίες της κατάστασης του RAN, των τερματικών και του περιβάλλοντός τους. Οι μηχανισμοί αυτοί αντιστοιχούνται σε συγκεκριμένα πρωτόκολλα, αναλόγως των υπό διαχείρισης RANs και των τερματικών.

Το **Management SAP** παρέχει υπηρεσίες διαχείρισης των οντοτήτων του προτύπου μέσω συμβατικών μηχανισμών διαχείρισης. Αυτό το SAP παρέχει ένα σύνολο από βασικούς μηχανισμούς για να ανταλλάξουν πληροφορίες οι οντότητες του προτύπου με αυτά τα συμβατικά συστήματα διαχείρισης.

## 5. Ανάπτυξη οντολογίας για δίκτυα B3G

### 5.1 Σκοπός - ευθύνες και λειτουργικές δυνατότητες της οντολογίας για δίκτυα B3G

#### 5.1.1 Εισαγωγή

Προκειμένου τα συστήματα B3G να εισέλθουν επιτυχώς στον εμπορικό κόσμο, απαιτείται η μελέτη πολλών συναφών παραγόντων που θα καθορίσουν την εξελικτική τους πορεία. Μάλιστα, η γενικότερη συλλειτουργία διαφορετικών τεχνολογιών και η σύγκλιση των τηλεπικοινωνιών και του διαδικτύου, επιβάλλουν τη συμμόρφωση με έννοιες όπως η απλή και ταυτόχρονα πολυδιάστατη λειτουργία αλγορίθμων και μηχανισμών, που θα δύνανται να ανανεώνονται συνεχώς, ενσωματώνοντας καινοτόμα συστατικά.

Επικεντρώνοντας την προσοχή μας στη διαχείριση των πόρων, απαραίτητη θεωρείται η ανάπτυξη ευφυών μεθόδων που θα βελτιστοποιούν τη χρήση τους επιτυχώς, ταχέως και με διαφάνεια. Οι μέθοδοι αυτές χρήζουν συνεχούς αναθεώρησης - βελτιστοποίησης, καθότι πρέπει να ανταποκρίνονται στις χωροχρονικές μεταβολές της κίνησης, προσαρμοσμένες στις εκάστοτε ανάγκες του υπό εξέταση τμήματος δικτύου και συνεπώς και του τελικού χρήστη. Αυτή η έννοια της «συνεχούς βελτιστοποίησης» συνιστά και την πεμπτουσία της διαφοροποίησης του οράματος B3G από τις συμβατικές - σημερινές τεχνολογίες, αν αναλογιστούμε ότι τα περισσότερα δίκτυα σήμερα που λειτουργούν με συγκεκριμένες τεχνολογίες, έχουν σχεδιαστεί εκ των προτέρων με τρόπο στατικό, ώστε να καλύπτουν τις

πιθανότερες ανάγκες που αναμένεται να παρουσιάσει η κίνηση ανά περιοχή και ανά χρονική περίοδο.

Αντίθετα, οι διεθνείς τάσεις σήμερα επιβάλλουν τον καθορισμό των παραμέτρων λειτουργίας των δικτύων με μεγαλύτερη ευελιξία, ώστε να επιτευχθεί στην πράξη η «προσαρμοστικότητα» που χαρακτηρίζει τα δυναμικά αναδιαρθρούμενα και γνωσιακά δίκτυα. Αυτό τελικά αναμένεται να βοηθήσει σημαντικά στη μείωση των ανεπαρειών που παρουσιάζει κάθε τεχνολογία χωριστά και άρα στην αποτελεσματικότερη ικανοποίηση των αναγκών του χρήστη.

Ένα B3G δίκτυο απαρτίζεται από πολυάριθμα τερματικά και στοιχεία δικτύου, διασπαρμένα ποικιλοτρόπως και με τρόπο χρονικά μεταβαλλόμενο, τα οποία πρέπει να αποτελέσουν αντικείμενο διαχείρισης. Ως εκ τούτου, οποιαδήποτε συγκεντρωτική απόπειρα διαχείρισης (centralized) θα αποδειχθεί εξαιρετικά πολύπλοκη και άρα τελικά αναποτελεσματική. Αντιθέτως, διεθνείς ερευνητικές προσπάθειες κατευθύνονται προς λύσεις περισσότερο (ή λιγότερο κατανεμημένες), όπως ακόμα και η αυτόνομη υπολογιστική (autonomic computing). Η αυτόνομη υπολογιστική προέρχεται από το ανθρώπινο νευρικό σύστημα. Όπως το νευρικό σύστημα επιτελεί αυτόβουλες ενέργειες (πχ. κυκλοφορία αίματος) ώστε να απελευθερώσει τον εγκέφαλο να ασχοληθεί με άλλες ενέργειες, έτσι και τα συστήματα αυτόνομης υπολογιστικής επιτελούν αυτοβούλως λειτουργίες που προηγουμένως απαιτούσαν σημαντική επέμβαση του ανθρώπινου παράγοντα (πχ. Η αναδιάρθρωση μιας συσκευής) και στόχο έχουν τη βελτιστοποίηση της λειτουργίας ενός δικτύου ή μιας συσκευής.

Σε γενικές γραμμές λοιπόν, μόνο τέτοιες κατανεμημένες προσεγγίσεις μπορούν να εγγυηθούν τον απαιτούμενο βαθμό δυνατότητας κλιμάκωσής τους αναλόγως των αναγκών (δηλαδή να πραγματοποιηθούν με τρόπο περισσότερο ή λιγότερο κατανεμημένο – scalability and modularity), ώστε να ελαττωθεί η τελική πολυπλοκότητα.

Με αυτή τη λογική, οι αλγόριθμοι διαχείρισης πόρων σε B3G δίκτυα μπορούν να ταξινομηθούν ιεραρχικά σε 2 συνεργαζόμενες βαθμίδες. Οι μηχανισμοί που υπάγονται στη βαθμίδα ελέγχου ενός ολόκληρου τμήματος του δικτύου (B3G wireless network segment)

έχουν ως πρωτεύοντα στόχο τους να συντονίζονται με το δίκτυο-κορμό, καθώς και να συντονίζουν – κατευθύνουν τις αποφάσεις των μηχανισμών που υπάγονται στη βαθμίδα ελέγχου ενός στοιχείου δικτύου, που δύναται να είναι σημείο πρόσβασης (B3G access point), σταθμός βάσης, ή οποιοδήποτε άλλο στοιχείο δικτύου. Σημειώνουμε και πάλι εδώ τη λογική που ακολουθούμε της ύπαρξης πολυαρίθμων στοιχείων δικτύου σε ένα ολόκληρο τμήμα δικτύου. Τέλος, δε θα πρέπει να παραλείψουμε και την ύπαρξη μηχανισμών που ελέγχουν τερματικά.

Διατηρώντας την προσοχή μας επικεντρωμένη σε κάθε βαθμίδα χωριστά, υποθέτουμε ότι αυτή αποτελείται από πλήθος αλγορίθμων, συνεργαζόμενων μεταξύ τους.

### **5.1.2 Σκοπός και λειτουργικές δυνατότητες οντολογίας**

Μία οντολογία για B3G δίκτυα πρέπει ως εκ τούτου να κινηθεί με γνώμονα τα παραπάνω, προκειμένου να διευκολύνει την αναφορά, τη λειτουργία και τη συνεχή ανάπτυξη και εξέλιξη των δικτύων B3G.

Σκοπός ανάπτυξης της προτεινόμενης οντολογίας είναι η μοντελοποίηση όλων των πιθανών διαδικασιών και συναλλαγών που απαιτείται να διενεργεί μια οντότητα που εμπλέκεται στη διαχείριση ετερογενών δικτύων με μια άλλη οντότητα που εμπλέκεται στο ίδιο σενάριο, χρησιμοποιώντας τον απλούστερο δυνατό τρόπο.

Τα οφέλη από τη χρήση μιας οντολογίας για τη διαχείριση δικτύων B3G συνίστανται στα παρακάτω:

1) Οι διάφορες διαχειριστικές οντότητες σε ένα B3G περιβάλλον ανταλλάσσουν μεγάλο όγκο πληροφορίας και αυτός ο όγκος δυσχεραίνει την ορθή επικοινωνία μεταξύ τους και άρα και αυτή καθαυτή τη λειτουργία της διαχείρισης. Επομένως, η οργάνωση αυτού του όγκου πληροφορίας θα διευκολύνει σημαντικά το έργο της διαχείρισης και αυτή ακριβώς είναι μια βασική λειτουργία της οντολογίας.

2) Η οργάνωση του όγκου της μεταβιβαζόμενης πληροφορίας μεταξύ των διαχειριστικών οντοτήτων θα επιφέρει και μια συστηματοποίηση και τυποποίηση του λογισμικού (software) που θα αναπτυχθεί για την εμπορευματοποίηση των διαχειριστικών οντοτήτων. Όταν η προς μετάδοση πληροφορία είναι τυποποιημένη και οργανωμένη σωστά με χρήση κάποιας οντολογίας, τότε και η ανάπτυξη του λογισμικού από τις εταιρείες ανάπτυξης software καθίσταται ευκολότερη και εμπορικά συμφέρουσα.

3) Η εύκολη και γρήγορη ανάπτυξη λογισμικού σε συνδυασμό με τη συστηματοποίηση και οργάνωση της προς ανταλλαγή πληροφορίας θα έχει ως αποτέλεσμα και την ενασχόληση περαιτέρω ερευνητικά με τη διαχείριση δικτύων B3G και ως εκ τούτου, νέες δυνατότητες και νέοι αλγόριθμοι διαχείρισης θα μπορούν ευκολότερα να αναπτυχθούν και να βελτιώσουν τους υπάρχοντες.

Από τα παραπάνω είναι προφανές ότι η ανάπτυξη οντολογίας εξυπηρετεί πολλαπλούς σκοπούς στα δίκτυα B3G και η ύπαρξή της είναι παραπάνω από αναγκαία για τη βελτίωση της λειτουργίας των εν λόγω δικτύων. Η περιγραφή της οντολογίας ακολουθεί παρακάτω, με ταυτόχρονη εξέταση της ύπαρξης της δυνατότητας απλούστευσης του προτύπου P1900, το οποίο τρόπον τινά αποτελεί από μόνο του μια προσπάθεια δημιουργίας οντολογίας.

## 5.2 To Protégé 2000

Το Protégé είναι ένα open-source πρόγραμμα της ακαδημαϊκής κοινότητας του Πανεπιστημίου του Stanford για τη σύνταξη οντολογιών και βάσεων γνώσεων. Αποτελεί ένα επεκτάσιμο περιβάλλον το οποίο είναι ανεξάρτητο από πλατφόρμα λογισμικού και διατίθεται ελεύθερα στο διαδίκτυο. Το Protégé 2000 αποτελεί το πιο σύγχρονο εργαλείο σε αυτή τη σειρά.

Ο σκοπός του εργαλείου είναι να καθοδηγήσει την ανάπτυξη συστημάτων καθιστώντας δυνατή την επαναχρησιμοποίηση οντολογιών και μεθόδων επίλυσης προβλημάτων, μειώνοντας επομένως το χρόνο που απαιτείται για την ανάπτυξη και συντήρηση προγραμμάτων.

Το Protégé 2000 δίνει τη δυνατότητα στους χρήστες του να κατασκευάζουν οντολογίες για θεματικές περιοχές, να προσαρμόζουν στις δικές τους απαιτήσεις τις φόρμες εισαγωγής δεδομένων και να εισάγουν τα δεδομένα. Επίσης ξεχωρίζει, για ακόμα δύο χαρακτηριστικά του, σε σχέση με τα περισσότερα περιβάλλοντα επεξεργασίας οντολογιών: την εξελιξιμότητα και την επεκτασιμότητα του. Η βασισμένη σε ενότητες αρχιτεκτονική (component-based architecture) επιτρέπει στους δημιουργούς – προγραμματιστές να προσθέτουν λειτουργικότητα στο σύστημα απλά προσθέτοντας νέα κατάλληλα, αυτόνομα τμήματα κώδικα (plugins). Συνεπώς, μπορεί εύκολα να επεκταθεί ώστε να περιλαμβάνει γραφικά συστατικά όπως γραφικές παραστάσεις και πίνακες, μέσα όπως ήχο, εικόνες και video, καθώς και πολλαπλές μορφές αποθήκευσης όπως OWL, RDF, XML και HTML. Η αντίστοιχη βιβλιοθήκη του (Protégé Plugin Library) περιέχει συνεισφορές προγραμματιστών από ολόκληρο τον κόσμο.

Με το Protégé 2000 είναι δυνατή η ταυτόχρονη εργασία με κλάσεις οντολογιών και στιγμιότυπα (instances) αυτών των κλάσεων. Μια οντολογία είναι ένα μοντέλο ενός συγκεκριμένου πεδίου γνώσης, το οποίο στο Protégé αναπαριστάται ως ένα σύνολο κλάσεων με τις αντίστοιχες ιδιότητες (slots). Ένα στιγμιότυπο είναι ένα συγκεκριμένο για μια κλάση, το οποίο εισάγεται σε μια βάση γνώσεων (π.χ. το όνομα John Smith μπορεί να είναι ένα στιγμιότυπο του slot Name). Το Protégé υποστηρίζει δομή κλάσεων, με ιεραρχία αυτών και πολλαπλή κληρονομικότητα, μετακλάσεις και ιεραρχία αυτών κ.ο.κ.

Τέλος, διευκολύνει την συμφωνία με το πρωτόκολλο Open Knowledge Base Connectivity (OKBC) για πρόσβαση σε συστήματα βάσεων γνώσεων και δίνει τη δυνατότητα να εκτελούνται εφαρμογές βασισμένες στα συστατικά του Protégé 2000 και ενσωματωμένες στο περιβάλλον του.

## 5.3 Περιγραφή και αναπαράσταση οντολογίας

### 5.3.1 Περιγραφή

Η προτεινόμενη οντολογία περιγράφει και τυποποιεί την πληροφορία που αναφέρεται στα ακόλουθα:

A) τη διάρθρωση (configuration) της συνολικής δικτυακής υποδομής (infrastructure)

B) το προσφερόμενο και εξυπηρετούμενο φορτίο στο υπό εξέταση στοιχείο ή τμήμα δικτύου

Γ) τη διάρθρωση (configuration) των τερματικών που εμπύπτουν στη συγκεκριμένη περιοχή

Δ) τις πολιτικές που υποδεικνύονται από τον πάροχο του δικτύου και πρέπει με κάποιο τρόπο να γίνονται γνωστές και να μεταφέρονται καταλλήλως και στα τερματικά, αλλά και στους σταθμούς βάσης

Σε γενικές γραμμές, η παρούσα οντολογία χρησιμοποιείται ως τρόπος για να ομαδοποιήσει καταλλήλως τις δομές όλων των διαχειριστικών οντοτήτων που εμπλέκονται στη διαχείριση δικτύων (και τερματικών) B3G, ήτοι τις εισόδους και εξόδους των, όπως αυτές περιγράφηκαν στις παραγράφους 2.2 και 2.3 της παρούσης διατριβής.

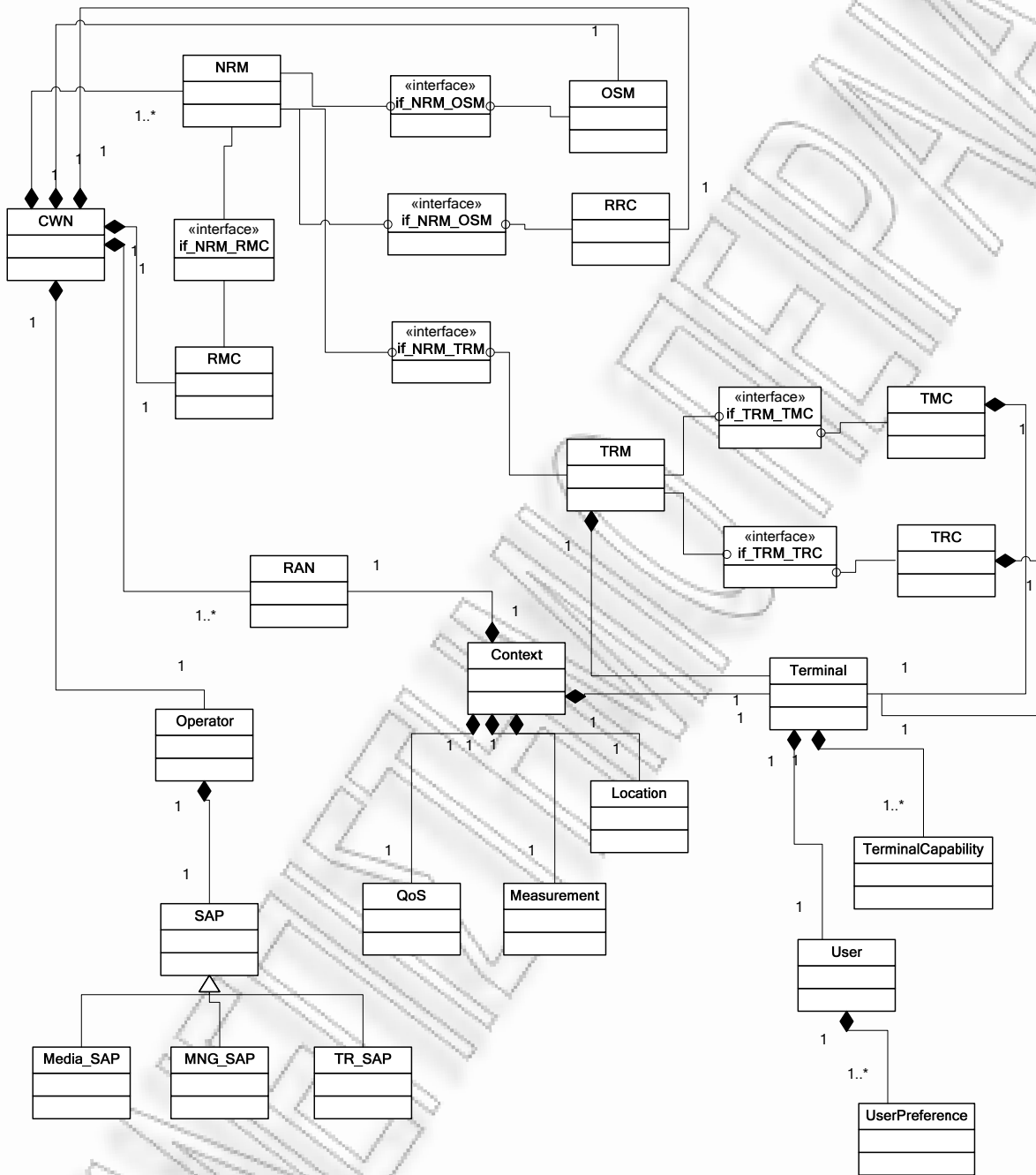
### 5.3.2 Αναπαράσταση

Η παρούσα παράγραφος παρουσιάζει τα αποτελέσματα που προέκυψαν από τη χρήση του εργαλείου PROTÉGÉ για τη δημιουργία οντολογίας. Για τη δημιουργία αυτή, πάρθηκαν όσες δομές και παράμετροι χρειάστηκαν από το μοντέλο πληροφορίας (information model) του προτύπου P1900. Η τελική πρόταση ομοιάζει αυτής του προτύπου P1900 και θα



εξεταστούν τυχόν αδυναμίες του προτύπου που βελτιώνονται με τη χρήση της συγκεκριμένης οντολογίας. Οι δομές δε θα ξαναπεριγραφούν εδώ, ωστόσο χρησιμοποιούνται με στόχο την κάλυψη όλων των πιθανών σεναρίων που προέκυψαν από τη μελέτη του προτύπου P1900, αλλά και από τις προ-περιγραφείσες ανάγκες λειτουργικής διαχείρισης των δικτύων B3G.

Το παρακάτω διάγραμμα (Εικόνα 5-1) παρουσιάζει το μοντέλο πληροφορίας της οντολογίας που προκύπτει. Πιο συγκεκριμένα, απεικονίζει το διάγραμμα κλάσεων της υλοποίησης σύμφωνα με την αρχιτεκτονική που περιγράφηκε στην ενότητα (§4.4). Επιπλέον, το παράρτημα 8.3 περιλαμβάνει αναλυτική λίστα των κλάσεων (ιδιότητες και μέθοδοι ανά κλάση). Οι παρακάτω ενότητες επικεντρώνονται στις οντότητες που αποτελούν τον κορμό της προτεινόμενης οντολογίας όπως οι κλάσεις OSM, NRM, TRM και στις διεπαφές μεταξύ τους.



Εικόνα 5.1: UML class diagram: κλάσεις υλοποίησης της οντολογίας – συσχέτιση με κλάσεις του μονέλου P1900.

### 5.3.2.1 OSM

Η OSM οντότητα καθιστά ικανό τον πάροχο (Operator) για έλεγχο των αποφάσεων της NRM για την δυναμική ανάθεση του φάσματος σύμφωνα με της τρέχουσες απαιτήσεις. Για να πραγματοποιηθεί η συγκεκριμένη διαδικασία γίνεται ορισμός και χρήση πολιτικών ανάθεσης φάσματος από το ίδιο το πρότυπο. Αυτές οι πολιτικές αποτελούν το ρυθμιστικό πλαίσιο κάνοντας την χρήση ενός συνόλου κανόνων οι οποίοι ορίζουν τον τρόπο διαχείρισης των διαθέσιμων ζωνών φάσματος του κάθε παρόχου. Οι πολιτικές αυτές αποτελούν την έκφραση των αντικειμενικών στόχων του παρόχου σε σχέση με το διαθέσιμο φάσμα και την διάθεση του στα τερματικά. Αυτές οι πολιτικές υλοποιούνται από την NRM οντότητα.

#### Ιδιότητες:

- private Vector<RAN> RANs; //αντικείμενα RANS που συσχετίζονται με ένα μόνο πάροχο

#### Μέθοδοι:

- Μέθοδοι εισόδου απαιτήσεων για αλλαγές στις πολιτικές ανάθεσης φάσματος- οι μέθοδοι προσπελάζονται από την οντότητα **Operator**
- Μέθοδοι διαχείρισης των **RANs** για κάθε οντότητα **Operator**

### 5.3.2.2 NRM

Η NRM οντότητα είναι υπεύθυνη για την διαχείριση τόσο των διαθέσιμων δικτύων όσο και των τερματικών που περιλαμβάνονται σε αυτά με σκοπό την επίτευξη της βελτιστοποίησης τόσο στον τομέα της διαχείρισης όσο και στον τομέα της ποιότητας υπηρεσίας.

#### Ιδιότητες:

private Vector<Terminal> Terminals;

private Vector<RAN> RANs;

**Μέθοδοι:**

Μέθοδος	Περιγραφή
private Measurement GetMeasurements(RAN someRAN);	Ανακτά μετρήσεις ( <b>measurements</b> ) από την οντότητα <b>RAN</b> μέσω της <b>RMC</b>
private Context GetContext(RAN someRAN);	Ανακτά πληροφορίες και τις επεξεργάζεται από τα <b>RAN</b> και <b>Terminal</b> αντικείμενα
private void SendContext(Context someContext, RAN someRAN);	Στέλνει ένα αντικείμενο <b>Context</b> σε όλες τις οντότητες <b>RANs</b> περιλαμβάνοντας: <ul style="list-style-type: none"> <li>○ z;vnew syxnot;htvn (frequency bands)</li> <li>○ dikai;vmata xr;hshw f;asmataw (spectrum usage rights)</li> <li>○ <b>RAT</b></li> </ul>
private Measurement GetMeasurements(Terminal someTerminal);	Ανακτά μετρήσεις ( <b>measurements</b> ) των τερματικών μέσω της <b>TMC</b>
private Context GetContext(Terminal someTerminal);	Ανακτά πληροφορίες για τη κατάσταση του κάθε <b>Terminal</b> (context) από την οντότητα <b>TRM</b> περιλαμβάνοντας τα παρακάτω: <ul style="list-style-type: none"> <li>○ Λειτουργικές δυνατότητες</li> <li>○ προτιμήσεις</li> <li>○ απαιτούμενο QoS</li> <li>○ Μετρήσεις</li> <li>○ Τοποθεσία</li> </ul>
private Object OptimumAllocation(Vector<RAN> someRANs, Vector<Terminal> someTerminals);	Καθορίζει ένα βέλτιστο τρόπο κατανομής των συχνοτήτων και των RATs σε όλα τα αντικείμενα <b>RAN</b> λαμβάνοντας υπόψη το φάσμα που είναι ελεύθερο ή το φάσμα που έχει χρησιμοποιηθεί
private Object CreatePolicy(Object OptimumAllocationResult);	Οι αποφάσεις μεταφράζονται σε πολιτικές που τις στέλνει σε όλες τις οντότητες των τερματικών
private boolean SendPolicy(RAN someRAN);	Καθορίζει τους κανόνες και τις πληροφορίες και αποφασίζει τη κατανομή των συχνοτήτων και τα <b>RATs</b> για όλες τις

	<p>οντότητες <b>RAN</b> που διαχειρίζεται.</p> <p>Κατόπιν στέλνει εντολές στις οντότητες <b>RAN RRC</b> μέσω του για αναδιαμόρφωση τους σχετικά με:</p> <ul style="list-style-type: none"> <li>○ συχνότητες</li> <li>○ RAT</li> </ul>
private boolean SendPolicy(Terminal someTerm);	<p>Δίνει εντολή στα αντικείμενα Terminal να αναδιαμορφωθούν κάθε φορά που:</p> <ul style="list-style-type: none"> <li>○ Νέος μεταγωγέας δικτύου (carrier) προστίθεται</li> <li>○ Γίνονται αλλαγές στο φάσμα</li> <li>○ κα</li> </ul>
private void SendPolicyToRepository(Object CreatedPolicy, Date CurrentDateTime);	<p>Κάνει εκτίμηση των πολιτικών και στέλνει το αποτέλεσμα της εκτίμησης στην οντότητα Repository</p>
private void SendNetworkRecalculationRequest(TRM someTRM);	<p>Παρέχει πολιτικές (περιορισμούς) μέσω του <b>TRM</b> προς τις οντότητες <b>Terminal</b> για το βέλτιστο σχηματισμό του δικτύου και τη προσαρμογή τους σε κάθε αλλαγή.</p>
private void SendContext(Context someContext, Terminal someTerm);	<p>Στέλνει στο Terminal τα παρακάτω (Context):</p> <ul style="list-style-type: none"> <li>○ Φάσμα συχνοτήτων των RANs</li> <li>○ κανάλια</li> <li>○ δικαιώματα χρήσης φάσματος</li> <li>○ λειτουργικές δυνατότητες ανά RAN</li> <li>○ <b>Μετρήσεις</b> (SNR, traffic load, BLER, throughput, packet loss rate, latency)</li> </ul>

### 5.3.2.3 TRM

Η TRM οντότητα είναι υπεύθυνη για την διαχείριση του τερματικού όσο αφορά την κατανομημένη βελτιστοποίηση των διαθέσιμων πόρων μεταξύ του τερματικού και του χρησιμοποιούμενου δικτύου αλλά και την παροχή της βέλτιστης δυνατής ποιότητας επικοινωνίας. Η επίτευξη της βελτιστοποίησης πραγματοποιείται μεν υπό τους πλαίσιο που ορίζεται από την NRM και εκφράζεται δε από τις πολιτικές επιλογής πόρων ο οποίος είναι σύμφωνος με τις προτιμήσεις του εκάστοτε χρήστη.

Ιδιότητες:

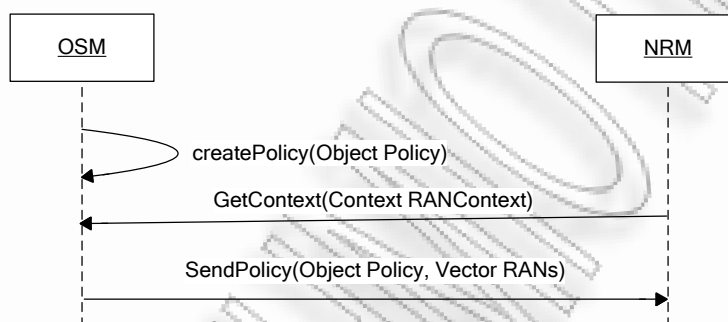
```
private Vector<Terminal> Terminals;
```

Μέθοδοι:

Μέθοδος	Περιγραφή
private void SendContext(Terminals someTerminal, NRM someNRM);	Συγκεντρώνει πληροφορίες για τη κατάσταση των τερματικών και τη στέλνει στην οντότητα NRM
private boolean SendPolicy(Terminal someTerminal, TMC someTMC);	Στέλνει πολιτικές εφαρμογής από τα τερματικά μέσω της οντότητας TMC
private boolean SendReconfigurationRequest(Object Policy, TRC someTRC);	Στέλνει εντολή αναδιαμόρφωσης στην οντότητα TRC

### 5.3.2.4 Interface OSM - NRM

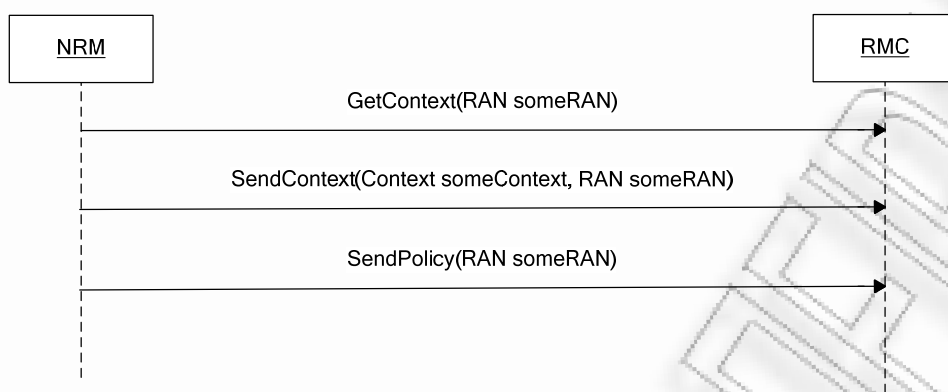
Το NRM - RMC Interface είναι η διεπαφή η οποία χρησιμοποιείται για να μεταφέρει τις πολιτικές ανάθεσης φάσματος από OSM στην NRM και πληροφορίες σε αποφάσεις ανάθεσης φάσματος από NRM σε OSM.



Εικόνα 5.2: UML sequence diagram: κλάσεις υλοποίησης της OSM-NRM διεπαφής

### 5.3.2.5 Interface NRM – RMC

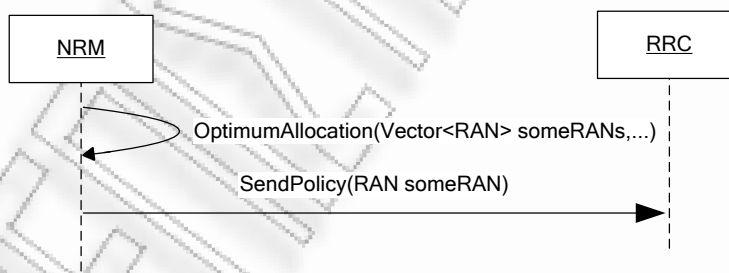
Η NRM - RMC διεπαφή χρησιμοποιείται για την μεταφορά πληροφοριών οι οποίες είναι απαραίτητες μεταξύ NRM και RMC, όπως αιτήσεις πληροφορίας περιβάλλοντος του RAN από τον NRM στον RMC και ανταποκρίσεις στις αιτήσεις του περιβάλλοντος του RAN από τον RMC στον NRM.



Εικόνα 5.3: UML sequence diagram: κλάσεις υλοποίησης της NRM-RMC διεπαφής

### 5.3.2.6 Interface NRM – RRC

Η NRM - RRC διεπαφή χρησιμοποιείται για την μεταφορά αιτημάτων αναδιάρθρωσης RAN από την NRM στην RRC οντότητα.

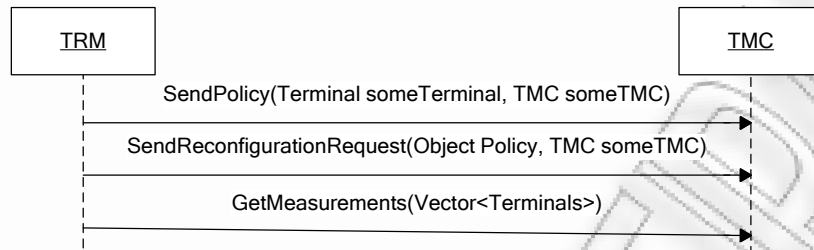


Εικόνα 5.4: UML sequence diagram: κλάσεις υλοποίησης της NRM-RRC διεπαφής

### 5.3.2.7 Interface TRM - TMC

Η διεπαφή μεταξύ TRM και TMC χρησιμοποιείται για να μεταφέρει αιτήσεις πληροφοριών περιβάλλοντος τερματικών από TRM σε TMC και ανταποκρίσεις στις αιτήσεις πληροφοριών από TMC σε TRM.

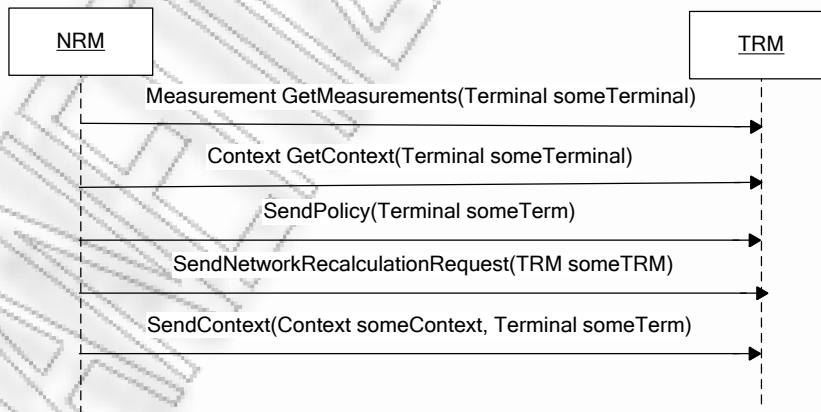




Εικόνα 5.5: UML sequence diagram: κλάσεις υλοποίησης της TRM-TMC διεπαφής

### 5.3.2.8 Interface NRM-TRM

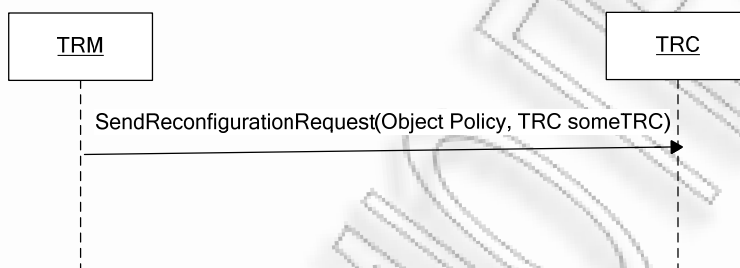
Η διεπαφή μεταξύ **NRM** και **TRM** χρησιμοποιείται για να μεταφέρει πολιτικές επιλογής πόρων, πληροφορίες περιβάλλοντος RAN και πληροφορίες περιβάλλοντος τερματικών (του ενός σε σχέση με άλλα) από NRM σε TRM; και πληροφορίες περιβάλλοντος τερματικών σε σχέση με άλλα τερματικά της TRM από TRM σε NRM.



Εικόνα 5.6: UML sequence diagram: κλάσεις υλοποίησης της NRM-TRM διεπαφής

### 5.3.2.9 Interface TRM - TRC

Το TRM – TRC interface είναι η διεπαφή η οποία χρησιμοποιείται για την μεταφορά πληροφοριών που αφορούν τις αιτήσεις για αναδιάρθρωση τερματικών μεταξύ των TRM και TRC αλλά και τις ανταποκρίσεις στις προαναφερθείσες αιτήσεις από τον TRC στον TRM.



Εικόνα 5.7: UML sequence diagram: κλάσεις υλοποίησης της TRM-TRC διεπαφής

### 5.3.2.10 Reconfiguration and measurement SAP

Το συγκεκριμένο SAP είναι υπεύθυνο για την παροχή υπηρεσιών αναδιάρθρωσης και μέτρησης για όλα τα διαχειριζόμενα RAN και τερματικά. Για την επίτευξη του στόχου του παρέχονται ένα σύνολο από βασικούς μηχανισμούς για της οντότητες που περιγράφονται από το πρότυπο με σκοπό την συλλογή πληροφοριών που αφορούν την κατάσταση του RAN, των τερματικών και του περιβάλλοντος τους. Οι μηχανισμοί αυτοί αντιστοιχούνται σε συγκεκριμένα πρωτόκολλα τα οποία αλλάζουν ανάλογα με τα διαθέσιμα RAN και τερματικά υπο διαχείριση. Το παράρτημα 8.3 περιλαμβάνει λίστα των μεθόδων που υλοποιούνται στη κλάση SAP.

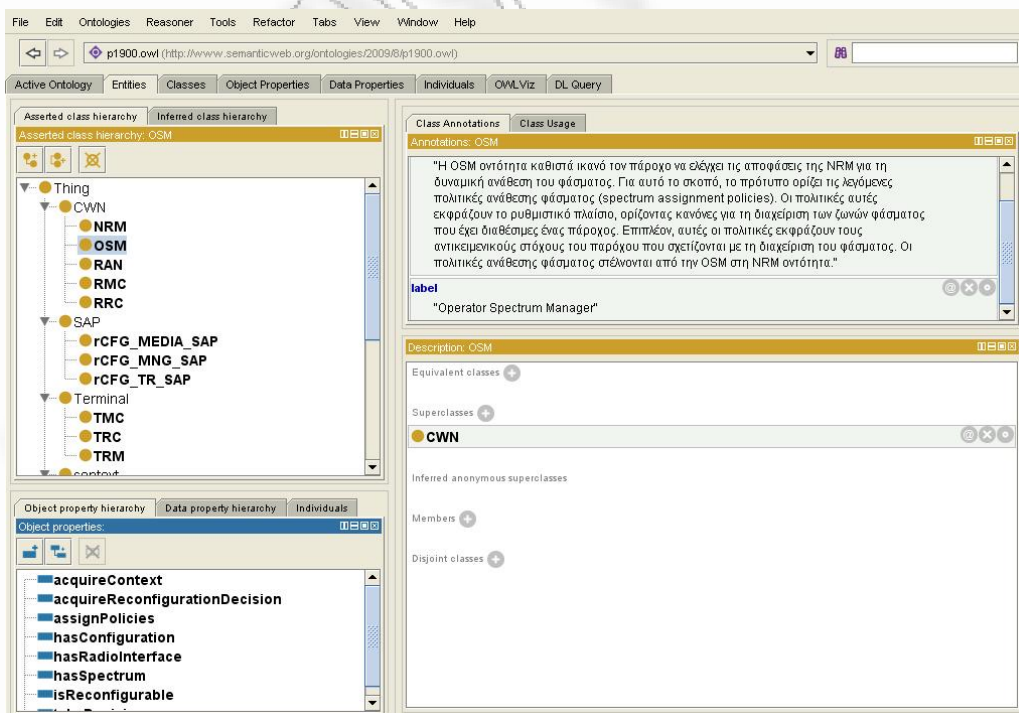
### 5.3.3 Οπτική αναπαράσταση με την χρήση του εργαλείου PROTEGÉ.

Οι παρακάτω εικόνες περιγράφουν τις δομές του προτύπου που χρησιμοποιήθηκαν, όπως φαίνονται στο PROTEGÉ.

Η κλάση *OntoMapper* είναι ουσιαστικά η κλάση-δοχείο (“container”) μέσα στην οποία δημιουργούνται και προστίθενται οι κλάσεις των οντοτήτων, οι διεπαφές μεταξύ τους και τα μηνύματα που ανταλλάσσουν μεταξύ τους (Εικόνα 5-1 έως 5-7). Η κλάση *OntoMapper* περιλαμβάνει 2 κύριες συλλογές (collections) κλάσεων για την απεικόνιση των παραπάνω:

- *RDFResource*
- *RDFProperty*.

Ως υπο-κλάσεις της κλάσης *RDFResource* ορίζονται οι κλάσεις των οντοτήτων και οι διεπαφές τους. Τέτοιες είναι οι κλάσεις όπως *CWN*, *SAP*, *Terminal*. Κάτω από την *CWN* αντιστοιχούν οι κλάσεις που συνθέτονται στην *CWN*: *NRM*, *OSM*, *RAN*, *RMC*, *RRC*. Αντίστοιχα, κάτω από τη *SAP* απεικονίζονται όλες οι υπο-κλάσεις. Κάτω από τη κλάση *Terminal* ορίζονται οι *TRC*, *TMC*, *TRM* για τη διαχείριση τερματικών.

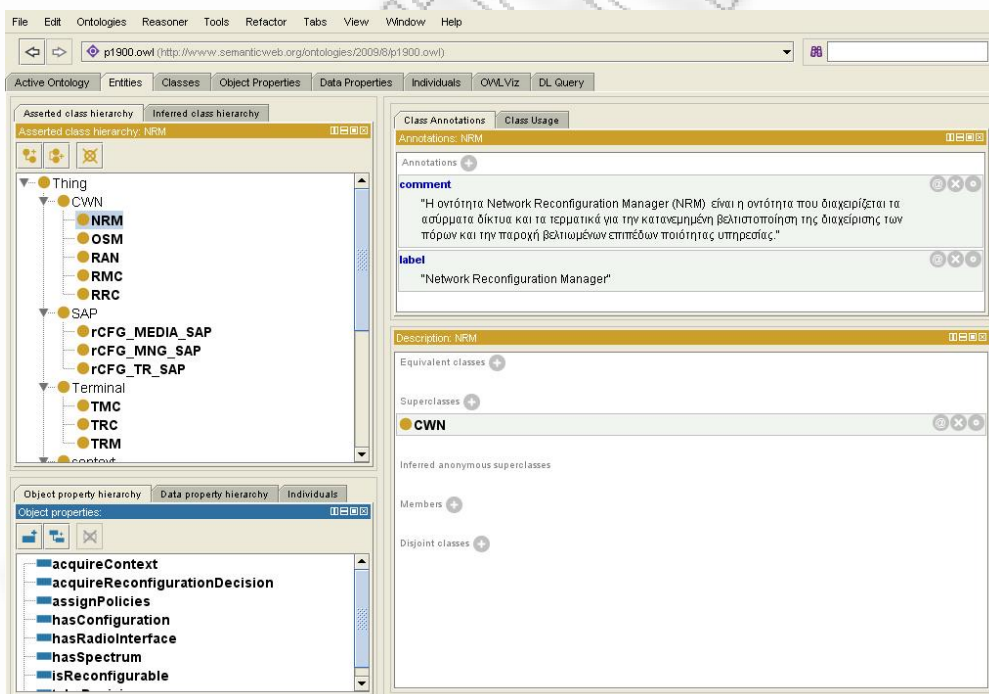


Εικόνα 5.8: OSM

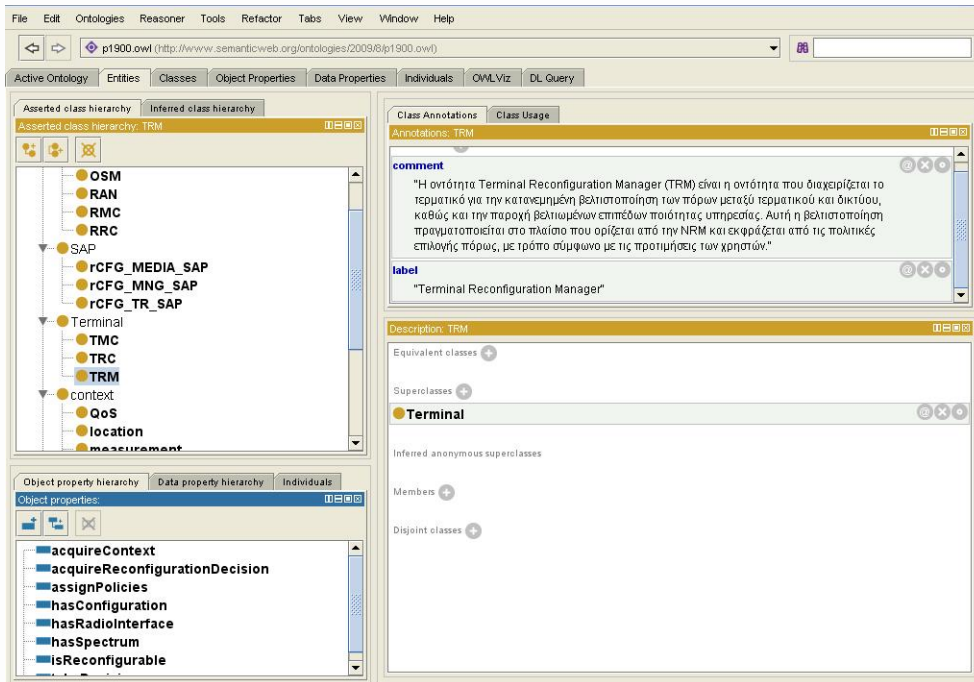
Ως υπο-κλάσεις της RDFResource κληρονομούν μία σειρά μεθόδων που διευκολύνουν τη διαχείριση τους στον container της κλάσης OntoMapper:

```
public RDFSNamedClass getOSMClass() {
public OSM createOSM(String name) {}
public OSM getOSM(String name) {}
public Collection<OSM> getAllOSMInstances() {}
public Collection<OSM> getAllOSMInstances(boolean transitive) {}
```

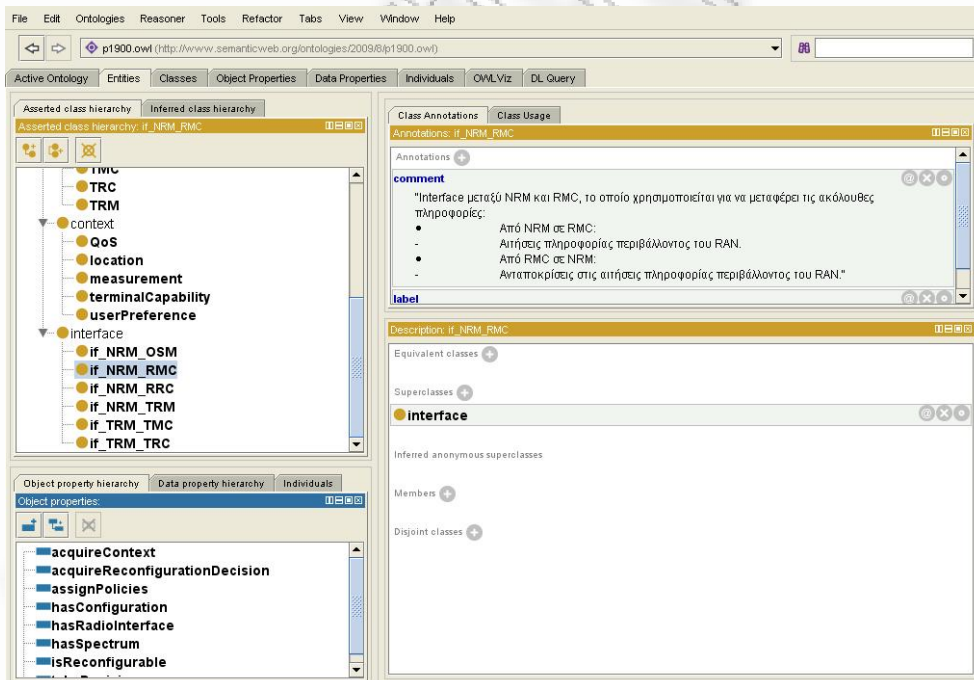
Αντίστοιχες μέθοδοι δημιουργούνται για τις υπόλοιπες κλάσεις που αντιπροσωπεύουν τις οντότητες του μοντέλου. Οι μέθοδοι αυτοί και γενικότερα ο πηγαίος κώδικας που δημιουργείται από το Protégé περιγράφεται στην ενότητα §8.1.



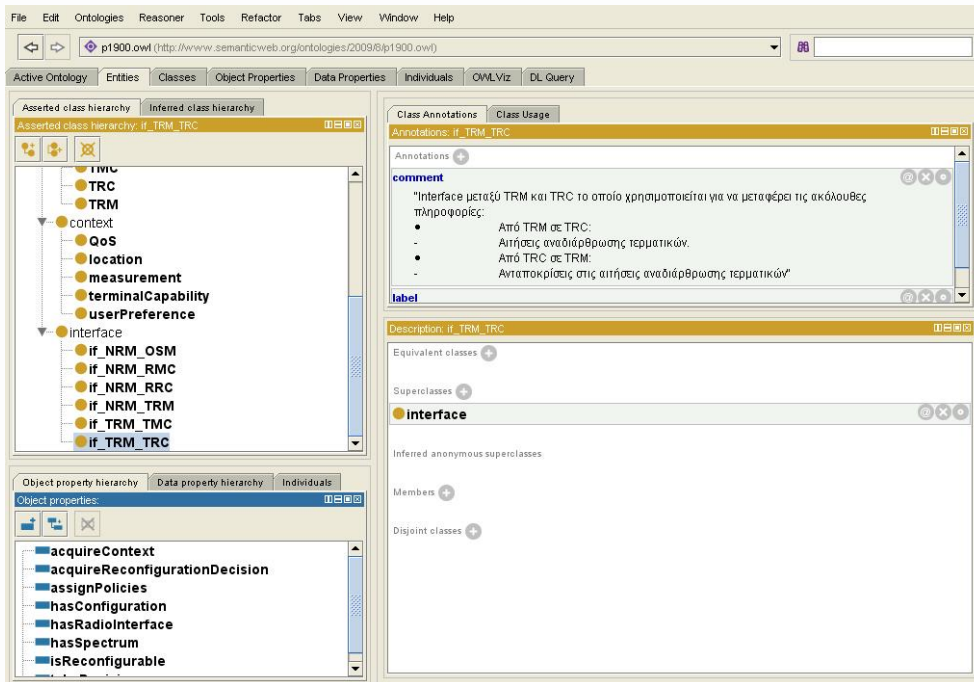
Εικόνα 5.9: NRM



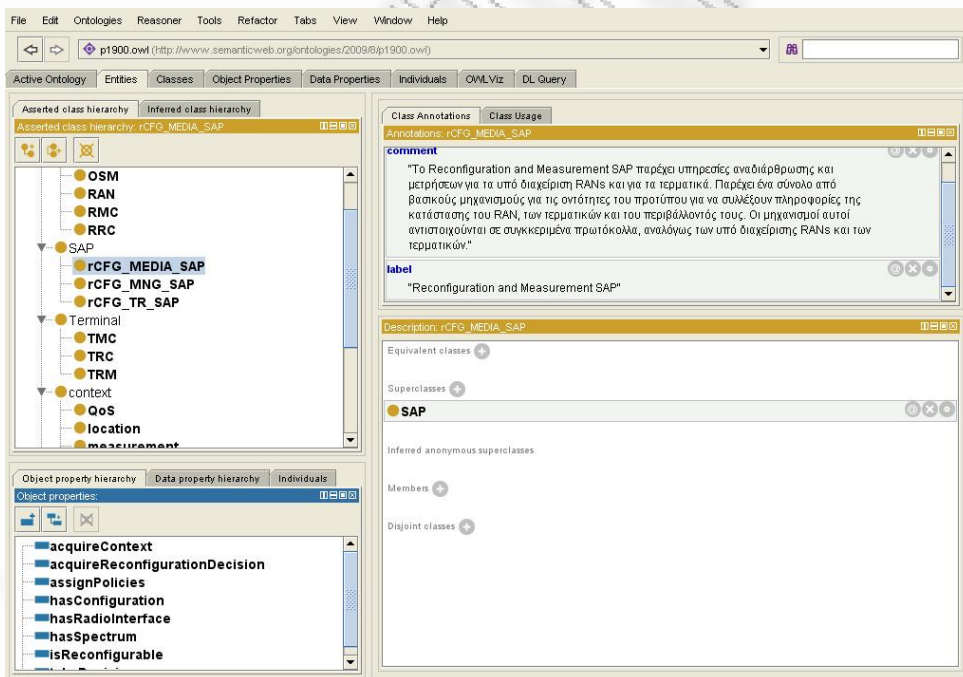
Εικόνα 5.10: TRM



Εικόνα 5.11: if RMC

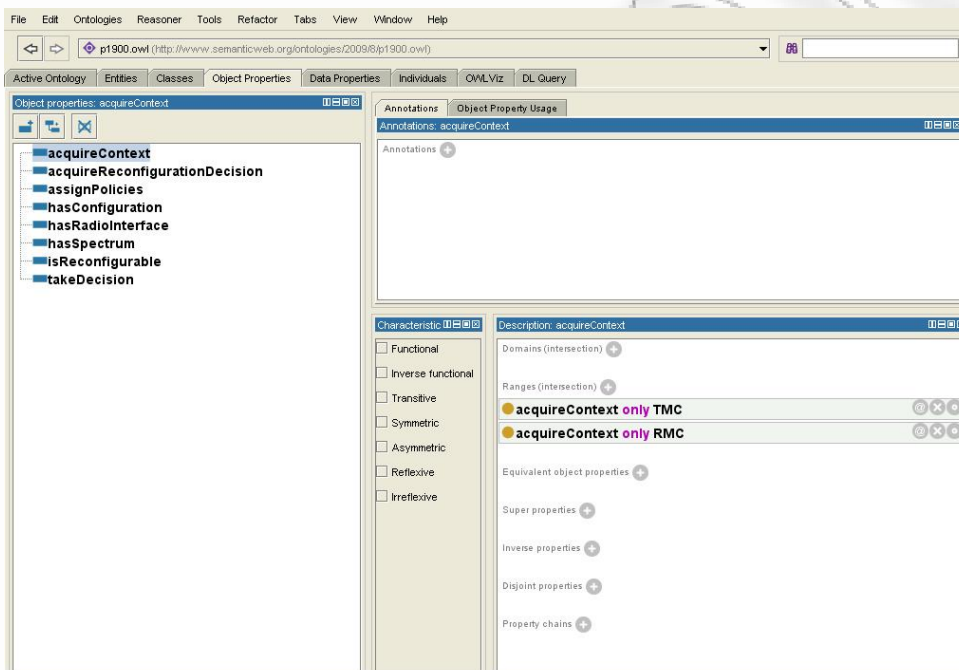


Εικόνα 5.12: ifTRC



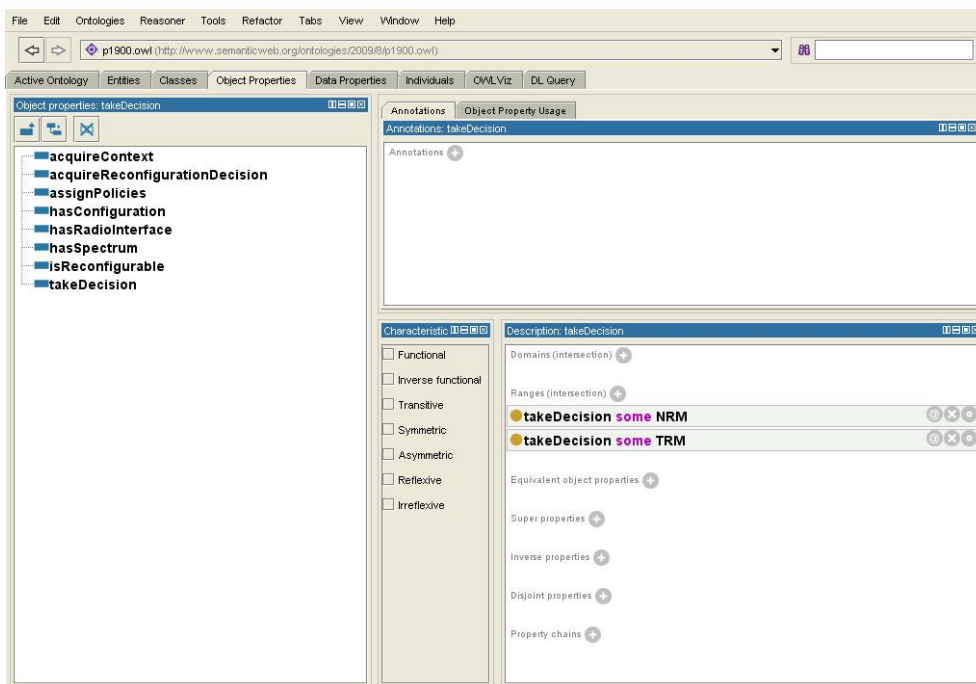
Εικόνα 5.13: MEDIA SAP

Ως υπο-κλάσεις της κλάσης `RDFProperty` ορίζονται οι κλάσεις των μηνυμάτων που ανταλλάσσονται μεταξύ των κλάσεων-οντοτήτων. Τέτοιες είναι οι κλάσεις όπως `acquireContext`, `acquireReconfigurationDecision`, `assignPolicies`, `hasConfiguration`, `hasRadioInterface`, `hasSpectrum`, `isReconfigurable`, `takeDecision`. Στο εργαλείο ορίζονται επίσης ποιές κλάσεις – οντότητες καλούν τις υπο-κλάσεις της `RDFProperty`. Έτσι, η `acquireContext` καλείται από τις TMC, και RMC για την λήψη ενημέρωσης σχετικά με πληροφορίες για τη κατάσταση του εκάστοτε τερματικού ή του δικτύου RAN αντίστοιχα.



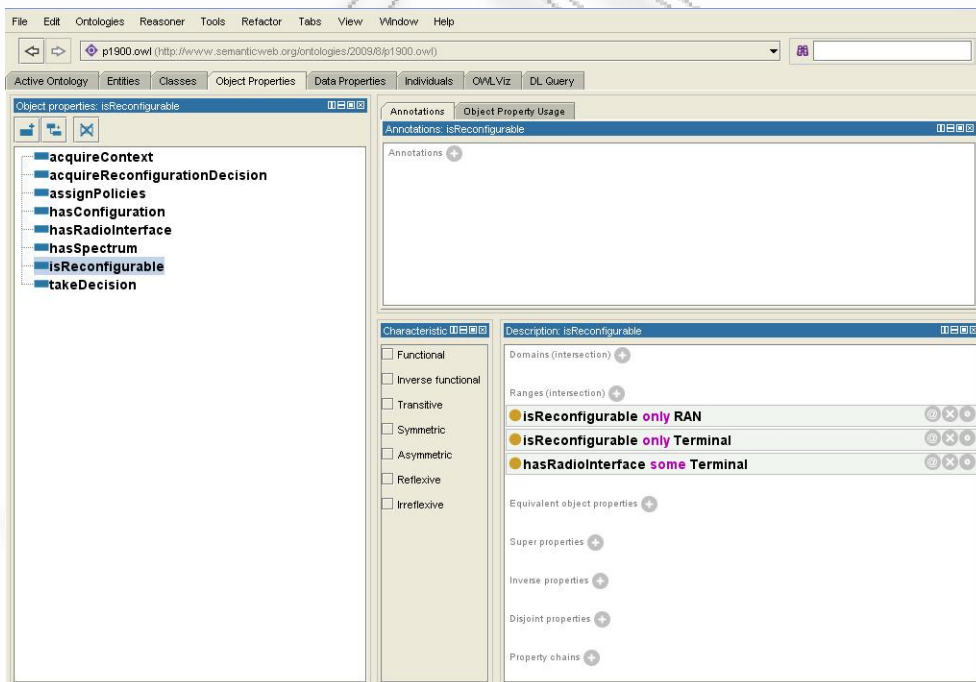
Εικόνα 5.14: propContext

Η `takeDecision` καλείται από τις NRM, TRM για την εκτέλεση αποφάσεων αναδιάρθρωσης του δικτύου ή των τερματικών με βάση τις πολιτικές τους.



Εικόνα 5.15: propDecision

Οι κλάσεις isReconfigurable, hasRadioInterface αφορούν ιδιότητες των οντοτήτων RAN, και Terminal αντίστοιχα για τον έλεγχο της κατάστασης παραμετροποίησης τους.





Εικόνα 5.16: propReconfig

Από τα παραπάνω στοιχεία φαίνεται ότι με τις επιλεχθείσες δομές του προτύπου P1900 το πρότυπο καθίσταται ελάχιστα πιο πολύπλοκο από το επιθυμητό.

Η απλοποίηση φάνηκε δυνατή με τη δημιουργία οντολογίας με τη χρήση του PROTÉGÉ, όπου τελικά δε χρειάστηκαν όλες οι δομές του προτύπου, αλλά ακόμα κι έτσι, όλες οι απαιτούμενες λειτουργίες επιτελούνται σωστά. Για αυτό το λόγο και θεωρείται καλύτερη η εν λόγω πρόταση.

Η επόμενη παράγραφος περιέχει στοιχεία για την αξιολόγηση τόσο του προτύπου, όσο και της πρότασης που προέκυψε από την ανάπτυξη της οντολογίας με το PROTÉGÉ.

#### 5.4 Ανάπτυξη οντολογίας σε αντικειμενοστραφή μοντέλο

Αφού παρουσιάστηκε η οπτικοποίηση της οντολογίας κάνοντας χρήση του εργαλείου Portege, κρίνεται σκόπιμη η περαιτέρω ανάπτυξη και παρουσίαση του βάσει της αντιστοίχησης του σε αντικειμενοστραφές μοντέλο. Ο λόγος για μια τέτοια ενέργεια είναι πως με την πραγματοποίηση της ανάπτυξης είναι δυνατόν να μεταφραστεί η γνώση σε οντότητες και με αυτό τον τρόπο να χρησιμοποιηθεί για περαιτέρω έρευνα σε πρακτικό επίπεδο κάνοντας χρήση αντικειμενοστραφούς προγραμματισμού.

Η δημιουργία μίας οντολογίας, πέραν του προσδιορισμού των απαιτήσεων, κρίνεται άκρως σημαντική καθώς αποτελεί έναν χρήσιμο τρόπο μετάφρασης και ενσωμάτωσης της γνώσης εντός καλά ορισμένων οντοτήτων. Σκόπιμο, βέβαια, κρίνεται να τονιστεί ότι μία οντολογία αποθηκεύεται σε μέσα τα οποία δεν μπορούν να χρησιμοποιηθούν από όλες τις γλώσσες προγραμματισμού (κυρίως τις διαδικαστικές), με την τρέχουσα να χρησιμοποιείται σχεδόν αποκλειστικά από αντικειμενοστραφείς γλώσσες

Το επόμενο βήμα, μετά την δημιουργία της οντολογίας, είναι η αντιστοίχιση αυτής σε αντικειμενοστραφή μέσα. Η προσέγγισή μας, λοιπόν, έγκειται στον προσδιορισμό των αντικειμένων και των μεταξύ τους σχέσεων, καθώς και στη χρήση της γλώσσας προγραμματισμού JAVA (και των χαρακτηριστικών της όπως η διασυνδεσιμότητα, η υποκατηγοριοποίηση και η συνάθροιση) με στόχο να δημιουργήσουμε μία επαναχρησιμοποιήσιμη ομάδα κλάσεων βασισμένες στο το Stanford OWL API. Η ομάδα αυτή θα είναι ό,τι κοντινότερο στην αντικειμενοστραφή αντιστοίχιση της οντολογίας του P1900 που θα μπορεί να χρησιμοποιηθεί από οποιαδήποτε εφαρμογή και να κάνει χρήση της προηγούμενης γνώσης προκειμένου να βελτιστοποιήσει τη διαχείριση και την διαλειτουργικότητα των δικτύων υψηλών ταχυτήτων.

#### **5.4.1 Ανάλυση ανάπτυξης**

Η αντιστοίχιση επιτεύχθηκε με την εκτέλεση ενός ενδιάμεσου βήματος στο οποίο κατασκευάστηκε ένα αντικειμενοστραφές μοντέλο. Ας δούμε, όμως, αρχικά, από ποια στοιχεία αποτελείται μία οντολογία:

1. Από τις έννοιες (όπως αντικείμενα με χαρακτηριστικά) που χρησιμοποιούνται για την περιγραφή του τομέα, και οι οποίες μπορούν εύκολα να διαμοιραστούν ανάμεσα σε εκείνους που δουλεύουν εντός του τελευταίου
2. Από τις σχέσεις και τους περιορισμούς που διέπουν τις έννοιες αυτές, και
3. Από τις συμπεριφορές και τους επιχειρησιακούς κανόνες οι οποίοι περιγράφουν τον τρόπο με τον οποίο μπορούν να αλληλεπιδρούν οι γενικές έννοιες εντός του τομέα.

Στο σημείο αυτό, αξίζει να τονιστεί ότι ένα Αντικειμενοστραφές Μοντέλο μπορεί να χρησιμοποιηθεί σαν μέσο για την αναπαράσταση της γνώσης ενός τομέα, κατά τη διάρκεια της δημιουργίας ενός λογισμικού. Ωστόσο, μία τέτοια θεώρηση δεν μπορεί να λειτουργήσει πλήρως σαν μία οντολογία. Ο βασικότερος λόγος για το τελευταίο, απλός αν και όχι

προφανής, έγκειται στο ότι αν και ένα Αντικειμενοστραφές Μοντέλο είναι ικανό να αναπαραστήσει τις έννοιες, τις σχέσεις και τους περιορισμούς των τελευταίων (ως ένα βαθμό), η τεχνολογία δεν έχει φτάσει στην επαρκή ωριμότητα προκειμένου να περιγράψει με πλήρη τρόπο τις συμπεριφορές (όπως τους επιχειρησιακούς κανόνες) που συσχετίζονται με τις έννοιες. Συνεπώς, μόνο ένα Αντικειμενοστραφές Μοντέλο ως μέσο αναπαράστασης της γνώσης, δεν είναι επαρκές. Με πιο απλά λόγια, ένα Αντικειμενοστραφές Μοντέλο δεν μπορεί να θεωρηθεί ισοδύναμο με μία οντολογία.

Επιπροσθέτως, αξίζει να αναφερθεί ότι τα Αντικειμενοστραφή Μοντέλα και οι οντολογίες έχουν διαφορετικά πεδία εκτέλεσης. Μία οντολογία, λοιπόν, κατασκευάζεται για τη σύλληψη, τη διαμοίραση και την επαναχρησιμοποίηση της γνώσης του τομέα, ενώ μπορεί να εφαρμοστεί σε όσες περιοχές έχουν ανάγκη από τη γνώση αυτή. Από την άλλη το Αντικειμενοστραφές Μοντέλο αποτελεί συχνά μία μερική εφαρμογή μίας οντολογίας, με την έννοια ότι παρέχει μόνο ένα υποσύνολο των συνολικών εννοιών και σχέσεων της τελευταίας. (Εντούτοις, να τονίσουμε στο σημείο αυτό, ότι στην περίπτωση που οι τελικοί χρήστες είναι συστήματα λογισμικού, τότε μία οντολογία δύναται να αναπαρασταθεί από ένα Αντικειμενοστραφές Μοντέλο μαζί με την επιχειρησιακή λογική που περιγράφει τη συμπεριφορά των αντικειμένων όπως αυτή προσδιορίζεται από το μοντέλο (με τα τελευταία πιθανώς υλοποιημένα ως πράκτορες λογισμικού)).

Όπως έχει ήδη αναφερθεί, κατά τη διαδικασία της μοντελοποίησης, κρίνεται σκόπιμη η εκτέλεση ενός ενδιάμεσου βήματος κατά το οποίο κατασκευάστηκε ένα αντικειμενοστραφές μοντέλο. Το αποτέλεσμα του βήματος αυτού ήταν η δημιουργία ενός μοντέλου της οντολογίας εντός ενός αντικειμενοστραφούς πλαισίου εφαρμογής.

Η πλειονότητα των αρχών αντιστοίχισης είναι ανεξάρτητη από την χρησιμοποιούμενη γλώσσα προγραμματισμού. Κάποιες διαφοροποιήσεις μπορούν να παρουσιαστούν, ανάλογα από την υποστήριξη πολλαπλής κληρονομικότητας. Το πρόβλημα της πολλαπλής κληρονομικότητας επιλύεται μέσω της χρήσης διεπαφών, στις περιπτώσεις που η δοθείσα αντικειμενοστραφής γλώσσα δεν το υποστηρίζει.

Μία οντολογία είναι, μπορούμε να πούμε, η καταγραφή ενός τομέα. Αποτελείται από έννοιες, τις μεταξύ τους σχέσεις και τους περιορισμούς από τους οποίους διέπονται. Περιγράφεται από γλώσσες οντολογίας, όπως εκείνη της OWL, ενώ παράλληλα θα μπορούσε να θεωρηθεί και σαν ένα μοντέλο δεδομένων.

Οντολογικές και αντικειμενοστραφείς αναπαραστάσεις διαφέρουν ως προς την εκφραστικότητά τους, Πιο συγκεκριμένα, οι οντολογίες έχουν υψηλότερη εκφραστικότητα. Για παράδειγμα, στις οντολογίες, τα μέλη μίας οντότητας ως προς την κλάση, μπορούν να αναπαρασταθούν μέσω περιορισμών, ενώ στο αντικειμενοστραφές μοντέλο κάτι τέτοιο θα ήταν αδύνατο. Επιπρόσθετα, με την OWL είναι δυνατός ο προσδιορισμός ιεραρχίας στα χαρακτηριστικά, κάτι που δε γίνεται εντός αντικειμενοστραφών περιβαλλόντων.

Κατά μία έννοια, η αντικειμενοστραφής απεικόνιση είναι όμοια με την οντολογική. Στην αντικειμενοστραφή προσέγγιση έχουμε να κάνουμε με αντικείμενα και τη συμπεριφορά αυτών. Ανάμεσά τους υπάρχουν διάφορες σχέσεις, όμοιες με τις οντολογικές κλάσεις. Έτσι, στην περίπτωση που χρειαστεί απλά η αποθήκευση των τιμών των χαρακτηριστικών κλάσεων οντολογίας, δε χρειάζεται να κάνουμε χρήση της περίπλοκης οντολογικής απεικόνισης, παρά μόνο να χρησιμοποιήσουμε την αντίστοιχη αντικειμενοστραφή.

#### **5.4.2 Παρουσίαση βασικών λειτουργιών**

Η κλάση OntoMapper παρέχει τις βασικές λειτουργίες για τη δημιουργία μίας οντολογίας ενός τομέα ευφυών δικτύων. Πιο συγκεκριμένα, κατασκευάζει την Java αντιστοιχία του Protege, για δοθέν OWL μοντέλο, Java Interface και όνομα, δίνοντας παράλληλα τις κατάλληλες μεθόδους για την δημιουργία, κλήση και χειρισμό των διαφόρων οντοτήτων του τομέα. Παράλληλα, η κατασκευή των ανωτέρω γίνεται με κατάλληλες κλήσεις μέσω URI για τη λήψη των οντοτήτων της οντολογίας.

Όσον αφορά το τμήμα των οντοτήτων, όπως το Cognitive Wireless Network, ο Network Configuration Manager και άλλα, υλοποιούνται οι παρακάτω μέθοδοι :

- CWN createCWN(String name)

Δημιουργεί ένα νέο CWN με δοθέν όνομα επιστρέφοντας ένα νέο Implementation του δικτύου, για συγκεκριμένο OWL μοντέλο. Στην περίπτωση που το όνομα δε δοθεί, αναζητείται και χρησιμοποιείται κάποια από τις ελεύθερες ονομασίες των resources.

- CWN getCWN(String name)

Χρησιμοποιείται για τη λήψη ενός ήδη κατασκευασμένου CWN. Στην περίπτωση που τα resources είναι κενά, δεν επιστρέφει τίποτα, ενώ στην αντίθετη περίπτωση είτε επιστρέφει το ζητούμενο CWN είτε κατασκευάζει ένα νέο, με δοθέν όνομα, εάν το επιθυμητό CWN δεν ευρεθεί.

- Collection<CWN> getAllCWNInstances()

Επιστρέφει μία συλλογή των υπαρχόντων CWNs

- Collection<CWN> getAllCWNInstances(boolean transitive)

Επιστρέφει όλα τα CWN τα οποία συνδέονται (ή αντίστοιχα ΔΕ συνδέονται, αν transitive = false) με κάποιο κοινό χαρακτηριστικό, όπως "όλα τα CWN της Χ περιοχής"

Εάν και παραπάνω παρουσιάστηκαν οι μέθοδοι για την κατασκευή και τη λήψη των CWN οντοτήτων, οι ίδιες (με διαφορετική ονομασία, όπως createNRM(), createRAT κ.ο.κ.) υλοποιούνται και για τις οντότητες:

- ran (Radio Access Network)

- rmc (RAN Measurement Controller)
- rrc (RAN Reconfiguration Controller)
- sap (Service Access point)
- tmc (Terminal Measurement Controller)
- trc (Terminal Reconfiguration Controller)
- trm (Terminal Reconfiguration Manager)
- nrm (Network Reconfiguration Manager)
- osm (Operator Spectrum Manager)
- qos (Quality of Service)
- terminal (Οντότητα που ορίζει το τερματικό)
- context (Οντότητα που ορίζει το περιβάλλον)
- if\_NRM\_OSM (Interface για την επικοινωνία μεταξύ Network Reconfiguration Manager και Operator Spectrum Manager)
- if\_NRM\_RMC (Interface για την επικοινωνία μεταξύ Network Reconfiguration Manager και RAN Measurement Controller)
- if\_NRM\_RRC (Interface για την επικοινωνία μεταξύ Network Reconfiguration Manager και RAN Reconfiguration Controller)
- if\_NRM\_TRM (Interface για την επικοινωνία μεταξύ Network Reconfiguration Manager και Terminal Reconfiguration Manager)
- if\_TRM\_TMC (Interface για την επικοινωνία μεταξύ Terminal Reconfiguration Manager και Terminal Measurement Controller)

- if\_TRM\_TRCC (Interface για την επικοινωνία μεταξύ Terminal Reconfiguration Manager και Terminal Reconfiguration Controller)
- interface (Οντότητα που ορίζει το interface)
- location (Οντότητα που αφορά την τοποθεσία)
- measurement (Οντότητα που ορίζει την μέτρηση)
- rCFG\_MEDIA\_SAP (Οντότητα που ορίζει την υπηρεσία του reconfiguration Media SAP)
- rCFG\_MNG\_SAP (Οντότητα που ορίζει την υπηρεσία του reconfiguration Management SAP)
- rCFG\_TR\_SAP (Οντότητα που ορίζει την υπηρεσία του Transport SAP)
- terminal capability (Οντότητα που ορίζει τις δυνατότητες του τερματικού)
- user preference (Οντότητα που ορίζει τις προτιμήσεις του χρήστη)

Ομοίως, για τη λήψη των συνθηκών που επικρατούν στο εκάστοτε δίκτυο, έχουν υλοποιηθεί οι παρακάτω μέθοδοι:

- RDFProperty getAcquireContextProperty()

Λαμβάνει το context το οποίο έχει να κάνει με τις συνθήκες κίνησης και δικτύου, όπως: αριθμός ενεργών συνεδριών, ζητούμενες υπηρεσίες ανά συνεδρία, διαθέσιμα φέροντα και άλλα. Παράλληλα, παρέχει πληροφορία που αφορά τα κανάλια, όπως ο σηματοθορυβικός λόγος

- RDFProperty getAcquireReconfigurationDecisionProperty()

Παρέχει την πληροφορία πάνω στα εκάστοτε επιλεγθέντα reconfigurations του infrastructure ή/και των τερματικών

- RDFProperty getAssignPoliciesProperty()

Επιστρέφει την κατάλληλη πολιτική που πρέπει να εφαρμόσουν τα τερματικά.

Ακολούθως έχουν υλοποιηθεί και οι μέθοδοι που ακολουθούν, με σκοπό να ορίσουν τις ιδιότητες:

- RDFProperty getHasConfigurationProperty(), η οποία αφορά την ιδιότητα της παραμετροποίησης

- RDFProperty getHasRadioInterfaceProperty(), η οποία αφορά την ιδιότητα της ραδιοδιεπαφής

- RDFProperty getHasSpectrumProperty(), η οποία αφορά την ιδιότητα του φάσματος

- RDFProperty getIsReconfigurableProperty(), η οποία αφορά την ιδιότητα της αναδιάρθρωσης

- RDFProperty getTakeDecisionProperty(), η οποία αφορά την ιδιότητα της λήψης απόφασης

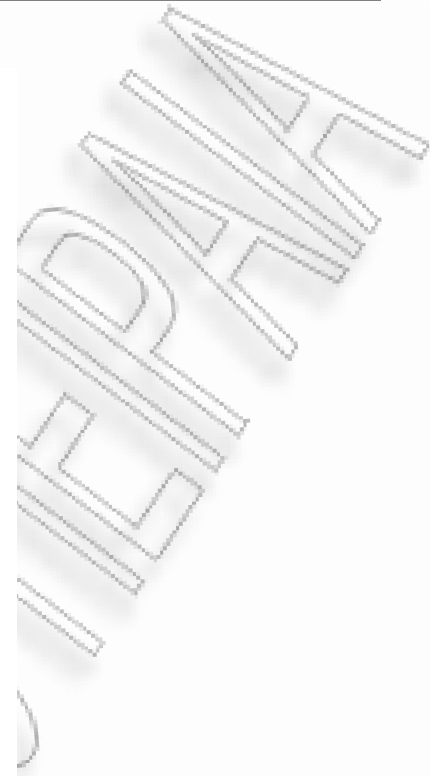
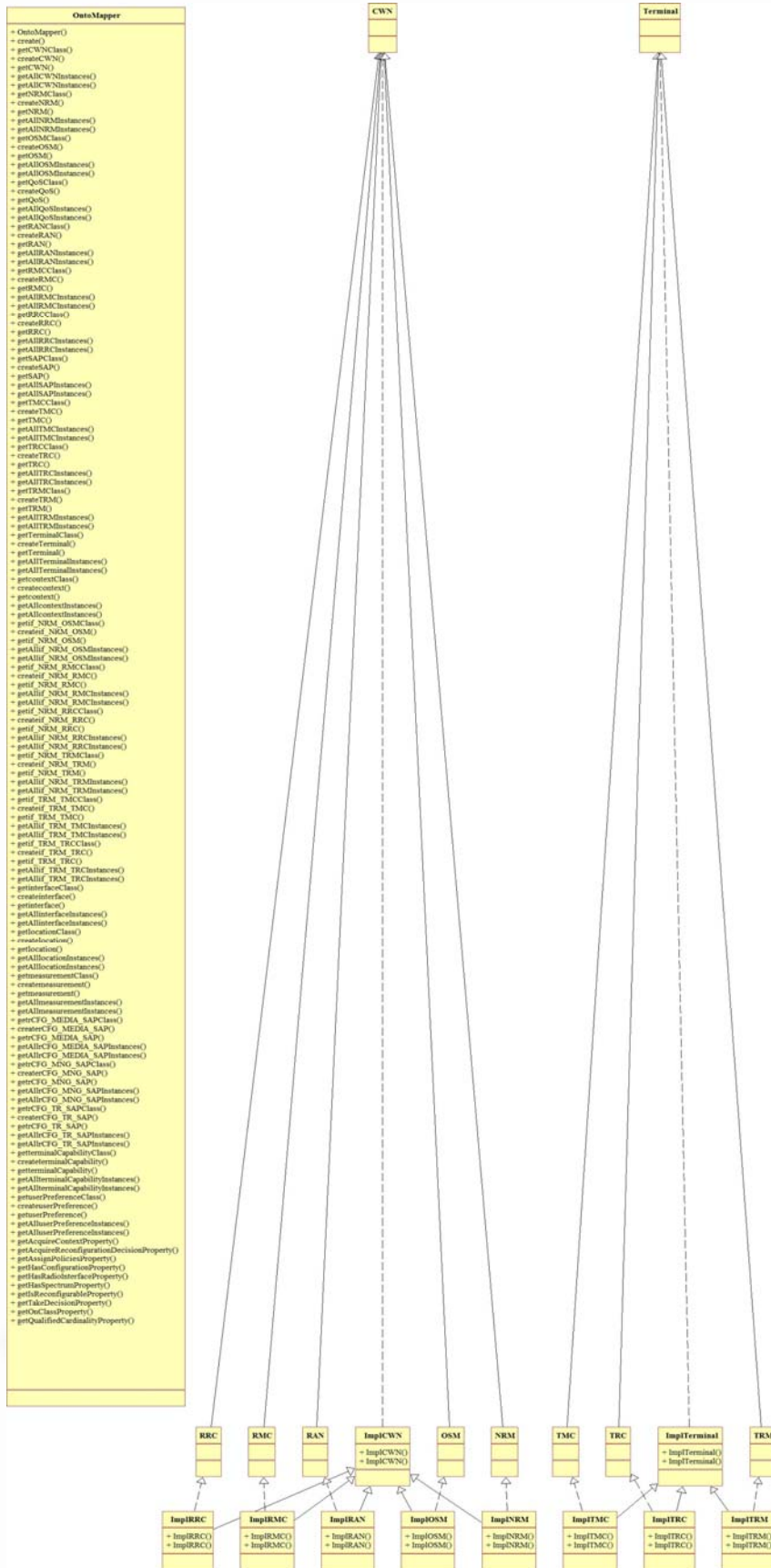
- RDFProperty getOnClassProperty(), η οποία αφορά την ιδιότητα της επιλογής κλάσης

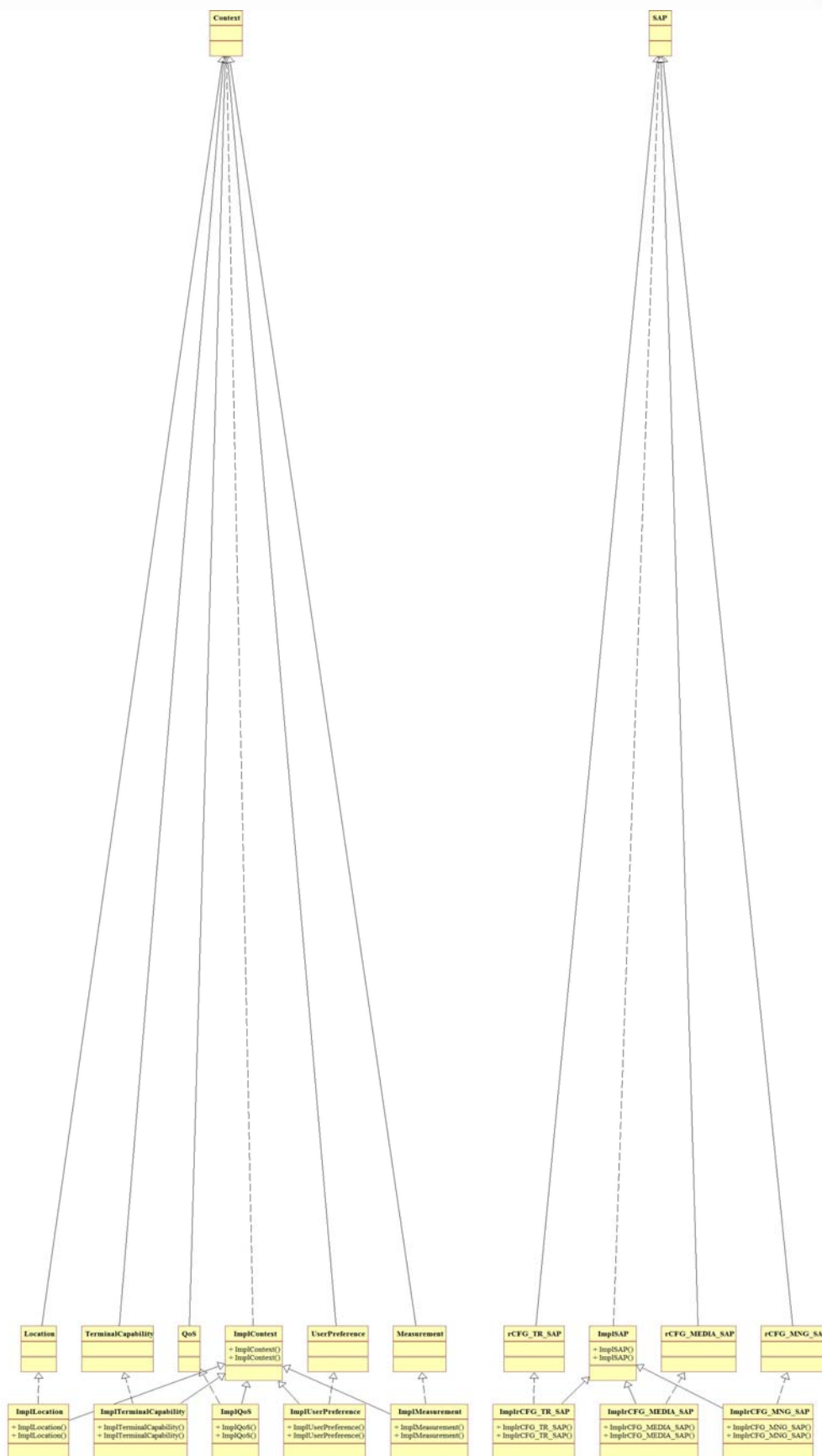
- RDFProperty getQualifiedCardinalityProperty(), η οποία αφορά την ιδιότητα της πληθικότητας γνωρισμάτων

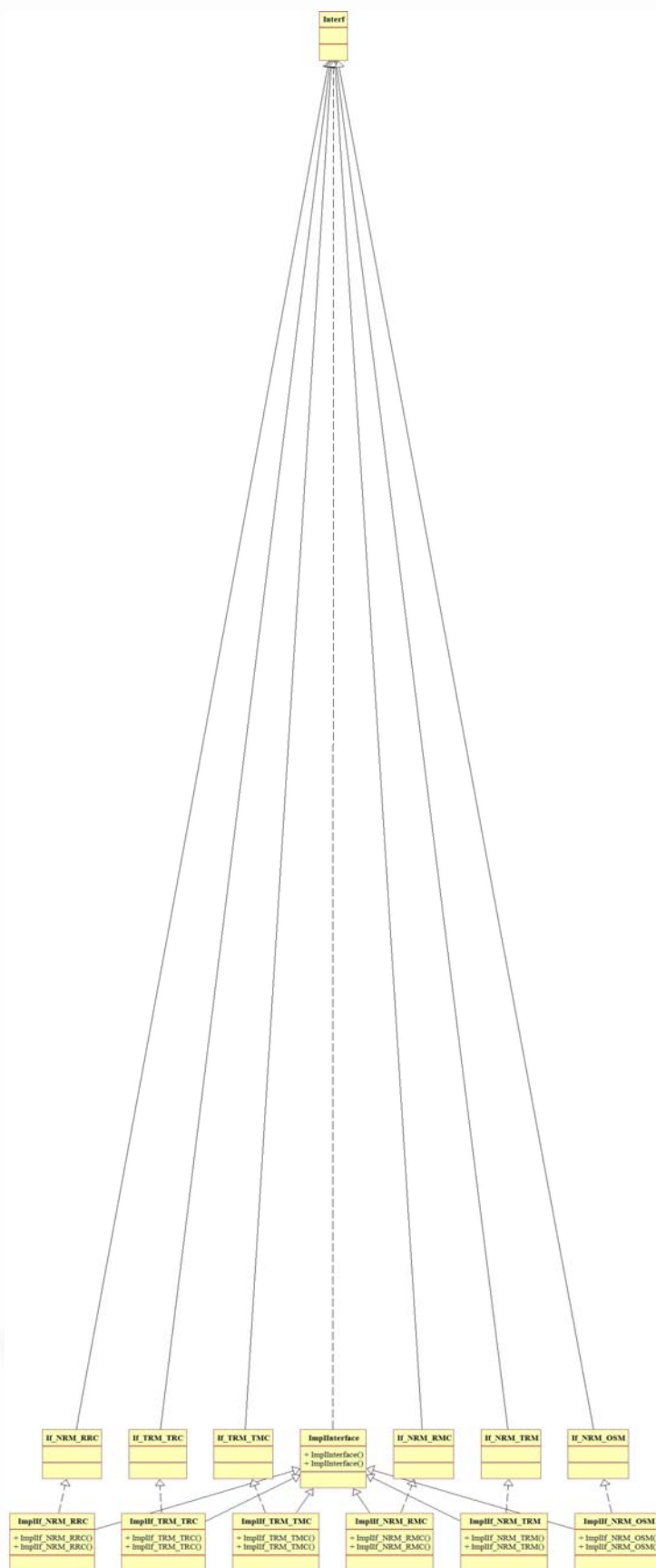


### **5.4.3 Οπτική παρουσίαση του αντικειμενοστραφούς μοντέλου**

Στην συνέχεια ακολουθεί η οπτική παρουσίαση του αντικειμενοστραφούς μοντέλου το οποίο περιγράφηκε έτσι ώστε να είναι εμφανή τόσο τα αντικείμενα και οι υλοποιημένες μέθοδοι, όσο και οι σχέσεις μεταξύ τους.





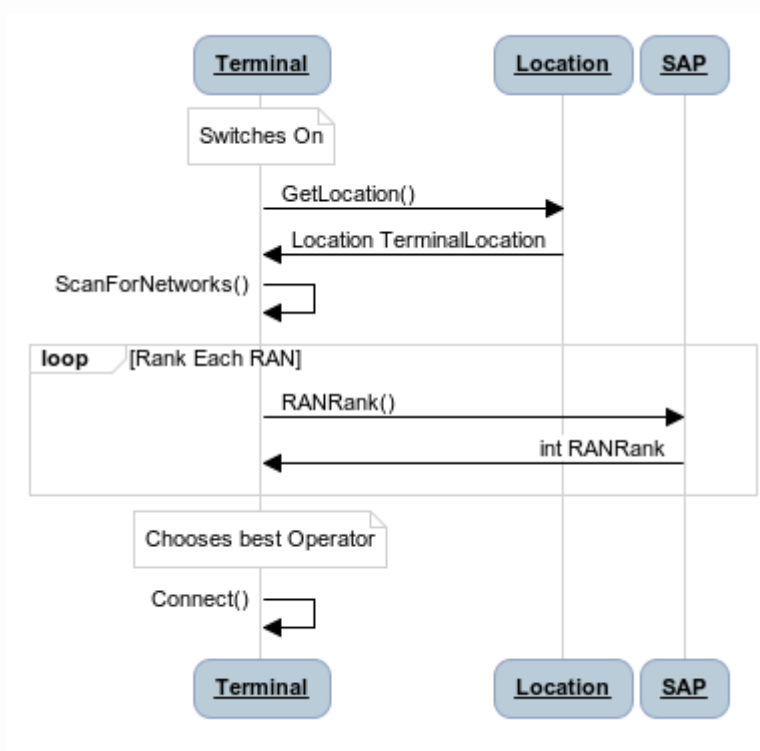


#### **5.4.4 Case Studies**

Αφού έχουν παρατεθεί αναλυτικά οι κλάσεις και οι μέθοδοι κρίνεται σκόπιμο να παρουσιαστούν κάποια case studies για την περαιτέρω κατανόηση του τρόπου λειτουργίας του πρωτοκόλλου. Ετσι θα είναι εφικτή η μελέτη της ανταλλαγής των μηνυμάτων εντός του δικτύου κατά κέραια σενάρια όπως η σύνδεση του τερματικού στο δίκτυο ή η αναδιάρθρωση. Για αυτό το λόγο θα παρουσιαστούν 3 case studies. Κατά το πρώτο, πραγματοποιείται η ευφυής αναζήτηση και σύνδεση ενός γνωσιακού, αναδιαρθρώσιμου τερματικού σε δίκτυο. Στο δεύτερο παρουσιάζεται η μεταφορά πληροφορίας που αφορά στο Context ανάμεσα στις οντότητες δικτύου και τερματικών προκειμένου να ληφθεί η απόφαση για τον τρόπο αναδιάρθρωσής τους σε ένα ετερογενές περιβάλλον. Τέλος, η τρίτη μελέτη περίπτωσης αφορά στην αναδιάρθρωση των δικτύων και των τερματικών, η οποία ακολουθεί τη διαδικασία μεταφοράς δεδομένων Context και κατασκευής του βέλτιστου Policy για το εκάστοτε περιβάλλον εκτέλεσης.

#### **Terminal Connection**

Έστω ότι σε μια τυχαία γεωγραφική περιοχή ένα γνωσιακό, αναδιαρθρούμενο τερματικό εκκινεί τη λειτουργία του. Ανοίγει επικοινωνία με την κλάση Location η οποία του επιστρέφει την τοποθεσία του και εν συνεχεία ανιχνεύει τα διαθέσιμα δίκτυα της περιοχής. Μόλις ανακτήσει τη λίστα των δικτύων που λειτουργούν γύρω του, επικοινωνεί με την κλάση SAP για να λάβει την κατάταξη κάθε δικτύου όσον αφορά στις εφαρμογές που έχει επιλέξει το τερματικό.



Λαμβάνοντας το βαθμό για το κάθε δίκτυο, το τερματικό συγκρίνει τα τελικά αποτελέσματα υπολογίζοντας το επίπεδο ποιότητας υπηρεσίας που προσφέρει το κάθε ένα από αυτά, μαζί με τα επίπεδα ποιότητας που έχει ορίσει ο χρήστης στο προφίλ του καθώς και με τις λοιπές προτιμήσεις του, όπως η προτιμητέα τεχνολογία, προτιμητέος πάροχος, προτεραιότητα επικοινωνίας και άλλα. Εκτελώντας τους κατάλληλους υπολογισμούς με βάση τις συλλεχθείσες και αποθηκευμένες πληροφορίες, εξάγει τον βέλτιστο πάροχο εκ της λίστας των διαθέσιμων, στον οποίο και συνδέεται.

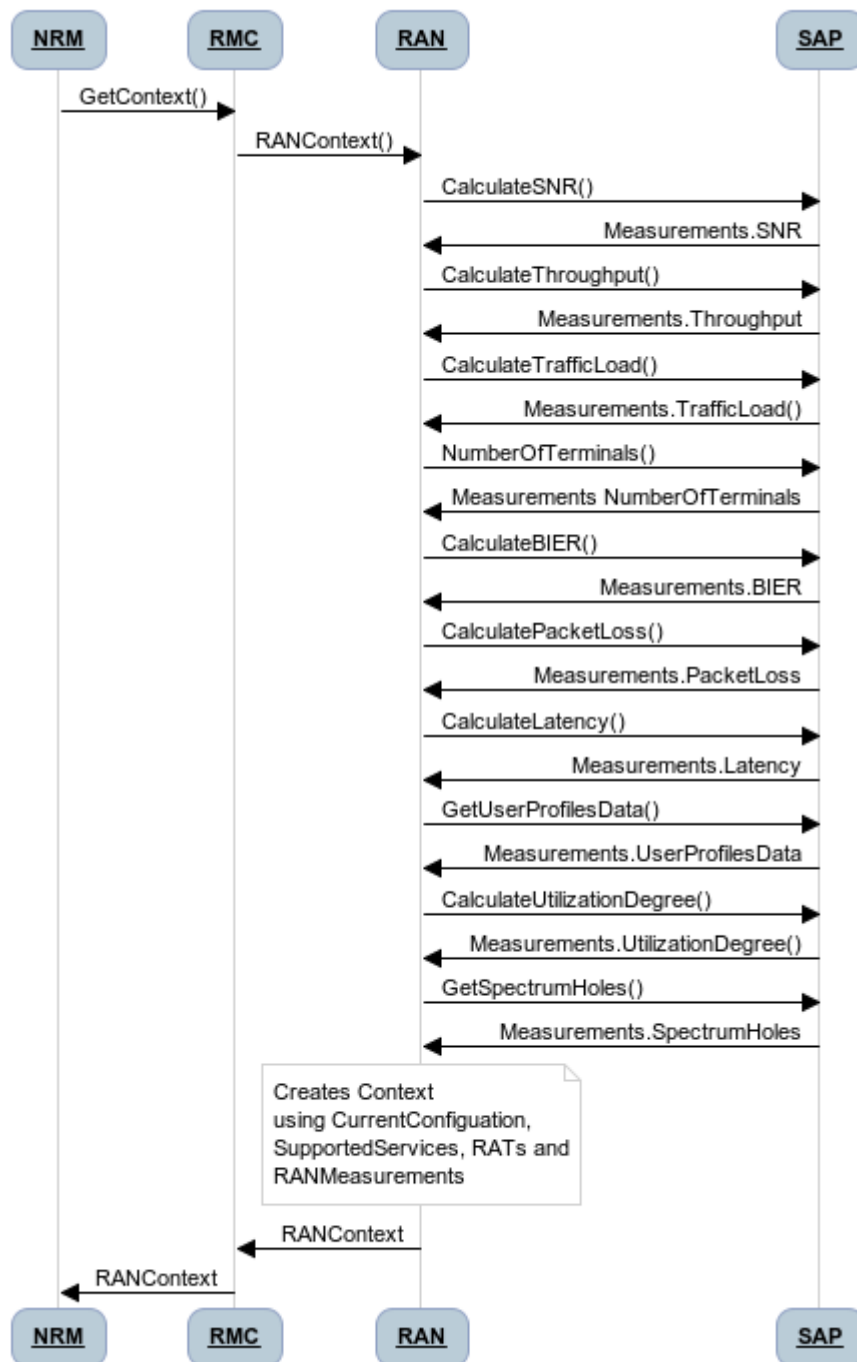
### ***RAN Context Exchange***

Κατά τη λειτουργία ενός δικτύου, η κλάση NRM μπορεί να κρίνει απαραίτητη την αναδιάρθρωση του τελευταίου, για παράδειγμα λόγω της υπερφόρτωσης κάποιου καναλιού. Εκκινώντας τη διαδικασία της αναδιάρθρωσης, αποστέλλει στο RMC του μία αίτηση λήψης του γενικού πλαισίου του δικτύου (Context). Το RMC, εν συνεχεία, επικοινωνεί με τα RANs τα οποία ελέγχει και τους ζητάει να του αποστείλουν τα αντίστοιχα Contextes τους. Προκειμένου τα τελευταία να μην εκτελούν δύσκολους υπολογισμούς οι οποίοι ενδεχομένως θα απαιτούσαν σημαντική υπολογιστική ισχύ όπως επίσης και μεγάλο όγκο δεδομένων, απαιτήσεις που με την σειρά τους επιρεάζουν την

αυτονομία του τερματικού, επικοινωνούν με τη σειρά τους με την κλάση SAP η οποία αναλαμβάνει να εκτελέσει τις αντίστοιχες διαδικασίες και να τα τροφοδοτήσει με τα τελικά αποτελέσματα.

Έτσι, κάθε RAN ζητάει από την SAP τις διάφορες μετρήσεις που το αφορούν (SNR, Throughput, Traffic Load, Αριθμός Τερματικών στο δίκτυο, BIER, Packet Loss, Καθυστερήσεις, User Profiles Data, Βαθμός χρησιμοποίησης, Spectrum Holes). Κρίνεται σκόπιμη η επισήμανση, στο σημείο αυτό, ότι στην εικόνα που ακολουθεί το RAN φαίνεται να αποστέλλει αιτήσεις για κάθε μία τιμή μέτρησης στο SAP και αναμένει την απάντηση του τελευταίου. Στην πραγματικότητα σύμφωνα με το εξεταζόμενο πρωτόκολλο, το RAN αποστέλλει μία αίτηση στο SAP, το οποίο απαντά με ένα αντικείμενο της κλάσης Measurements με τις αντίστοιχες ιδιότητες του τελευταίου να έχουν τις τιμές που αφορούν στο συγκεκριμένο RAN. Η sequential επικοινωνία που απεικονίζεται παρακάτω έχει γίνει καθαρά για λόγους απεικόνισης της λεπτομέρειας των ανταλλασσόμενων πληροφοριών.

Όταν, λοιπόν, το εκάστοτε RAN λάβει τις μετρήσεις του από το SAP, δημιουργεί ένα πακέτο στο οποίο εκχωρεί τις τελευταίες μαζί με πληροφορίες οι οποίες αφορούν στο τρέχον Configuration του, στα Services που δύναται να υποστηρίξει καθώς και στις τεχνολογίες τις οποίες μπορεί να χρησιμοποιήσει. Το πακέτο αυτό, με τη μορφή Context αποστέλλεται στο RMC από το οποίο ελέγχεται, με το τελευταίο να το προωθεί στον αντίστοιχο NRM.





## ***Terminal Context Exchange***

Έχοντας λάβει το Context από το RRC, όπως αναλύθηκε και παραπάνω, το NRM προχωρά στη συλλογή στοιχείων όσον αφορά στα τερματικά. Αποστέλλει, λοιπόν, αίτηση για λήψη Context στο αντίστοιχο TRM, το οποίο με τη σειρά του την προωθεί στα τερματικά τα οποία βρίσκονται κάτω από τον έλεγχό του.

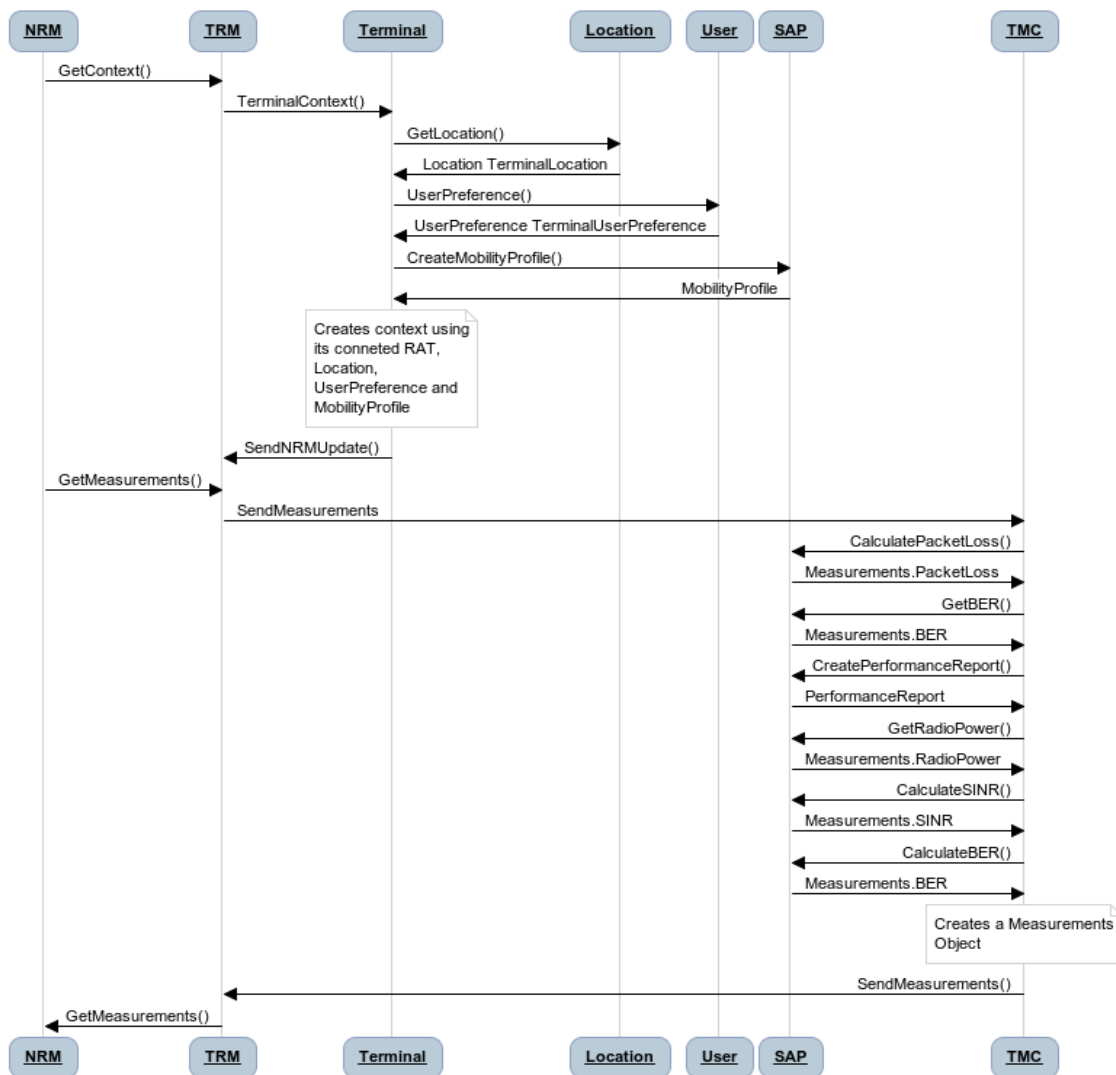
Κάθε τερματικό εφ' όσον είναι απαραίτητο, επικοινωνεί με την κλάση Location για να λάβει την τρέχουσα τοποθεσία του. Κατόπιν, συλλέγει πληροφορίες οι οποίες αφορούν στις προτιμήσεις χρήστη και έχουν να κάνουν με τον πάροχο, την τεχνολογία, τις υπηρεσίες, τις εφαρμογές καθώς και τα επιθυμητά και πραγματικά επίπεδα ποιότητας ανά υπηρεσία και εφαρμογή. Έχοντας λάβει, λοιπόν, από την κλάση User την απαραίτητη πληροφορία, επικοινωνεί με την κλάση SAP προκειμένου να της ζητήσει την αποστολή του προφίλ κινητικότητας του τερματικού. Η SAP εκτελεί τους απαιτούμενους υπολογισμούς, δημιουργεί το Mobility Profile πακέτο και το αποστέλλει στον αιτούντα.

Το τελευταίο λοιπόν, έχοντας όλες τις πληροφορίες που αφορούν στο προφίλ του, δημιουργεί το Context πακέτο εισάγοντάς του τα παραπάνω, και το αποστέλλει το TRM από το οποίο ελέγχεται. Το TRM με τη σειρά του, θα το προωθήσει στο NRM.

Με τη λήψη των προφίλ των τερματικών της περιοχής, το NRM προχωρά στη ζήτηση των μετρήσεων που αφορούν στα τελευταία. Στο σημείο αυτό, θα εξηγηθεί ο λόγος της διπλής αίτησης πληροφοριών από το NRM στα Τερματικά, έναντι της μίας κατά την επικοινωνία του NRM με τα RANs. Το θέμα είναι ότι τα τερματικά, ανά τακτά χρονικά διαστήματα ή on demand, δύναται να αποστέλλουν τις πληροφορίες προφίλ τους στο NRM, κάτι που δεν κάνουν τα RANs. Ενώ, δηλαδή, τα RANs αποστέλλουν όλο το Context τους στο NRM, τα τερματικά μπορεί να αποστείλουν τμήματα αυτού (προφίλ και μετρήσεις), προκειμένου στις περιπτώσεις απλού update των RANs να ελαχιστοποιηθεί το απαιτούμενο πακέτο πληροφορίας και να μην αποσταλούν δεδομένα μετρήσεων ενώ δεν είναι απαραίτητα.

Συνεχίζοντας, λοιπόν, το NRM έχει λάβει το Context όσον αφορά στο προφίλ των τερματικών, και αιτείται των μετρήσεών τους. Αποστέλλει την αίτηση στο TMC που ελέγχει, το οποίο για ελαχιστοποίηση της απαιτούμενης υπολογιστικής ισχύος και μνήμης, επικοινωνεί με την οντότητα SAP η οποία υπολογίζει και απαντά με το ενημερωμένο αντικείμενο κλάσης Measurements που περιλαμβάνει τις τιμές Packet Loss, BER, Radio Power, SINR και BIER. Όπως προαναφέρθηκε, η sequential απεικόνιση της αίτησης και λήψης των μετρήσεων, γίνεται καθαρά για λόγους αποσαφήνισης και καλύτερης κατανόησης, ενώ στην πραγματικότητα εκτελείται με μία κλήση αίτησης και μία κλήση απάντησης.

Μαζί με τα παραπάνω, το TMC ζητά από την κλάση SAP να του αποστείλει μία αναφορά της επίδοσής των τερματικών. Το SAP συλλέγει τις απαιτούμενες πληροφορίες, δημιουργεί την αναφορά και την αποστέλλει στον αιτούντα ο οποίος προωθεί το αντικείμενο κλάσης Measurements στο NRM.

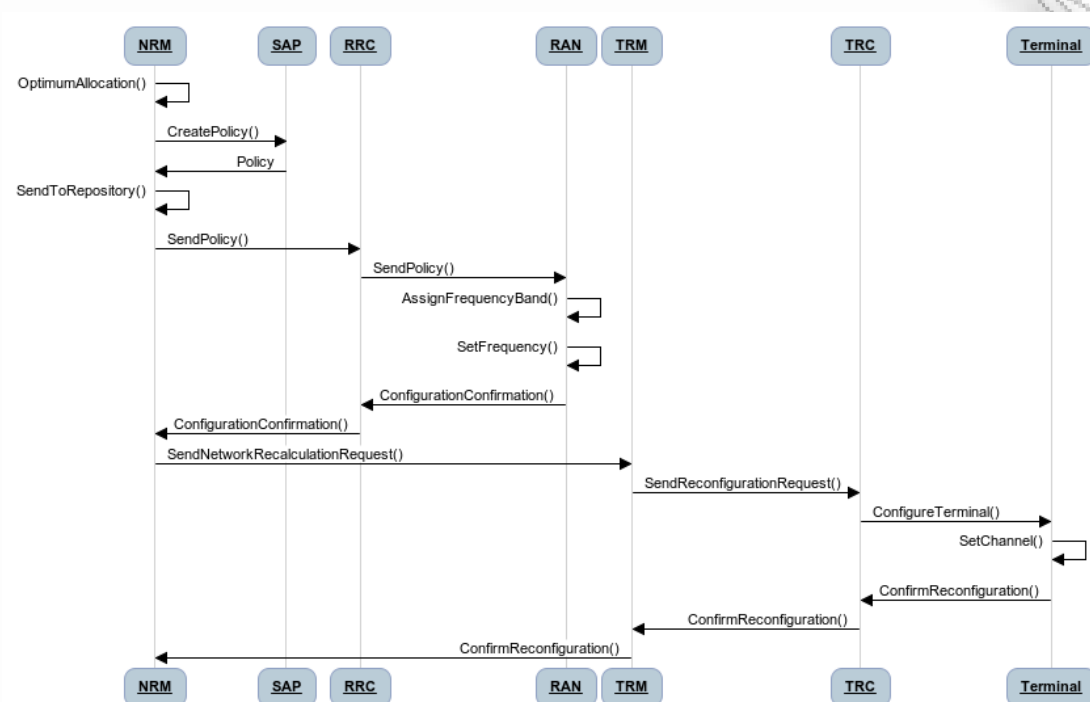


## **Reconfiguration**

Αφού έχει συλλέξει το Context Information τόσο από τα δίκτυα όσο και από τα τερματικά που βρίσκονται κάτω από τον έλεγχό του, το NRM υπολογίζει τη βέλτιστη κατανομή των πόρων και της κίνησης στην περιοχή του. Έχοντας καταλήξει στο καλύτερο δυνατό αποτέλεσμα, το αποστέλλει στην κλάση SAP προκειμένου η τελευταία να τη μετατρέψει στο ορθό πακέτο Policy το οποίο να είναι αναγνώσιμο από τις οντότητες της εκάστοτε τεχνολογίας. Η SAP του επιστρέφει την δημιουργημένη πολιτική, την οποία το NRM αποθηκεύει στο Repository του.

Έχοντας υπολογίσει τη βέλτιστη κατανομή και δημιουργήσει το αντίστοιχο Policy, σειρά έχει η ενημέρωση των οντοτήτων δικτύου προκειμένου τα τελευταία να εκτελέσουν τις απαιτούμενες λειτουργίες αναδιάρθρωσης. Έτσι, το NRM αποστέλλει το πακέτο Policy στο RRC το οποίο με τη σειρά του διανέμει στα RANs που ελέγχει. Τα τελευταία, ρυθμίζουν καταλλήλως τις μπάντες συχνοτήτων που χρησιμοποιούν και τις συχνότητες στις οποίες εκπέμπουν, και με το επιτυχές πέρας της διαδικασίας αποστέλλουν μήνυμα επιβεβαίωσης στο RRC. Το RRC, προωθεί την επιβεβαίωση αναδιάρθρωσης στο NRM.

Εν συνεχεία, το NRM επικοινωνεί με το TRM για να του αποστείλει αίτηση επανυπολογισμού δικτύου, την οποία το τελευταίο προωθεί στο TRC. Το TRC καλεί τα τερματικά να πραγματοποιήσουν αναδιάρθρωση. Στη βέλτιστη περίπτωση, κάποια από τα τερματικά που άκουγαν στο κανάλι που υπέστη συμφόρηση (που ήταν και ο λόγος για την εκκίνηση της όλης επικοινωνίας, όπως είδαμε στο δεύτερο Case Study), θα αλλάξουν σε ένα διαφορετικό, το οποίο να συμφωνεί τόσο με την αντικειμενική συνάρτηση των ιδίων όσο και με το Policy που αποφασίστηκε από το NRM. Με την επιτυχή εκτέλεση των απαιτούμενων βημάτων, θα αποστείλουν μήνυμα επιβεβαίωσης στο TRC, το οποίο θα προωθηθεί στο TRM και τέλος, στο NRM.



## 5.5 Αξιολόγηση οντολογίας και σύγκριση με το πρότυπο P1900

Η περιγραφείσα οντολογία εξυπηρετεί όλα τα σενάρια λειτουργίας δικτύων B3G. Συγκεκριμένα:

A) εάν ξεκινήσει η λειτουργία ενός δικτύου από το σταθμό βάσης, τότε αυτός στέλνει στη διαχειριστική του οντότητα (NRM) ένα μήνυμα βάσει της οντολογίας, το οποίο περιέχει το προφίλ του σταθμού βάσης και την αρχική του διάρθρωση (configuration) που πρέπει να είναι γνωστή στη διαχειριστική οντότητα του σταθμού βάσης, ώστε να πραγματοποιηθεί εν συνεχεία οποιαδήποτε σχετική διαχειριστική λειτουργία.

B) ομοίως, όταν ένας σταθμός βάσης έχει κάποιο πρόβλημα που οφείλεται είτε σε υπερφόρτωσή του είτε σε λειτουργική του αδυναμία, τότε ομοίως στέλνει σε οποιαδήποτε εμπλεκόμενη οντότητα διαχείρισης (ακόμα και στην TRM, όχι μόνο στην NRM) μήνυμα που περιγράφει το φορτίο του ή/και το πρόβλημά του.

Γ) όταν ένα τερματικό θέλει να ξεκινήσει μια υπηρεσία, η διαχειριστική του οντότητα TRM επικοινωνεί με την οντότητα NRM για την περιγραφή της υπηρεσίας και τη διαπραγμάτευση με τα διαθέσιμα δίκτυα / σταθμούς βάσης στην περιοχή.

Δ) όταν απαιτείται μια βελτιστοποίηση των πόρων του δικτύου, η οντότητα NRM στέλνει σχετικά μηνύματα τόσο στους σταθμούς βάσης όσο και στα εμπλεκόμενα τερματικά που περιέχουν τις αντίστοιχες κλάσεις της οντολογίας, όπως έχουν περιγραφεί ανωτέρω.

Φαίνεται λοιπόν ότι οποιοδήποτε πιθανό σενάριο λειτουργίας ενός δικτύου B3G καλύπτεται από την περιγραφείσα οντολογία. Οπωσδήποτε βέβαια, και το προαναφερθέν πρότυπο IEEE P1900 έχει ως σκοπό την πληρέστερη λειτουργική περιγραφή υπό τύπον οντολογίας όλων των δυνατών λειτουργιών ενός δικτύου B3G. Σε γενικές γραμμές είναι αναμφίβολη η τεράστια σημασία της ανάπτυξης προτύπων που να καλύπτουν μεγάλο τμήμα του υπό έρευνα κόσμου των τηλεπικοινωνιών. Τα πρότυπα συνιστούν σύνολα από οντολογίες, άρα κοινούς τρόπους αναφοράς σε συγκεκριμένα θέματα.

Το πρότυπο P1900.4 της IEEE παρουσιάζει αρκετό ενδιαφέρον όσο αφορά την ανάπτυξη ενός ολοκληρωμένου συνόλου από οντολογίες που αναφέρονται στη διαχείριση τερματικών και στοιχείων δικτύου σε δυναμικά αναδιαρθρούμενα, ετερογενή περιβάλλοντα. Δεδομένης της πληρότητας του εν λόγω προτύπου, θα προτεινόταν εύκολα τα κατωτέρω για την απλούστευσή του.

#### Συγκεκριμένα:

A) το παρόν πρότυπο διαθέτει μια πολύ γενική μορφή, που δεν του επιτρέπει να αναφέρεται με σαφήνεια στις λεπτομέρειες που σχετίζονται με την επικοινωνία των διαφόρων οντοτήτων. Αυτό σημαίνει ότι οι οντότητες έχουν ήδη οριστεί, αλλά ο τρόπος επικοινωνίας (interfaces) δεν έχει ακόμα οριστεί με τρόπο λεπτομερή. Αυτό σημαίνει ότι η πληροφορία που ανταλλάσσεται μεταξύ των οντοτήτων πρέπει να οριστεί με τρόπο σαφέστατο, που να ακολουθεί συγκεκριμένα πρότυπα, ούτως ώστε να αποτελεί από μόνη της μια οντολογία, ως ένας κοινός τύπος.

Β) το παρόν πρότυπο δεν αναφέρει με σαφήνεια τον τρόπο επικοινωνίας μεταξύ δικτύου και τερματικού. Αναμφίβολα η υποστήριξη κάθε τρόπου είναι επιθυμητή, αλλά πρέπει να οριστεί η δομή της πληροφορίας που θα μεταφέρεται από το δίκτυο στο τερματικό και αντίστροφα με τρόπο συγκεκριμένο, έτσι ώστε να αποτελεί σημείο αναφοράς για μελλοντικές μελέτες.

Γ) το παρόν πρότυπο μπορεί να απλοποιηθεί ίσως ακόμα περισσότερο, συγκεντρώνοντας κάποια use cases, έτσι ώστε να περιλάβει λιγότερες οντότητες τελικά, οι οποίες να είναι απαραίτητες για προτυποποίηση.

Δ) τέλος, το παρόν πρότυπο αναμφίβολα αποτελεί μια ολοκληρωμένη οντολογία στον τομέα των ασυρμάτων επικοινωνιών και του τρόπου με τον οποίο επικοινωνεί το τερματικό με το δίκτυο ακόμα και στο σημείο που βρίσκεται σήμερα. Έτσι, αποτελεί ένα εξαιρετικά θετικό βήμα προς την κατεύθυνση της ενοποίησης των ετερογενών περιβαλλόντων, τα οποία σίγουρα θα αποτελούν ένα όλο και πιο συχνό φαινόμενο στον κόσμο των τηλεπικοινωνιών.

Ε) Σε κάθε περίπτωση, τόσο το παρόν πρότυπο όσο και η οντολογία χρειάζεται να εμπλουτιστούν με μοντέλα μάθησης, τα οποία θα συνεισφέρουν στην περαιτέρω ανάπτυξη και χρήση του προτύπου σε περισσότερες εφαρμογές, με χρήση cognitive δικτύων.

Γενικά συμπεράσματα επί των οντολογιών και του τρόπου με τον οποίο θα μπορούσαν αυτές να προχωρήσουν στο μέλλον, παρατίθενται στο επόμενο κεφάλαιο.

## 6. Συμπεράσματα – Κατευθύνσεις

### 6.1 Σύνοψη και συμπεράσματα

Η εργασία αυτή προτείνει μία οντολογία για την μοντελοποίηση των πληροφοριών και των διαδικασιών που εμπλέκονται στη διαχείριση ετερογενών δικτύων και πιο συγκεκριμένα στα πλαίσια της εξέλιξης τους ως συνιστώσες ενός B3G δικτύου. Η οντολογία που προτείνεται θεωρείται συνέχεια του προτύπου P1900 του Ινστιτούτου IEEE με σκοπό την απλούστευση της υλοποίησης του. Για τη δημιουργία του προτεινόμενου μοντέλου πάρθηκαν όσες δομές και παράμετροι χρειάστηκαν από το μοντέλο πληροφορίας (information model) του προτύπου P1900 και ενσωματώθηκαν στο προτεινόμενο μοντέλο.

Δύο είναι οι στόχοι της προσπάθειας επέκτασης του P1900 προτύπου: α) μείωση της πολυπλοκότητας της μοντελοποίησης των βασικών εννοιών ως κλάσεις και υπο-κλάσεις, και β) τυποποίηση του λογισμικού που θα αντικατοπτρίζει την οντολογία ενός ενιαίου B3G δικτύου. Για το σκοπό αυτό προτάθηκε η αναπαράσταση και υλοποίηση της οντολογίας ως ένα αντικειμενοστραφές μοντέλο. Η διαδικασία αυτή δεν θα μπορούσε να επιτευχθεί χωρίς τη χρήση βασικών εργαλείων υλοποίησης. Επομένως, για το παραπάνω σκοπό χρησιμοποιήθηκε το Protégé 2000.

Αρχική προϋπόθεση για την υλοποίηση του αντικειμενοστραφούς μοντέλου της οντολογίας ήταν η αποσαφήνιση της αρχιτεκτονικής του συστήματος διαχείρισης του B3G δικτύου, όπως περιγράφηκε στην ενότητα (§4.4). Κατόπιν, ορίστηκε το διάγραμμα κλάσεων του αντικειμενοστραφούς μοντέλου της οντολογίας (Εικόνα 5-1) και αποτέλεσε τον οδηγό υλοποίησης του μοντέλου ως RDFSchema στο Protégé 2000.

Με τη συνδρομή του εργαλείου ορίστηκαν οι κλάσεις και οι μέθοδοι διαχείρισης των οντοτήτων διαχείρισης δικτύου (OSM, RMC, NRM, RRC), των οντοτήτων διαχείρισης του τερματικού (TMC,

TRM, TRC) καθώς και 6 διεπαφές επικοινωνίας μεταξύ των οντοτήτων (NRM – TRM, TRM – TRC, TRM - TMC, NRM – RRC, NRM – RMC, NRM – OSM) όπως και 3 «service access points» (transport SAP, reconfiguration and measurement SAP, management SAP).

Έτσι το αντικειμενοστραφές μοντέλο της οντολογίας που προκύπτει ουσιαστικά επαναχρησιμοποιεί βασικές κλάσεις του P1900, όπως Context, QoS, Spectrum Assignment, Reconfiguration, Policies, Terminals και με την υλοποίηση των 6 προαναφερθέντων διεπαφών υλοποιεί τα 3 βασικά use cases του προτύπου της IEEE 1900.4, δηλαδή:

1. Δυναμική ανάθεση φάσματος (dynamic spectrum assignment)
2. Δυναμική κοινή χρήση φάσματος (dynamic spectrum sharing) και
3. Κατανεμημένη βελτιστοποίηση χρήσης πόρων (distributed radio resource usage optimisation).

Η υλοποίηση του παραπάνω μοντέλου ως ένα κοινά αποδεκτό σύστημα διαχείρισης δυναμικά αναδιαρθρούμενων και ευφυών B3G δικτύων (Dynamic Self-organizing Network Planning and Management – DSNPM- εφαρμογή) στοχεύει στην εξασφάλιση της αποδοτικότερης διασυνεργασίας μεταξύ των διαφόρων RATs και των παρεχομένων υπηρεσιών τους έχοντας πλέον ένα κοινά αποδεκτό πληροφοριακό μοντέλο που μειώνει το κόστος επένδυσης σε μη τυποποιημένο λογισμικό, όπως και το κόστος υποδομής σε δίκτυα και τερματικά.

Οι διαφοροποιήσεις των εμπλεκόμενων δικτύων (πχ σε εύρος ζώνης, ακτίνα κάλυψης, καθυστέρηση, προτιμήσεις χρηστών, επίπεδο ασφάλειας, απαιτούμενη ισχύς και κόστος υπηρεσίας) παραμετροποιούνται στις σχετικές κλάσεις-οντότητες (πχ Context, QoS, Measurement, User Preferences). Κατόπιν, οι οντότητες διαχείρισης των εμπλεκόμενων δικτύων και τερματικών παρέχουν μηχανισμούς που καθιστούν πραγματοποιήσιμη την συνεχή ενημέρωση για τις συνθήκες στα εκάστοτα δίκτυα και τερματικά και την



αξιολόγηση τους σύμφωνα με δείκτες KPI και αλγορίθμους βελτιστοποίησης και αναδιάρθρωσης. Στη συνέχεια αποφασίζουν την δυναμική εκτέλεση πολιτικών και αποφάσεων για την αλλαγή παραμέτρων των πρωτοκόλλων μιας τεχνολογίας (πχ συχνότητα λειτουργίας) και της εν γένει διάρθρωσής της, όποτε οι εξωτερικές συνθήκες το επιβάλλουν.

Οι κλάσεις αυτές αποτελούν τη διεπαφή των μηχανισμών που εξασφαλίζουν τη συνεχή ανταλλαγή πληροφοριών μεταξύ του συστήματος και του περιβάλλοντός του, και της λήψης αποφάσεων είτε σε επίπεδο δικτύου (πχ κατανομή της κίνησης με βάση τον όγκο των φορτίων ή τη καθυστέρηση) ή σε επίπεδο εφαρμογών (πχ καλύτερο δυνατό QoS στους χρήστες).

Με λίγα λόγια, το μοντέλο αυτό αποτελεί προσπάθεια μεταφοράς του προβλήματος ενός κοινά αποδεκτού προτύπου διαχείρισης και διαλειτουργικότητας ετερογενών δικτύων σε πρόβλημα υλοποίησης ενός κοινά αποδεκτού πλαισίου εφαρμογής (application framework) που ενσωματώνει τις βασικές αρχές του προτύπου αλλά επιτρέπει επιπλέον την εξέλιξη και επέκτασή του. Ο πηγαίος κώδικας που προκύπτει ως έξοδο της γραφικής αναπαράστασης στο Protégé 2000 είναι άμεσα διαθέσιμος προς προσομοίωση διαφορετικών σεναρίων διαχείρισης υπηρεσιών και τερματικών σε ένα B3G δίκτυο.

## 6.2 Γενικές κατευθύνσεις

Με γνώμονα τα παραπάνω, η ανάπτυξη οντολογιών για δίκτυα πέραν της τρίτης γενιάς αποτελεί επιτακτική ανάγκη για την εξεύρεση λύσεων σε σύνθετα θέματα που άπτονται του ενδιαφέροντος των διαχειριστών τέτοιων δικτύων και ερευνητών. Έτσι, έμφαση θα πρέπει να δοθεί στους εξής στόχους:

- Την ανάγκη αυτόνομης λειτουργίας των συστημάτων επικοινωνιών, στο βαθμό του εφικτού, με στόχο την αποτελεσματικότερη, σταθερή και κλιμακούμενη διαχείριση. Αυτό θα επιτευχθεί με την ανάπτυξη των κατάλληλων οντολογιών.
- Ανάπτυξη οντολογιών που θα ενσωματώνουν ολοκληρωμένες τεχνικές μάθησης, οι οποίες θα κατορθώνουν να καθοδηγούν με τον καλύτερο τρόπο τη συμπεριφορά ενός δικτύου.
- Την αποτελεσματικότερη μελέτη νέων συστημάτων, όπως το WiMAX και την ενσωμάτωσή του σε περιβάλλοντα B3G. Αυτό μπορεί και πάλι να πραγματοποιηθεί με την ανάπτυξη των κατάλληλων οντολογιών, για την εξεύρεση ενός κοινού τρόπου αναφοράς.

Αδιαμφισβήτητα, η ευρύτητα που χαρακτηρίζει τον τομέα των ασυρμάτων επικοινωνιών κάνει δυνατή τη συνεχή διεύρυνση της ανωτέρω λίστας και καθιστά πραγματοποιήσιμη την υπόσχεση συνέχισης των προσπαθειών μας για ανταπόκρισή μας στις επιταγές της – συνεχώς εξελισσόμενης - τεχνολογίας.

## 7. Βιβλιογραφία

### 7.1 Βιβλία

- Δίκτυα Υπολογιστών 4<sup>η</sup> έκδοση, A.S. Tanenbaum, 2003
- Συστήματα Κινητών Επικοινωνιών, Α. Κανάτας, Φ. Κωνσταντίνου, Γ. Πάντος, 2008
- WiMax Explained, Lawrence Harte, Dr. Kalai Kalaichevian, 2007
- P.Demestichas, G.Dimitrakopoulos, K.Tsagkaris, V.Stavroulaki, A.Katidiotis, "Introducing cognitive systems to the B3G wireless world", Cognitive Wireless Networks: Concepts, Methodologies and Visions Inspiring the Age of Enlightenment of Wireless Communications, Springer, pp.253-269, Dordrecht, The Netherlands, 2007.

### 7.2 Άρθρα

- [1] Institute of Electrical and Electronics Engineers (IEEE), 802 standards, [www.ieee802.org](http://www.ieee802.org), 2007
- [2] A. Jamalipour, T. Wada, T. Yamazato "A tutorial on multiple access technologies for beyond 3G mobile networks", *IEEE Commun. Mag.*, vol. 43, no 2, Feb. 2005, pp. 110-117.
- [3] P.Demestichas, G.Vivier, K.El-Khazen, M.Theologou, "Evolution in wireless systems management concepts: from composite radio to reconfigurability", *IEEE Commun. Mag.*, Vol. 42, No. 5, pp. 90-98, May 2004.
- [4] J. Mitola III, G. Maguire Jr. "Cognitive radio: making software radios more personal", *IEEE Personal Commun.*, Vol. 6, No 4 , pp13 -18, August 1999.
- [5] R.W.Thomas, L.A. DaSilva, A.B. MacKenzie, "Cognitive Networks", IEEE DySPAN 2005, First Symposium on Dynamic Access Networks, Baltimore USA, 8-11 November 2005.

- [6] M. Thathachar and P. Sastry, *Networks of Learning Automata*, Kluwer Academic Publishers, 2004.
- [7] F. Howarth, "Multi-agent technology: removing the 'artificial' from AI", IT-Director, <http://www.it-director.com/article.php?articleid=11774>, March 2004.
- [8] Ontology Bean Generator for Jade 2.5, <http://hcs.science.uva.nl/usr/aart/beangenerator/index25.html>
- [9] OWL-QL Project for the Stanford Knowledge Systems Laboratory, <http://ksl.stanford.edu/projects/owl-ql>
- [10] A. Seaborne, "Jena Tutorial, A Programmer's Introduction to RDQL", <http://jena.sourceforge.net/tutorial/RDQL/index.html>, February 2004.
- [11] R. Seeley "The Semantic Web: The OWL has landed", ADT Magazine, <http://www.adtmag.com/article.asp?id=8144>, August 2003.
- [12] Software Agents: An Overview, <http://more.btexact.com/projects/agents/publish/papers/review2.htm>
- [13] T. Sundsted, "An introduction to agents", JavaWorld, <http://www.javaworld.com/javaworld/jw-06-1998>, June 1998.
- [14] G. Antoniou and F. van Harmelen, "Web Ontology Language: OWL", <http://www.cs.vu.nl/~frankh/postscript/OntoHandbook03OWL.pdf>
- [15] The World Wide Web Consortium, OWL, <http://www.w3.org/TR/2004/>
- [16] IEEE Std 1900.4-2009: "IEEE Standard for Architectural Building Blocks Enabling Network-Device Distributed Decision Making for Optimized Radio Resource Usage in Heterogeneous Wireless Access Networks"
- [17] S.Buljore et al., "IEEE P1900.4 System Overview on Architecture and Enablers for Optimised Radio and Spectrum resource usage" in Proc. IEEE Dynamic Spectrum Access Networks conference (DySPAN) 2008, Chicago, 14th– 17th October, 2008
- [18] G.Dimitrakopoulos, K.Moessner, C.Kloeck, D.Grandblaise, Sophie Gault, O.Sallent, K.Tsagkaris, P. Demestichas, "Adaptive Resource Management Platform for

Reconfigurable Networks”, ACM/Springer Mobile Networks and Applications journal,  
Vol. 11, N° 6, December 2006, pp. 799-811.

- [19] Ofcom – “Spectrum Framework Review”  
<http://www.ofcom.org.uk/consult/condocs/sfr/> - link valid as of December 2005.
- [20] K.Moessner, et al, “Dynamic radio resource allocation strategies and timescales”, SDR  
Forum technical Conference 2005, Anaheim, CA, USA, November 2005.
- [21] J. Strassner, “Autonomic networking – theory and practice”, In Proc. 9th IFIP/IEEE  
International Symposium on Network Management (IM’2005), Nice, France, May 2005.
- [22] J. Kephart, D. Chess, “The vision of autonomic computing”, IEEE Computer, Vol. 36,  
No.1, pp. 41-50, January 2003.
- [23] M.Mouly and M.B Pautet. The GSM System for Mobile Communications. M.Mouly, 49  
rue Louise Bruneau, Palaiseau, France 1992
- [24] The 3Rd Generation Partnership Project (3GPP) Web site, [www.3gpp.org](http://www.3gpp.org) , 2007
- [25] H.Holma, A.Toscali, W-CDMA for UMTS, J.Wiley & Sons, second edition 2002
- [26] A.J.Viterbi, CDMA, Principles of Spread Spectrum Communications, Addison-  
Wesley1995.
- [27] D.Knisely, S.Kumer, S.Laha and S.Nanada, “Evolution of wireless data services: IS-95 to  
CDMA2000”, IEEE Commun.Mag., vol.36, pp.140-149, Oct 1998
- [28] IEEE Personal Communications, Special Issue on The Evolution of TDMA to 3G, vol.6,  
no.3, June 1999.
- [29] C.Bettstetter, H.-J.Vogel and J.Eberspracher, “GSM Phase 2+, General Packet Radio  
Service GPRS: Architecture, Protocols and Interfaces”, IEEE Communications Surveys,  
vol.2, no.3, Third Quarter 1999.
- [30] R. Kalden, I. Meirick, M. Meyer, “Wireless Internet access based on GPRS”, IEEE  
Personal Commun., Vol. 7 No. 2, April 2000
- [31] Institute of Electrical and Electronics Engineers (IEEE), 802 standards,  
[www.ieee802.org](http://www.ieee802.org), 2007

- [32] WiMAX Forum, <http://www.wimaxforum.org>, 2007
- [33] Digital Video Broadcasting (DVB), Web site, [www.dvb.org](http://www.dvb.org), 2007
- [34] Bluetooth, [www.bluetooth.com](http://www.bluetooth.com), 2007
- [35] ZigBee Alliance, [www.zigbee.org](http://www.zigbee.org), 2007
- [36] A. Jamalipour, T. Wada, T. Yamazato "A tutorial on multiple access technologies for beyond 3G mobile networks", IEEE Commun. Mag., vol. 43, no 2, Feb. 2005, pp. 110-117.
- [37] Wireless World Research Forum (WWRF), [www.wireless-world-research.org](http://www.wireless-world-research.org), 2007.
- [38] N. Niebert, A. Schieder, J. Zander , R. Hancock (Eds.), "Ambient Networks: Co-operative Mobile Networking for the Wireless World", Wiley, 2007.
- [39] Aarts, E., Harwig, R., Schuurmans, M., "Ambient Intelligence", in Denning, P.J. (Ed) The Invisible Future, ACM Press. pp. 235-250, 2001.
- [40] E. Aarts and S. Marzano, eds. "The New Everyday: Visions of Ambient Intelligence", 010 Publishing, Rotterdam, The Netherlands, 2003.
- [41] E. Aarts, "Ambient Intelligence: A Multimedia Perspective," IEEE MultiMedia, vol. 11, no. 1, pp. 12-19, January/March, 2004
- [42] G. Riva, F. Vatalaro, F. Davide, M. Alcañiz (Eds.), "Ambient Intelligence" , IOS Press, 2005
- [43] M. Weiser, "The Computer for the Twenty-First Century," Scientific Amer., vol. 265, no. 3, 1991, pp. 94-104. (the father of ubiquitous computing)
- [44] M. Brunner, A. Gunnar, H. Abrahamsoson et al. "Ambient Networks Management Challenges and Approaches", Lecture Notes in Computer Science, Book chapter in Mobility Aware Technologies and Applications, Vol. 3284, pp 196-216, 2004
- [45] Gellersen, H.-W., Schmidt, A., Beigl, M. "Adding Some Smartness to Devices and Everyday Things", Proceedings of IEEE Workshop on Mobile Computing Systems and Applications 2000 (WMCSA '00), Dec. 2000, Monterey, USA, IEEE Press

- [46] ITEA Ambience Project (000003), available online at <http://www.hitech-projects.com/euprojects/ambience/>
- [47] M. Satyanarayanan, "Pervasive Computing: Vision and Challenges," IEEE Personal Communication, vol. 8, No.4, Aug. 2001, pp. 10-17.
- [48] C. Basile, M.-O. Killijian and D. Powell. A survey of dependability issues in mobile wireless networks. Technical report, Laboratory for Analysis and Architecture of Systems, National Center for Scientific Research, Toulouse, France, Feb 2003
- [49] Hiroshi Harada, Joint Session of 1900.4, 1900.5, 1900.6 WGs and WS Radio Overview of 1900.4a, 2010
- [50] Hiroshi Harada (SCC41 Acting Chair), Paul Houzé (P1900.4 chair), Lynn Grande (P1900.5 chair), Klaus Moessner (P1900.6 chair) IEEE SCC41 Standards for Dynamic Spectrum Access Networks, 2010
- [51] Peter Bartalos and Maria Bielikova, An approach to object-ontology mapping
- [52] Dencho N. Batanov, Merging Ontologies and Object-Oriented technologies for software development, 2006
- [53] Tao YANG, Using Classified Interrelated Object Model to Represent Ontology, 2008
- [54] Aristi Galani, Kostas Tsagkaris, Panagiotis Demestichas, A Functional Architecture for Optimized Radio Resource and Spectrum Usage in Heterogeneous Wireless Networks, 2008
- [55] Soodesh Buljore, Hiroshi Harada, Paul Houze, Kostas Tsagkaris, Vladimir Ivanov, Klaus Nolte, Tim Farnham, Oliver Holland, Makis Stamatelatos, IEEE P1900.4 Standard: Reconfiguration of Multi-Radio Systems, 2008
- [56] Soodesh Buljore, Vincent Merat. Hiroshi Harada, Stanislav Filin, Paul Houze, Kostas Tsagkaris, Vladimir Ivanov, Klaus Nolte, Tim Farnham, Oliver Holland, IEEE P1900.4 System Overview on Architecture and Enablers for Optimised Radio and Spectrum resource usage, 2008

### 7.3 Εργασίες / Διατριβές

- [1] Βελτιστοποίηση γνωσιακών σύνθετων ασύρματων δικτύων υψηλών ταχυτήτων, Α. Σατσάκης, 2010
- [2] Διαχείριση τερματικών σε περιβάλλοντα δικτύων υψηλών ταχυτήτων B3G, Α. Σαρσεμπάγιεβα, 2018

### 7.4 URLs

- <http://www.intel.com/technology/wimax/>
- [www.wimaxforum.org/](http://www.wimaxforum.org/)
- “Ontology”, Wikipedia, <http://en.wikipedia.org>
- The Foundation for Intelligent Physical Agents, <http://www.fipa.org/>
- The World Wide Web Consortium, OWL, <http://www.w3.org/TR/2004/>
- UMBC AgentWeb, <http://www.cs.umbc.edu/kqml/>
- W3C Semantic Web, <http://www.w3.org/2001/sw/>
- W3Schools, [http://www.w3schools.com/rdf/rdf\\_owl.asp](http://www.w3schools.com/rdf/rdf_owl.asp)
- Web Ontology Language (OWL), <http://www.w3.org/2004/OWL/>
- A. Pease, “Why use OWL?”, <http://www.xfront.com/why-use-owl.html>
- Protégé OWL Plugin, <http://protege.stanford.edu/plugins/owl/>



- Protégé, <http://protege.stanford.edu/>
- Resource Description Framework (RDF), <http://www.w3.org/RDF/>
- WiMAX Forum, <http://www.wimaxforum.org>, 2007
- Digital Video Broadcasting (DVB), Web site, [www.dvb.org](http://www.dvb.org), 2007
- Bluetooth, [www.bluetooth.com](http://www.bluetooth.com), 2007
- ZigBee Alliance, [www.zigbee.org](http://www.zigbee.org), 2007
- Wireless World Research Forum (WWRF), [www.wireless-world-research.org](http://www.wireless-world-research.org), 2007.
- FP6/IST project E2R (End-to-End Reconfigurability), [www.e2r.motlabs.com](http://www.e2r.motlabs.com), 2007.
- Institute of Electrical and Electronic Engineers (IEEE) web site, <http://www.ieee.org> ,  
May 2009.
- Third (3rd) Generation Partnership Project (3GPP), Web site, [www.3gpp.org](http://www.3gpp.org) , 2007
- Digital Video Broadcasting (DVB), Web site, [www.dvb.org](http://www.dvb.org), 2007

## 8. Παράρτημα

### 8.1 Πηγαίος Κώδικας

```
package gr.papei.ontomap.entities;
```

```
import gr.papei.ontomap.entities.impl.*;
```

```
import edu.stanford.smi.protege.model.FrameID;
```

```
import edu.stanford.smi.protege.owl.model.*;
```

```
import edu.stanford.smi.protege.owl.model.impl.OWLUtil;
```

```
import edu.stanford.smi.protege.owl.javacode.ProtegeJavaMapping;
```

```
import java.util.*;
```

```
public class OntoMapper {
```

```
    private OWLModel owlModel;
```

```
    static {
```

```
        ProtegeJavaMapping.add("http://www.semanticweb.org/ontologies/2009/8/p1900.owl#CWN",  
                                CWN.class, ImplCWN.class);
```

```
ProtegeJavaMapping.add("http://www.semanticweb.org/ontologies/2009/8/p1900.owl#NRM",  
NRM.class, ImplNRM.class);  
  
ProtegeJavaMapping.add("http://www.semanticweb.org/ontologies/2009/8/p1900.owl#OSM",  
OSM.class, ImplOSM.class);  
  
ProtegeJavaMapping.add("http://www.semanticweb.org/ontologies/2009/8/p1900.owl#QoS",  
QoS.class, ImplQoS.class);  
  
ProtegeJavaMapping.add("http://www.semanticweb.org/ontologies/2009/8/p1900.owl#RAN",  
RAN.class, ImplRAN.class);  
  
ProtegeJavaMapping.add("http://www.semanticweb.org/ontologies/2009/8/p1900.owl#RMC",  
RMC.class, ImplRMC.class);  
  
ProtegeJavaMapping.add("http://www.semanticweb.org/ontologies/2009/8/p1900.owl#RRC",  
RRC.class, ImplRRC.class);  
  
ProtegeJavaMapping.add("http://www.semanticweb.org/ontologies/2009/8/p1900.owl#SAP",  
SAP.class, ImplSAP.class);  
  
ProtegeJavaMapping.add("http://www.semanticweb.org/ontologies/2009/8/p1900.owl#TMC",  
TMC.class, ImplTMC.class);  
  
ProtegeJavaMapping.add("http://www.semanticweb.org/ontologies/2009/8/p1900.owl#TRC",  
TRC.class, ImplTRC.class);  
  
ProtegeJavaMapping.add("http://www.semanticweb.org/ontologies/2009/8/p1900.owl#TRM",  
TRM.class, ImplTRM.class);  
  
ProtegeJavaMapping.add("http://www.semanticweb.org/ontologies/2009/8/p1900.owl#Terminal",  
Terminal.class, ImplTerminal.class);  
  
ProtegeJavaMapping.add("http://www.semanticweb.org/ontologies/2009/8/p1900.owl#context",  
Context.class, ImplContext.class);  
  
ProtegeJavaMapping.add("http://www.semanticweb.org/ontologies/2009/8/p1900.owl#if_NRM_OS  
M", If_NRM_OSM.class, ImplIf_NRM_OSM.class);  
  
ProtegeJavaMapping.add("http://www.semanticweb.org/ontologies/2009/8/p1900.owl#if_NRM_R  
MC", If_NRM_RMC.class, ImplIf_NRM_RMC.class);
```

```
ProtegeJavaMapping.add("http://www.semanticweb.org/ontologies/2009/8/p1900.owl#if_NRM_RRC", If_NRM_RRC.class, ImplIf_NRM_RRC.class);
```

```
ProtegeJavaMapping.add("http://www.semanticweb.org/ontologies/2009/8/p1900.owl#if_NRM_TRM", If_NRM_TRM.class, ImplIf_NRM_TRM.class);
```

```
ProtegeJavaMapping.add("http://www.semanticweb.org/ontologies/2009/8/p1900.owl#if_TRM_TMC", If_TRM_TMC.class, ImplIf_TRM_TMC.class);
```

```
ProtegeJavaMapping.add("http://www.semanticweb.org/ontologies/2009/8/p1900.owl#if_TRM_TRC", If_TRM_TRC.class, ImplIf_TRM_TRC.class);
```

```
ProtegeJavaMapping.add("http://www.semanticweb.org/ontologies/2009/8/p1900.owl#interface", Interf.class, ImplInterface.class);
```

```
ProtegeJavaMapping.add("http://www.semanticweb.org/ontologies/2009/8/p1900.owl#location", Location.class, ImplLocation.class);
```

```
ProtegeJavaMapping.add("http://www.semanticweb.org/ontologies/2009/8/p1900.owl#measurement", Measurement.class, ImplMeasurement.class);
```

```
ProtegeJavaMapping.add("http://www.semanticweb.org/ontologies/2009/8/p1900.owl#rCFG_MEDIA_SAP", rCFG_MEDIA_SAP.class, ImplrCFG_MEDIA_SAP.class);
```

```
ProtegeJavaMapping.add("http://www.semanticweb.org/ontologies/2009/8/p1900.owl#rCFG_MNG_SAP", rCFG_MNG_SAP.class, ImplrCFG_MNG_SAP.class);
```

```
ProtegeJavaMapping.add("http://www.semanticweb.org/ontologies/2009/8/p1900.owl#rCFG_TR_SAP", rCFG_TR_SAP.class, ImplrCFG_TR_SAP.class);
```

```
ProtegeJavaMapping.add("http://www.semanticweb.org/ontologies/2009/8/p1900.owl#terminalCapability", TerminalCapability.class, ImplTerminalCapability.class);
```

```
ProtegeJavaMapping.add("http://www.semanticweb.org/ontologies/2009/8/p1900.owl#userPreference", UserPreference.class, ImplUserPreference.class);
```

```
}
```

```
public OntoMapper(OWLModel owlModel) {
```

```
    this.owlModel = owlModel;
```

```
}
```

```
public <X> X create(Class<? extends X> javaInterface, String name) {
```

```
    return ProtegeJavaMapping.create(owlModel, javaInterface, name);
```

```
}
```

```
public RDFSNamedClass getCWNClass() {
```

```
    final String uri = "http://www.semanticweb.org/ontologies/2009/8/p1900.owl#CWN";
```

```
    final String name = owlModel.getResourceNameForURI(uri);
```

```
    return owlModel.getRDFSNamedClass(name);
```

```
}
```

```
public CWN createCWN(String name) {
```

```
    final RDFSNamedClass cls = getCWNClass();
```

```
    if (name == null) {
```

```
        name = owlModel.getNextAnonymousResourceName();
```

```
    }
```

```
    return new ImplCWN(owlModel, cls.createInstance(name).getFrameID());
```

```
}
```

```
public CWN getCWN(String name) {
```

```
RDFResource res = owlModel.getRDFResource(OWLUtil.getInternalFullName(owlModel, name));
if (res == null) {return null;}
if (res instanceof CWN) {
    return (CWN) res;
} else if (res.hasProtegeType(getCWNClass())) {
    return new ImplCWN(owlModel, res.getFrameID());
}
return null;
}

public Collection<CWN> getAllCWNInstances() {
    return getAllCWNInstances(false);
}

public Collection<CWN> getAllCWNInstances(boolean transitive) {
    Collection<CWN> result = new ArrayList<CWN>();
    final RDFSNamedClass cls = getCWNClass();
    RDFResource owlIndividual;
    for (Iterator it = cls.getInstance(transitive).iterator(); it.hasNext();) {
        owlIndividual = (RDFResource) it.next();
        result.add(new ImplCWN(owlModel, owlIndividual.getFrameID()));
    }
    return result;
}
```

```
public RDFSNamedClass getNRMClass() {  
    final String uri = "http://www.semanticweb.org/ontologies/2009/8/p1900.owl#NRM";  
    final String name = owlModel.getResourceNameForURI(uri);  
    return owlModel.getRDFSNamedClass(name);  
}  
  
public NRM createNRM(String name) {  
    final RDFSNamedClass cls = getNRMClass();  
    if (name == null) {  
        name = owlModel.getNextAnonymousResourceName();  
    }  
    return new ImplNRM(owlModel, cls.createInstance(name).getFrameID());  
}  
  
public NRM getNRM(String name) {  
    RDFResource res = owlModel.getRDFResource(OWLUtil.getInternalFullName(owlModel, name));  
    if (res == null) {return null;}  
    if (res instanceof NRM) {  
        return (NRM) res;  
    } else if (res.hasProtegeType(getNRMClass())) {  
        return new ImplNRM(owlModel, res.getFrameID());  
    }  
    return null;  
}  
  
public Collection<NRM> getAllNRMInstances() {
```

```
return getAllNRMInstances(false);
}

public Collection<NRM> getAllNRMInstances(boolean transitive) {
    Collection<NRM> result = new ArrayList<NRM>();
    final RDFSNamedClass cls = getNRMClass();
    RDFResource owlIndividual;
    for (Iterator it = cls.getInstances(transitive).iterator(); it.hasNext();) {
        owlIndividual = (RDFResource) it.next();
        result.add(new ImplNRM(owlModel, owlIndividual.getFrameID()));
    }
    return result;
}

public RDFSNamedClass getOSMClass() {
    final String uri = "http://www.semanticweb.org/ontologies/2009/8/p1900.owl#OSM";
    final String name = owlModel.getResourceNameForURI(uri);
    return owlModel.getRDFSNamedClass(name);
}

public OSM createOSM(String name) {
    final RDFSNamedClass cls = getOSMClass();
    if (name == null) {
        name = owlModel.getNextAnonymousResourceName();
    }
}
```



```
return new ImplOSM(owlModel, cls.createInstance(name).getFrameID());
}

public OSM getOSM(String name) {
    RDFResource res = owlModel.getRDFResource(OWLUtil.getInternalFullName(owlModel, name));
    if (res == null) {return null;}
    if (res instanceof OSM) {
        return (OSM) res;
    } else if (res.hasProtegeType(getOSMClass())) {
        return new ImplOSM(owlModel, res.getFrameID());
    }
    return null;
}

public Collection<OSM> getAllOSMInstances() {
    return getAllOSMInstances(false);
}

public Collection<OSM> getAllOSMInstances(boolean transitive) {
    Collection<OSM> result = new ArrayList<OSM>();
    final RDFSNamedClass cls = getOSMClass();
    RDFResource owlIndividual;
    for (Iterator it = cls.getInstance(transitive).iterator(); it.hasNext(); ) {
        owlIndividual = (RDFResource) it.next();
        result.add(new ImplOSM(owlModel, owlIndividual.getFrameID()));
    }
}
```

```
return result;
}

public RDFSNamedClass getQoSClass() {
    final String uri = "http://www.semanticweb.org/ontologies/2009/8/p1900.owl#QoS";
    final String name = owlModel.getResourceNameForURI(uri);
    return owlModel.getRDFSNamedClass(name);
}

public QoS createQoS(String name) {
    final RDFSNamedClass cls = getQoSClass();
    if (name == null) {
        name = owlModel.getNextAnonymousResourceName();
    }
    return new ImplQoS(owlModel, cls.createInstance(name).getFrameID());
}

public QoS getQoS(String name) {
    RDFResource res = owlModel.getRDFResource(OWLUtil.getInternalFullName(owlModel, name));
    if (res == null) {return null;}
    if (res instanceof QoS) {
        return (QoS) res;
    } else if (res.hasProtegeType(getQoSClass())) {
        return new ImplQoS(owlModel, res.getFrameID());
    }
}
```

```
return null;
}

public Collection<QoS> getAllQoSInstances() {
    return getAllQoSInstances(false);
}

public Collection<QoS> getAllQoSInstances(boolean transitive) {
    Collection<QoS> result = new ArrayList<QoS>();
    final RDFSNamedClass cls = getQoSClass();
    RDFResource owlIndividual;
    for (Iterator it = cls.getInstance(transitive).iterator(); it.hasNext();) {
        owlIndividual = (RDFResource) it.next();
        result.add(new ImplQoS(owlModel, owlIndividual.getFrameID()));
    }
    return result;
}

public RDFSNamedClass getRANClass() {
    final String uri = "http://www.semanticweb.org/ontologies/2009/8/p1900.owl#RAN";
    final String name = owlModel.getResourceNameForURI(uri);
    return owlModel.getRDFSNamedClass(name);
}

public RAN createRAN(String name) {
```

```
final RDFSNamedClass cls = getRANClass();

if (name == null) {
    name = owlModel.getNextAnonymousResourceName();
}

return new ImplRAN(owlModel, cls.createInstance(name).getFrameID());
}

public RAN getRAN(String name) {
    RDFResource res = owlModel.getRDFResource(OWLUtil.getInternalFullName(owlModel, name));
    if (res == null) {return null;}
    if (res instanceof RAN) {
        return (RAN) res;
    } else if (res.hasProtegeType(getRANClass())) {
        return new ImplRAN(owlModel, res.getFrameID());
    }
    return null;
}

public Collection<RAN> getAllRANInstances() {
    return getAllRANInstances(false);
}

public Collection<RAN> getAllRANInstances(boolean transitive) {
    Collection<RAN> result = new ArrayList<RAN>();

    final RDFSNamedClass cls = getRANClass();

    RDFResource owlIndividual;
```

```
for (Iterator it = cls.getInstances(transitive).iterator();it.hasNext();) {
    owlIndividual = (RDFResource) it.next();
    result.add(new ImplRAN(owlModel, owlIndividual.getFrameID()));
}
return result;
}

public RDFSNamedClass getRMCClass() {
    final String uri = "http://www.semanticweb.org/ontologies/2009/8/p1900.owl#RMC";
    final String name = owlModel.getResourceNameForURI(uri);
    return owlModel.getRDFSNamedClass(name);
}

public RMC createRMC(String name) {
    final RDFSNamedClass cls = getRMCClass();
    if (name == null) {
        name = owlModel.getNextAnonymousResourceName();
    }
    return new ImplRMC(owlModel, cls.createInstance(name).getFrameID());
}

public RMC getRMC(String name) {
    RDFResource res = owlModel.getRDFResource(OWLUtil.getInternalFullName(owlModel, name));
    if (res == null) {return null;}
    if (res instanceof RMC) {
```

```
    return (RMC) res;

} else if (res.hasProtegeType(getRMCClass())) {

    return new ImplRMC(owlModel, res.getFrameID());

}

return null;

}

public Collection<RMC> getAllRMCInstances() {

    return getAllRMCInstances(false);

}

public Collection<RMC> getAllRMCInstances(boolean transitive) {

    Collection<RMC> result = new ArrayList<RMC>();

    final RDFSNamedClass cls = getRMCClass();

    RDFResource owlIndividual;

    for (Iterator it = cls.getInstances(transitive).iterator(); it.hasNext();) {

        owlIndividual = (RDFResource) it.next();

        result.add(new ImplRMC(owlModel, owlIndividual.getFrameID()));

    }

    return result;

}

public RDFSNamedClass getRRCClass() {

    final String uri = "http://www.semanticweb.org/ontologies/2009/8/p1900.owl#RRC";

    final String name = owlModel.getResourceNameForURI(uri);
```

```
return owlModel.getRDFSNamedClass(name);
}

public RRC createRRC(String name) {
    final RDFSNamedClass cls = getRRCClass();
    if (name == null) {
        name = owlModel.getNextAnonymousResourceName();
    }
    return new ImplRRC(owlModel, cls.createInstance(name).getFrameID());
}

public RRC getRRC(String name) {
    RDFResource res = owlModel.getRDFResource(OWLUtil.getInternalFullName(owlModel, name));
    if (res == null) {return null;}
    if (res instanceof RRC) {
        return (RRC) res;
    } else if (res.hasProtegeType(getRRCClass())) {
        return new ImplRRC(owlModel, res.getFrameID());
    }
    return null;
}

public Collection<RRC> getAllRRCInstances() {
    return getAllRRCInstances(false);
}
```

```
public Collection<RRC> getAllRRCInstances(boolean transitive) {  
    Collection<RRC> result = new ArrayList<RRC>();  
  
    final RDFSNamedClass cls = getRRCClass();  
  
    RDFResource owlIndividual;  
  
    for (Iterator it = cls.getInstance(transitive).iterator(); it.hasNext();) {  
        owlIndividual = (RDFResource) it.next();  
  
        result.add(new ImplRRC(owlModel, owlIndividual.getFrameID()));  
    }  
  
    return result;  
}
```

```
public RDFSNamedClass getSAPClass() {  
    final String uri = "http://www.semanticweb.org/ontologies/2009/8/p1900.owl#SAP";  
  
    final String name = owlModel.getResourceNameForURI(uri);  
  
    return owlModel.getRDFSNamedClass(name);  
}
```

```
public SAP createSAP(String name) {  
    final RDFSNamedClass cls = getSAPClass();  
  
    if (name == null) {  
        name = owlModel.getNextAnonymousResourceName();  
    }  
  
    return new ImplSAP(owlModel, cls.createInstance(name).getFrameID());  
}
```



```
public SAP getSAP(String name) {  
    RDFResource res = owlModel.getRDFResource(OWLUtil.getInternalFullName(owlModel, name));  
    if (res == null) {return null;}  
    if (res instanceof SAP) {  
        return (SAP) res;  
    } else if (res.hasProtegeType(getSAPClass())) {  
        return new ImplSAP(owlModel, res.getFrameID());  
    }  
    return null;  
}  
  
public Collection<SAP> getAllSAPInstances() {  
    return getAllSAPInstances(false);  
}  
  
public Collection<SAP> getAllSAPInstances(boolean transitive) {  
    Collection<SAP> result = new ArrayList<SAP>();  
    final RDFSNamedClass cls = getSAPClass();  
    RDFResource owlIndividual;  
    for (Iterator it = cls.getInstances(transitive).iterator(); it.hasNext();) {  
        owlIndividual = (RDFResource) it.next();  
        result.add(new ImplSAP(owlModel, owlIndividual.getFrameID()));  
    }  
    return result;  
}
```

```
public RDFSNamedClass getTMCClass() {  
    final String uri = "http://www.semanticweb.org/ontologies/2009/8/p1900.owl#TMC";  
    final String name = owlModel.getResourceNameForURI(uri);  
    return owlModel.getRDFSNamedClass(name);  
}  
  
public TMC createTMC(String name) {  
    final RDFSNamedClass cls = getTMCClass();  
    if (name == null) {  
        name = owlModel.getNextAnonymousResourceName();  
    }  
    return new ImplTMC(owlModel, cls.createInstance(name).getFrameID());  
}  
  
public TMC getTMC(String name) {  
    RDFResource res = owlModel.getRDFResource(OWLUtil.getInternalFullName(owlModel, name));  
    if (res == null) {return null;}  
    if (res instanceof TMC) {  
        return (TMC) res;  
    } else if (res.hasProtegeType(getTMCClass())) {  
        return new ImplTMC(owlModel, res.getFrameID());  
    }  
    return null;  
}
```

```
public Collection<TMC> getAllTMCInstances() {  
    return getAllTMCInstances(false);  
}
```

```
public Collection<TMC> getAllTMCInstances(boolean transitive) {  
    Collection<TMC> result = new ArrayList<TMC>();  
    final RDFSNamedClass cls = getTMCClass();  
    RDFResource owlIndividual;  
    for (Iterator it = cls.getInstances(transitive).iterator(); it.hasNext();) {  
        owlIndividual = (RDFResource) it.next();  
        result.add(new ImplTMC(owlModel, owlIndividual.getFrameID()));  
    }  
    return result;  
}
```

```
public RDFSNamedClass getTRCClass() {  
    final String uri = "http://www.semanticweb.org/ontologies/2009/8/p1900.owl#TRC";  
    final String name = owlModel.getResourceNameForURI(uri);  
    return owlModel.getRDFSNamedClass(name);  
}
```

```
public TRC createTRC(String name) {  
    final RDFSNamedClass cls = getTRCClass();  
    if (name == null) {  
        name = owlModel.getNextAnonymousResourceName();  
    }  
}
```

```
}  
  
return new ImplTRC(owlModel, cls.createInstance(name).getFrameID());  
}  
  
public TRC getTRC(String name) {  
    RDFResource res = owlModel.getRDFResource(OWLUtil.getInternalFullName(owlModel, name));  
    if (res == null) {return null;}  
    if (res instanceof TRC) {  
        return (TRC) res;  
    } else if (res.hasProtegeType(getTRCClass())) {  
        return new ImplTRC(owlModel, res.getFrameID());  
    }  
    return null;  
}  
  
public Collection<TRC> getAllTRCInstances() {  
    return getAllTRCInstances(false);  
}  
  
public Collection<TRC> getAllTRCInstances(boolean transitive) {  
    Collection<TRC> result = new ArrayList<TRC>();  
    final RDFSNamedClass cls = getTRCClass();  
    RDFResource owlIndividual;  
    for (Iterator it = cls.getInstances(transitive).iterator(); it.hasNext();) {  
        owlIndividual = (RDFResource) it.next();  
        result.add(new ImplTRC(owlModel, owlIndividual.getFrameID()));  
    }  
}
```

```
}  
  
return result;  
  
}  
  
public RDFSNamedClass getTRMClass() {  
    final String uri = "http://www.semanticweb.org/ontologies/2009/8/p1900.owl#TRM";  
    final String name = owlModel.getResourceNameForURI(uri);  
    return owlModel.getRDFSNamedClass(name);  
}  
  
public TRM createTRM(String name) {  
    final RDFSNamedClass cls = getTRMClass();  
    if (name == null) {  
        name = owlModel.getNextAnonymousResourceName();  
    }  
    return new ImplTRM(owlModel, cls.createInstance(name).getFrameID());  
}  
  
public TRM getTRM(String name) {  
    RDFResource res = owlModel.getRDFResource(OWLUtil.getInternalFullName(owlModel, name));  
    if (res == null) {return null;}  
    if (res instanceof TRM) {  
        return (TRM) res;  
    } else if (res.hasProtegeType(getTRMClass())) {  
        return new ImplTRM(owlModel, res.getFrameID());  
    }  
}
```

```
}  
  
return null;  
  
}  
  
public Collection<TRM> getAllTRMInstances() {  
    return getAllTRMInstances(false);  
}  
  
public Collection<TRM> getAllTRMInstances(boolean transitive) {  
    Collection<TRM> result = new ArrayList<TRM>();  
  
    final RDFSNamedClass cls = getTRMClass();  
  
    RDFResource owlIndividual;  
  
    for (Iterator it = cls.getInstance(transitive).iterator(); it.hasNext();) {  
        owlIndividual = (RDFResource) it.next();  
  
        result.add(new ImplTRM(owlModel, owlIndividual.getFrameID()));  
    }  
  
    return result;  
}  
  
public RDFSNamedClass getTerminalClass() {  
    final String uri = "http://www.semanticweb.org/ontologies/2009/8/p1900.owl#Terminal";  
    final String name = owlModel.getResourceNameForURI(uri);  
  
    return owlModel.getRDFSNamedClass(name);  
}
```

```
public Terminal createTerminal(String name) {  
    final RDFSNamedClass cls = getTerminalClass();  
    if (name == null) {  
        name = owlModel.getNextAnonymousResourceName();  
    }  
    return new ImplTerminal(owlModel, cls.createInstance(name).getFrameID());  
}  
  
public Terminal getTerminal(String name) {  
    RDFResource res = owlModel.getRDFResource(OWLUtil.getInternalFullName(owlModel, name));  
    if (res == null) {return null;}  
    if (res instanceof Terminal) {  
        return (Terminal) res;  
    } else if (res.hasProtegeType(getTerminalClass())) {  
        return new ImplTerminal(owlModel, res.getFrameID());  
    }  
    return null;  
}  
  
public Collection<Terminal> getAllTerminalInstances() {  
    return getAllTerminalInstances(false);  
}  
  
public Collection<Terminal> getAllTerminalInstances(boolean transitive) {  
    Collection<Terminal> result = new ArrayList<Terminal>();  
    final RDFSNamedClass cls = getTerminalClass();
```

```
RDFResource owlIndividual;

for (Iterator it = cls.getInstances(transitive).iterator();it.hasNext()); {

    owlIndividual = (RDFResource) it.next();

    result.add(new ImplTerminal(owlModel, owlIndividual.getFrameID()));

}

return result;

}

public RDFSNamedClass getcontextClass() {

    final String uri = "http://www.semanticweb.org/ontologies/2009/8/p1900.owl#context";

    final String name = owlModel.getResourceNameForURI(uri);

    return owlModel.getRDFSNamedClass(name);

}

public Context createcontext(String name) {

    final RDFSNamedClass cls = getcontextClass();

    if (name == null) {

        name = owlModel.getNextAnonymousResourceName();

    }

    return new ImplContext(owlModel, cls.createInstance(name).getFrameID());

}

public Context getcontext(String name) {

    RDFResource res = owlModel.getRDFResource(OWLUtil.getInternalFullName(owlModel, name));

    if (res == null) {return null;}

}
```



```
if (res instanceof Context) {
    return (Context) res;
} else if (res.hasProtegeType(getcontextClass())) {
    return new ImplContext(owlModel, res.getFrameID());
}
return null;
}

public Collection<Context> getAllcontextInstances() {
    return getAllcontextInstances(false);
}

public Collection<Context> getAllcontextInstances(boolean transitive) {
    Collection<Context> result = new ArrayList<Context>();
    final RDFSNamedClass cls = getcontextClass();
    RDFResource owlIndividual;
    for (Iterator it = cls.getInstances(transitive).iterator(); it.hasNext();) {
        owlIndividual = (RDFResource) it.next();
        result.add(new ImplContext(owlModel, owlIndividual.getFrameID()));
    }
    return result;
}

public RDFSNamedClass getif_NRM_OSMClass() {
    final String uri = "http://www.semanticweb.org/ontologies/2009/8/p1900.owl#if_NRM_OSM";
```

```
final String name = owlModel.getResourceNameForURI(uri);

return owlModel.getRDFSNamedClass(name);

}

public If_NRM_OSM createIf_NRM_OSM(String name) {

    final RDFSNamedClass cls = getIf_NRM_OSMClass();

    if (name == null) {

        name = owlModel.getNextAnonymousResourceName();

    }

    return new ImplIf_NRM_OSM(owlModel, cls.createInstance(name).getFrameID());

}

public If_NRM_OSM getIf_NRM_OSM(String name) {

    RDFResource res = owlModel.getRDFResource(OWLUtil.getInternalFullName(owlModel, name));

    if (res == null) {return null;}

    if (res instanceof If_NRM_OSM) {

        return (If_NRM_OSM) res;

    } else if (res.hasProtegeType(getIf_NRM_OSMClass())) {

        return new ImplIf_NRM_OSM(owlModel, res.getFrameID());

    }

    return null;

}

public Collection<If_NRM_OSM> getAllIf_NRM_OSMInstances() {

    return getAllIf_NRM_OSMInstances(false);

}

}
```

```
public Collection<If_NRM_OSM> getAllIf_NRM_OSMInstances(boolean transitive) {  
    Collection<If_NRM_OSM> result = new ArrayList<If_NRM_OSM>();  
    final RDFSNamedClass cls = getIf_NRM_OSMClass();  
    RDFResource owlIndividual;  
    for (Iterator it = cls.getInstances(transitive).iterator(); it.hasNext();) {  
        owlIndividual = (RDFResource) it.next();  
        result.add(new ImplIf_NRM_OSM(owlModel, owlIndividual.getFrameID()));  
    }  
    return result;  
}
```

```
public RDFSNamedClass getIf_NRM_RMClass() {  
    final String uri = "http://www.semanticweb.org/ontologies/2009/8/p1900.owl#if_NRM_RMC";  
    final String name = owlModel.getResourceNameForURI(uri);  
    return owlModel.getRDFSNamedClass(name);  
}
```

```
public If_NRM_RMC createIf_NRM_RMC(String name) {  
    final RDFSNamedClass cls = getIf_NRM_RMClass();  
    if (name == null) {  
        name = owlModel.getNextAnonymousResourceName();  
    }  
    return new ImplIf_NRM_RMC(owlModel, cls.createInstance(name).getFrameID());  
}
```

```
public If_NRM_RMC getif_NRM_RMC(String name) {  
    RDFResource res = owlModel.getRDFResource(OWLUtil.getInternalFullName(owlModel, name));  
    if (res == null) {return null;}  
    if (res instanceof If_NRM_RMC) {  
        return (If_NRM_RMC) res;  
    } else if (res.hasProtegeType(getif_NRM_RMCClass())) {  
        return new ImplIf_NRM_RMC(owlModel, res.getFrameID());  
    }  
    return null;  
}  
  
public Collection<If_NRM_RMC> getAllif_NRM_RMCIstances() {  
    return getAllif_NRM_RMCIstances(false);  
}  
  
public Collection<If_NRM_RMC> getAllif_NRM_RMCIstances(boolean transitive) {  
    Collection<If_NRM_RMC> result = new ArrayList<If_NRM_RMC>();  
    final RDFSNamedClass cls = getif_NRM_RMCClass();  
    RDFResource owlIndividual;  
    for (Iterator it = cls.getInstances(transitive).iterator(); it.hasNext();) {  
        owlIndividual = (RDFResource) it.next();  
        result.add(new ImplIf_NRM_RMC(owlModel, owlIndividual.getFrameID()));  
    }  
    return result;  
}
```

```
public RDFSNamedClass getif_NRM_RRCClass() {
    final String uri = "http://www.semanticweb.org/ontologies/2009/8/p1900.owl#if_NRM_RRC";
    final String name = owlModel.getResourceNameForURI(uri);
    return owlModel.getRDFSNamedClass(name);
}

public If_NRM_RRC createif_NRM_RRC(String name) {
    final RDFSNamedClass cls = getif_NRM_RRCClass();
    if (name == null) {
        name = owlModel.getNextAnonymousResourceName();
    }
    return new ImplIf_NRM_RRC(owlModel, cls.createInstance(name).getFrameID());
}

public If_NRM_RRC getif_NRM_RRC(String name) {
    RDFResource res = owlModel.getRDFResource(OWLUtil.getInternalFullName(owlModel, name));
    if (res == null) {return null;}
    if (res instanceof If_NRM_RRC) {
        return (If_NRM_RRC) res;
    } else if (res.hasProtegeType(getif_NRM_RRCClass())) {
        return new ImplIf_NRM_RRC(owlModel, res.getFrameID());
    }
    return null;
}
```

```
public Collection<If_NRM_RRC> getAllIf_NRM_RRCInstances() {
    return getAllIf_NRM_RRCInstances(false);
}

public Collection<If_NRM_RRC> getAllIf_NRM_RRCInstances(boolean transitive) {
    Collection<If_NRM_RRC> result = new ArrayList<If_NRM_RRC>();
    final RDFSNamedClass cls = getIf_NRM_RRCClass();
    RDFResource owlIndividual;
    for (Iterator it = cls.getInstances(transitive).iterator(); it.hasNext();) {
        owlIndividual = (RDFResource) it.next();
        result.add(new ImplIf_NRM_RRC(owlModel, owlIndividual.getFrameID()));
    }
    return result;
}

public RDFSNamedClass getIf_NRM_TRMClass() {
    final String uri = "http://www.semanticweb.org/ontologies/2009/8/p1900.owl#if_NRM_TRM";
    final String name = owlModel.getResourceNameForURI(uri);
    return owlModel.getRDFSNamedClass(name);
}

public If_NRM_TRM createIf_NRM_TRM(String name) {
    final RDFSNamedClass cls = getIf_NRM_TRMClass();
    if (name == null) {
```

```

        name = owlModel.getNextAnonymousResourceName();
    }

    return new ImplIf_NRM_TRM(owlModel, cls.createInstance(name).getFrameID());
}

public If_NRM_TRM getIf_NRM_TRM(String name) {
    RDFResource res = owlModel.getRDFResource(OWLUtil.getInternalFullName(owlModel, name));
    if (res == null) {return null;}
    if (res instanceof If_NRM_TRM) {
        return (If_NRM_TRM) res;
    } else if (res.hasProtegeType(getIf_NRM_TRMClass())) {
        return new ImplIf_NRM_TRM(owlModel, res.getFrameID());
    }
    return null;
}

public Collection<If_NRM_TRM> getAllIf_NRM_TRMInstances() {
    return getAllIf_NRM_TRMInstances(false);
}

public Collection<If_NRM_TRM> getAllIf_NRM_TRMInstances(boolean transitive) {
    Collection<If_NRM_TRM> result = new ArrayList<If_NRM_TRM>();
    final RDFSNamedClass cls = getIf_NRM_TRMClass();
    RDFResource owlIndividual;
    for (Iterator it = cls.getInstances(transitive).iterator(); it.hasNext();) {
        owlIndividual = (RDFResource) it.next();
    }
}

```

```
        result.add(new ImplIf_NRM_TRM(owlModel, owlIndividual.getFrameID()));
    }
    return result;
}

public RDFSNamedClass getif_TRM_TMClass() {
    final String uri = "http://www.semanticweb.org/ontologies/2009/8/p1900.owl#if_TRM_TMC";
    final String name = owlModel.getResourceNameForURI(uri);
    return owlModel.getRDFSNamedClass(name);
}

public If_TRM_TMC createif_TRM_TMC(String name) {
    final RDFSNamedClass cls = getif_TRM_TMClass();
    if (name == null) {
        name = owlModel.getNextAnonymousResourceName();
    }
    return new ImplIf_TRM_TMC(owlModel, cls.createInstance(name).getFrameID());
}

public If_TRM_TMC getif_TRM_TMC(String name) {
    RDFResource res = owlModel.getRDFResource(OWLUtil.getInternalFullName(owlModel, name));
    if (res == null) {return null;}
    if (res instanceof If_TRM_TMC) {
        return (If_TRM_TMC) res;
    } else if (res.hasProtegeType(getif_TRM_TMClass())) {
```



```
        return new ImplIf_TRM_TMC(owlModel, res.getFrameID());
    }
    return null;
}

public Collection<If_TRM_TMC> getAllIf_TRM_TMCInstances() {
    return getAllIf_TRM_TMCInstances(false);
}

public Collection<If_TRM_TMC> getAllIf_TRM_TMCInstances(boolean transitive) {
    Collection<If_TRM_TMC> result = new ArrayList<If_TRM_TMC>();
    final RDFSNamedClass cls = getIf_TRM_TMCClass();
    RDFResource owlIndividual;
    for (Iterator it = cls.getInstances(transitive).iterator(); it.hasNext();) {
        owlIndividual = (RDFResource) it.next();
        result.add(new ImplIf_TRM_TMC(owlModel, owlIndividual.getFrameID()));
    }
    return result;
}

public RDFSNamedClass getIf_TRM_TRCClass() {
    final String uri = "http://www.semanticweb.org/ontologies/2009/8/p1900.owl#if_TRM_TRC";
    final String name = owlModel.getResourceNameForURI(uri);
    return owlModel.getRDFSNamedClass(name);
}
```

```
public If_TRM_TRC createIf_TRM_TRC(String name) {
    final RDFSNamedClass cls = getIf_TRM_TRCClass();
    if (name == null) {
        name = owlModel.getNextAnonymousResourceName();
    }
    return new ImplIf_TRM_TRC(owlModel, cls.createInstance(name).getFrameID());
}

public If_TRM_TRC getIf_TRM_TRC(String name) {
    RDFResource res = owlModel.getRDFResource(OWLUtil.getInternalFullName(owlModel, name));
    if (res == null) {return null;}
    if (res instanceof If_TRM_TRC) {
        return (If_TRM_TRC) res;
    } else if (res.hasProtegeType(getIf_TRM_TRCClass())) {
        return new ImplIf_TRM_TRC(owlModel, res.getFrameID());
    }
    return null;
}

public Collection<If_TRM_TRC> getAllIf_TRM_TRCInstances() {
    return getAllIf_TRM_TRCInstances(false);
}

public Collection<If_TRM_TRC> getAllIf_TRM_TRCInstances(boolean transitive) {
    Collection<If_TRM_TRC> result = new ArrayList<If_TRM_TRC>();
```

```
final RDFSNamedClass cls = getif_TRM_TRCClass();

RDFResource owlIndividual;

for (Iterator it = cls.getInstances(transitive).iterator(); it.hasNext();) {
    owlIndividual = (RDFResource) it.next();
    result.add(new ImplIf_TRM_TRC(owlModel, owlIndividual.getFrameID()));
}

return result;
}

public RDFSNamedClass getinterfaceClass() {
    final String uri = "http://www.semanticweb.org/ontologies/2009/8/p1900.owl#interface";
    final String name = owlModel.getResourceNameForURI(uri);
    return owlModel.getRDFSNamedClass(name);
}

public Interf createinterface(String name) {
    final RDFSNamedClass cls = getinterfaceClass();
    if (name == null) {
        name = owlModel.getNextAnonymousResourceName();
    }
    return new ImplInterface(owlModel, cls.createInstance(name).getFrameID());
}

public Interf getinterface(String name) {
    RDFResource res = owlModel.getRDFResource(OWLUtil.getInternalFullName(owlModel, name));
```

```
if (res == null) {return null;}

if (res instanceof Interf) {
    return (Interf) res;
} else if (res.hasProtegeType(getinterfaceClass())) {
    return new ImplInterface(owlModel, res.getFrameID());
}
return null;
}

public Collection<Interf> getAllinterfaceInstances() {
    return getAllinterfaceInstances(false);
}

public Collection<Interf> getAllinterfaceInstances(boolean transitive) {
    Collection<Interf> result = new ArrayList<Interf>();
    final RDFSNamedClass cls = getinterfaceClass();
    RDFResource owlIndividual;
    for (Iterator it = cls.getInstance(transitive).iterator(); it.hasNext();) {
        owlIndividual = (RDFResource) it.next();
        result.add(new ImplInterface(owlModel, owlIndividual.getFrameID()));
    }
    return result;
}

public RDFSNamedClass getLocationClass() {
```

```
final String uri = "http://www.semanticweb.org/ontologies/2009/8/p1900.owl#location";

final String name = owlModel.getResourceNameForURI(uri);

return owlModel.getRDFSNamedClass(name);
}

public Location createlocation(String name) {
    final RDFSNamedClass cls = getlocationClass();
    if (name == null) {
        name = owlModel.getNextAnonymousResourceName();
    }
    return new ImplLocation(owlModel, cls.createInstance(name).getFrameID());
}

public Location getlocation(String name) {
    RDFResource res = owlModel.getRDFResource(OWLUtil.getInternalFullName(owlModel, name));
    if (res == null) {return null;}
    if (res instanceof Location) {
        return (Location) res;
    } else if (res.hasProtegeType(getlocationClass())) {
        return new ImplLocation(owlModel, res.getFrameID());
    }
    return null;
}

public Collection<Location> getAlllocationInstances() {
    return getAlllocationInstances(false);
}
```

```
}
```

```
public Collection<Location> getAlllocationInstances(boolean transitive) {  
    Collection<Location> result = new ArrayList<Location>();  
    final RDFSNamedClass cls = getLocationClass();  
    RDFResource owlIndividual;  
    for (Iterator it = cls.getInstanceS(transitive).iterator(); it.hasNext();) {  
        owlIndividual = (RDFResource) it.next();  
        result.add(new ImplLocation(owlModel, owlIndividual.getFrameID()));  
    }  
    return result;  
}
```

```
public RDFSNamedClass getmeasurementClass() {  
    final String uri = "http://www.semanticweb.org/ontologies/2009/8/p1900.owl#measurement";  
    final String name = owlModel.getResourceNameForURI(uri);  
    return owlModel.getRDFSNamedClass(name);  
}
```

```
public Measurement createmeasurement(String name) {  
    final RDFSNamedClass cls = getmeasurementClass();  
    if (name == null) {  
        name = owlModel.getNextAnonymousResourceName();  
    }  
    return new ImplMeasurement(owlModel, cls.createInstance(name).getFrameID());  
}
```

```
}
```

```
public Measurement getmeasurement(String name) {  
    RDFResource res = owlModel.getRDFResource(OWLUtil.getInternalFullName(owlModel, name));  
    if (res == null) {return null;}  
    if (res instanceof Measurement) {  
        return (Measurement) res;  
    } else if (res.hasProtegeType(getmeasurementClass())) {  
        return new ImplMeasurement(owlModel, res.getFrameID());  
    }  
    return null;  
}
```

```
public Collection<Measurement> getAllmeasurementInstances() {  
    return getAllmeasurementInstances(false);  
}
```

```
public Collection<Measurement> getAllmeasurementInstances(boolean transitive) {  
    Collection<Measurement> result = new ArrayList<Measurement>();  
    final RDFSNamedClass cls = getmeasurementClass();  
    RDFResource owlIndividual;  
    for (Iterator it = cls.getInstances(transitive).iterator(); it.hasNext();) {  
        owlIndividual = (RDFResource) it.next();  
        result.add(new ImplMeasurement(owlModel, owlIndividual.getFrameID()));  
    }  
    return result;  
}
```

```
}

public RDFSNamedClass getrCFG_MEDIA_SAPClass() {
    final String uri =
"http://www.semanticweb.org/ontologies/2009/8/p1900.owl#rCFG_MEDIA_SAP";
    final String name = owlModel.getResourceNameForURI(uri);
    return owlModel.getRDFSNamedClass(name);
}

public rCFG_MEDIA_SAP creatorCFG_MEDIA_SAP(String name) {
    final RDFSNamedClass cls = getrCFG_MEDIA_SAPClass();
    if (name == null) {
        name = owlModel.getNextAnonymousResourceName();
    }
    return new ImplrCFG_MEDIA_SAP(owlModel, cls.createInstance(name).getFrameID());
}

public rCFG_MEDIA_SAP getrCFG_MEDIA_SAP(String name) {
    RDFResource res = owlModel.getRDFResource(OWLUtil.getInternalFullName(owlModel, name));
    if (res == null) {return null;}
    if (res instanceof rCFG_MEDIA_SAP) {
        return (rCFG_MEDIA_SAP) res;
    } else if (res.hasProtegeType(getrCFG_MEDIA_SAPClass())) {
        return new ImplrCFG_MEDIA_SAP(owlModel, res.getFrameID());
    }
}
```



```
return null;
}

public Collection<rCFG_MEDIA_SAP> getAllrCFG_MEDIA_SAPInstances() {
    return getAllrCFG_MEDIA_SAPInstances(false);
}

public Collection<rCFG_MEDIA_SAP> getAllrCFG_MEDIA_SAPInstances(boolean transitive) {
    Collection<rCFG_MEDIA_SAP> result = new ArrayList<rCFG_MEDIA_SAP>();
    final RDFSNamedClass cls = getrCFG_MEDIA_SAPClass();
    RDFResource owlIndividual;
    for (Iterator it = cls.getInstances(transitive).iterator(); it.hasNext();) {
        owlIndividual = (RDFResource) it.next();
        result.add(new ImplrCFG_MEDIA_SAP(owlModel, owlIndividual.getFrameID()));
    }
    return result;
}

public RDFSNamedClass getrCFG_MNG_SAPClass() {
    final String uri =
"http://www.semanticweb.org/ontologies/2009/8/p1900.owl#rCFG_MNG_SAP";
    final String name = owlModel.getResourceNameForURI(uri);
    return owlModel.getRDFSNamedClass(name);
}
```

```
public rCFG_MNG_SAP createrCFG_MNG_SAP(String name) {
    final RDFSNamedClass cls = getrCFG_MNG_SAPClass();
    if (name == null) {
        name = owlModel.getNextAnonymousResourceName();
    }
    return new ImplrCFG_MNG_SAP(owlModel, cls.createInstance(name).getFrameID());
}

public rCFG_MNG_SAP getrCFG_MNG_SAP(String name) {
    RDFResource res = owlModel.getRDFResource(OWLUtil.getInternalFullName(owlModel, name));
    if (res == null) {return null;}
    if (res instanceof rCFG_MNG_SAP) {
        return (rCFG_MNG_SAP) res;
    } else if (res.hasProtegeType(getrCFG_MNG_SAPClass())) {
        return new ImplrCFG_MNG_SAP(owlModel, res.getFrameID());
    }
    return null;
}

public Collection<rCFG_MNG_SAP> getAllrCFG_MNG_SAPInstances() {
    return getAllrCFG_MNG_SAPInstances(false);
}

public Collection<rCFG_MNG_SAP> getAllrCFG_MNG_SAPInstances(boolean transitive) {
    Collection<rCFG_MNG_SAP> result = new ArrayList<rCFG_MNG_SAP>();
    final RDFSNamedClass cls = getrCFG_MNG_SAPClass();
```

```
RDFResource owlIndividual;

for (Iterator it = cls.getInstances(transitive).iterator();it.hasNext()); {

    owlIndividual = (RDFResource) it.next();

    result.add(new ImplrCFG_MNG_SAP(owlModel, owlIndividual.getFrameID()));

}

return result;

}

public RDFSNamedClass getrCFG_TR_SAPClass() {

    final String uri = "http://www.semanticweb.org/ontologies/2009/8/p1900.owl#rCFG_TR_SAP";

    final String name = owlModel.getResourceNameForURI(uri);

    return owlModel.getRDFSNamedClass(name);

}

public rCFG_TR_SAP creatorCFG_TR_SAP(String name) {

    final RDFSNamedClass cls = getrCFG_TR_SAPClass();

    if (name == null) {

        name = owlModel.getNextAnonymousResourceName();

    }

    return new ImplrCFG_TR_SAP(owlModel, cls.createInstance(name).getFrameID());

}

public rCFG_TR_SAP getrCFG_TR_SAP(String name) {

    RDFResource res = owlModel.getRDFResource(OWLUtil.getInternalFullName(owlModel, name));

    if (res == null) {return null;}

}
```

```
if (res instanceof rCFG_TR_SAP) {
    return (rCFG_TR_SAP) res;
} else if (res.hasProtegeType(getrCFG_TR_SAPClass())) {
    return new ImplrCFG_TR_SAP(owlModel, res.getFrameID());
}
return null;
}

public Collection<rCFG_TR_SAP> getAllrCFG_TR_SAPInstances() {
    return getAllrCFG_TR_SAPInstances(false);
}

public Collection<rCFG_TR_SAP> getAllrCFG_TR_SAPInstances(boolean transitive) {
    Collection<rCFG_TR_SAP> result = new ArrayList<rCFG_TR_SAP>();
    final RDFSNamedClass cls = getrCFG_TR_SAPClass();
    RDFResource owlIndividual;
    for (Iterator it = cls.getInstances(transitive).iterator(); it.hasNext();) {
        owlIndividual = (RDFResource) it.next();
        result.add(new ImplrCFG_TR_SAP(owlModel, owlIndividual.getFrameID()));
    }
    return result;
}

public RDFSNamedClass getterminalCapabilityClass() {
```

```
final String uri =
"http://www.semanticweb.org/ontologies/2009/8/p1900.owl#terminalCapability";

final String name = owlModel.getResourceNameForURI(uri);

return owlModel.getRDFSNamedClass(name);
}

public TerminalCapability createterminalCapability(String name) {
final RDFSNamedClass cls = getterminalCapabilityClass();

if (name == null) {
name = owlModel.getNextAnonymousResourceName();
}

return new ImplTerminalCapability(owlModel, cls.createInstance(name).getFrameID());
}

public TerminalCapability getterminalCapability(String name) {
RDFResource res = owlModel.getRDFResource(OWLUtil.getInternalFullName(owlModel, name));

if (res == null) {return null;}

if (res instanceof TerminalCapability) {
return (TerminalCapability) res;
} else if (res.hasProtegeType(getterminalCapabilityClass())) {
return new ImplTerminalCapability(owlModel, res.getFrameID());
}

return null;
}

public Collection<TerminalCapability> getAllterminalCapabilityInstances() {
```

```
return getAllterminalCapabilityInstances(false);
}

public Collection<TerminalCapability> getAllterminalCapabilityInstances(boolean transitive) {
    Collection<TerminalCapability> result = new ArrayList<TerminalCapability>();
    final RDFSNamedClass cls = getterminalCapabilityClass();
    RDFResource owlIndividual;
    for (Iterator it = cls.getInstances(transitive).iterator(); it.hasNext();) {
        owlIndividual = (RDFResource) it.next();
        result.add(new ImplTerminalCapability(owlModel, owlIndividual.getFrameID()));
    }
    return result;
}

public RDFSNamedClass getuserPreferenceClass() {
    final String uri =
"http://www.semanticweb.org/ontologies/2009/8/p1900.owl#userPreference";
    final String name = owlModel.getResourceNameForURI(uri);
    return owlModel.getRDFSNamedClass(name);
}

public UserPreference createuserPreference(String name) {
    final RDFSNamedClass cls = getuserPreferenceClass();
    if (name == null) {
        name = owlModel.getNextAnonymousResourceName();
    }
}
```

```
}  
  
return new ImplUserPreference(owlModel, cls.createInstance(name).getFrameID());  
}  
  
public UserPreference getuserPreference(String name) {  
    RDFResource res = owlModel.getRDFResource(OWLUtil.getInternalFullName(owlModel, name));  
    if (res == null) {return null;}  
    if (res instanceof UserPreference) {  
        return (UserPreference) res;  
    } else if (res.hasProtegeType(getuserPreferenceClass())) {  
        return new ImplUserPreference(owlModel, res.getFrameID());  
    }  
    return null;  
}  
  
public Collection<UserPreference> getAlluserPreferenceInstances() {  
    return getAlluserPreferenceInstances(false);  
}  
  
public Collection<UserPreference> getAlluserPreferenceInstances(boolean transitive) {  
    Collection<UserPreference> result = new ArrayList<UserPreference>();  
    final RDFSNamedClass cls = getuserPreferenceClass();  
    RDFResource owlIndividual;  
    for (Iterator it = cls.getInstances(transitive).iterator(); it.hasNext();) {  
        owlIndividual = (RDFResource) it.next();  
        result.add(new ImplUserPreference(owlModel, owlIndividual.getFrameID()));  
    }  
}
```

```
}  
  
return result;  
  
}  
  
public RDFProperty getAcquireContextProperty() {  
    final String uri = "http://www.semanticweb.org/ontologies/2009/8/p1900.owl#acquireContext";  
    final String name = owlModel.getResourceNameForURI(uri);  
    return owlModel.getRDFProperty(name);  
}  
  
public RDFProperty getAcquireReconfigurationDecisionProperty() {  
    final String uri =  
"http://www.semanticweb.org/ontologies/2009/8/p1900.owl#acquireReconfigurationDecision";  
    final String name = owlModel.getResourceNameForURI(uri);  
    return owlModel.getRDFProperty(name);  
}  
  
public RDFProperty getAssignPoliciesProperty() {  
    final String uri = "http://www.semanticweb.org/ontologies/2009/8/p1900.owl#assignPolicies";  
    final String name = owlModel.getResourceNameForURI(uri);  
    return owlModel.getRDFProperty(name);  
}
```



```
public RDFProperty getHasConfigurationProperty() {  
    final String uri =  
"http://www.semanticweb.org/ontologies/2009/8/p1900.owl#hasConfiguration";  
    final String name = owlModel.getResourceNameForURI(uri);  
    return owlModel.getRDFProperty(name);  
}  
  
public RDFProperty getHasRadioInterfaceProperty() {  
    final String uri =  
"http://www.semanticweb.org/ontologies/2009/8/p1900.owl#hasRadioInterface";  
    final String name = owlModel.getResourceNameForURI(uri);  
    return owlModel.getRDFProperty(name);  
}  
  
public RDFProperty getHasSpectrumProperty() {  
    final String uri = "http://www.semanticweb.org/ontologies/2009/8/p1900.owl#hasSpectrum";  
    final String name = owlModel.getResourceNameForURI(uri);  
    return owlModel.getRDFProperty(name);  
}  
  
public RDFProperty getIsReconfigurableProperty() {  
    final String uri =  
"http://www.semanticweb.org/ontologies/2009/8/p1900.owl#isReconfigurable";
```

```
final String name = owlModel.getResourceNameForURI(uri);

return owlModel.getRDFProperty(name);

}

public RDFProperty getTakeDecisionProperty() {

    final String uri = "http://www.semanticweb.org/ontologies/2009/8/p1900.owl#takeDecision";

    final String name = owlModel.getResourceNameForURI(uri);

    return owlModel.getRDFProperty(name);

}

public RDFProperty getOnClassProperty() {

    final String uri = "http://www.w3.org/2002/07/owl#onClass";

    final String name = owlModel.getResourceNameForURI(uri);

    return owlModel.getRDFProperty(name);

}

public RDFProperty getQualifiedCardinalityProperty() {

    final String uri = "http://www.w3.org/2002/07/owl#qualifiedCardinality";

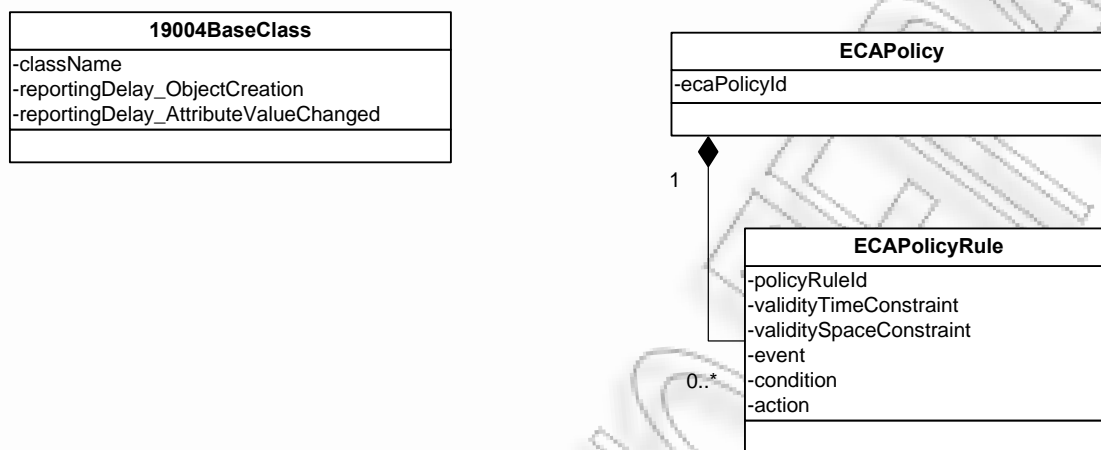
    final String name = owlModel.getResourceNameForURI(uri);

    return owlModel.getRDFProperty(name);

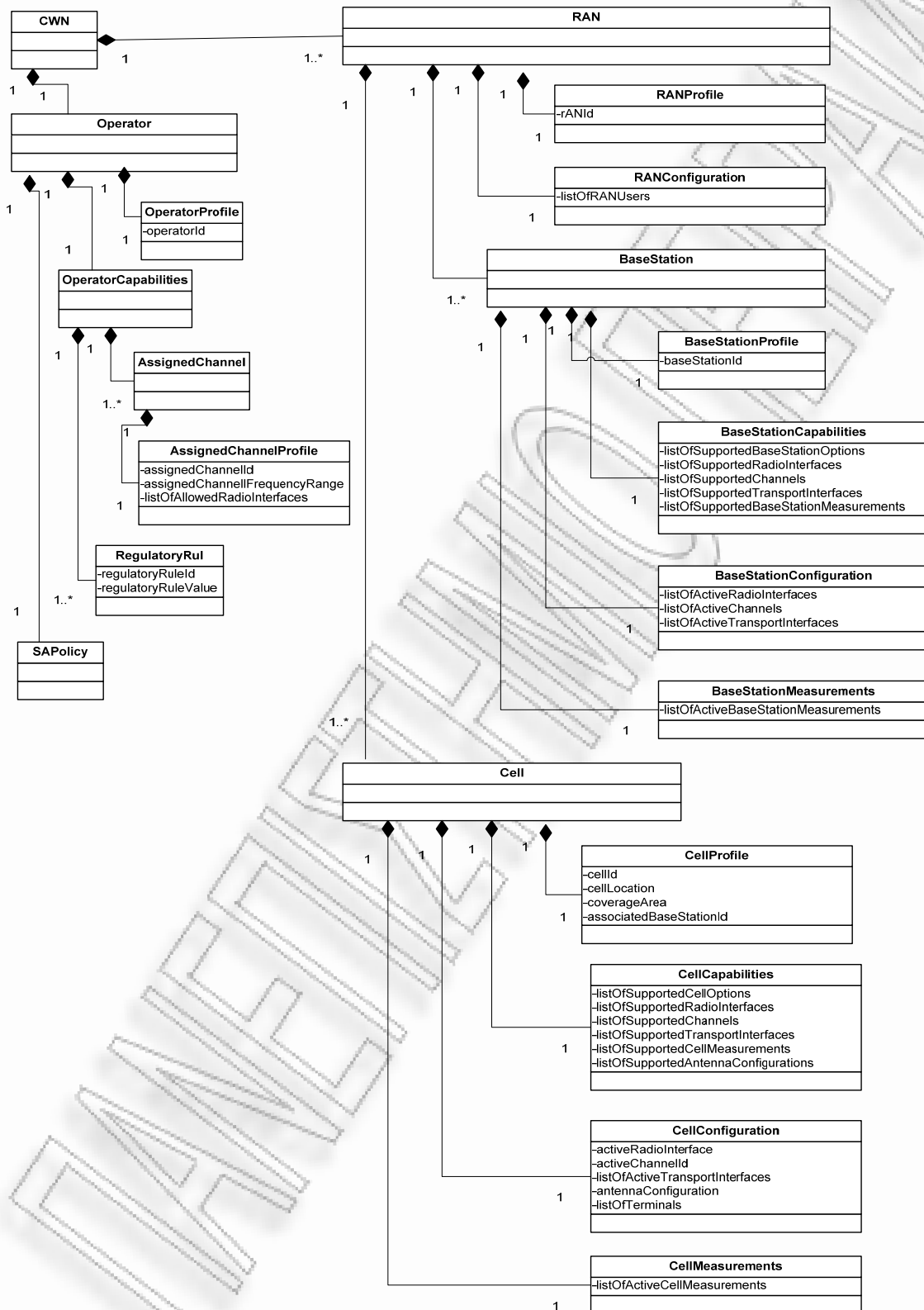
}

}
```

## 8.2 Διάγραμμα κλάσεων της οντολογίας P1900



Εικόνα 8-1: UML class diagram: κοινές κλάσεις του μοντέλου

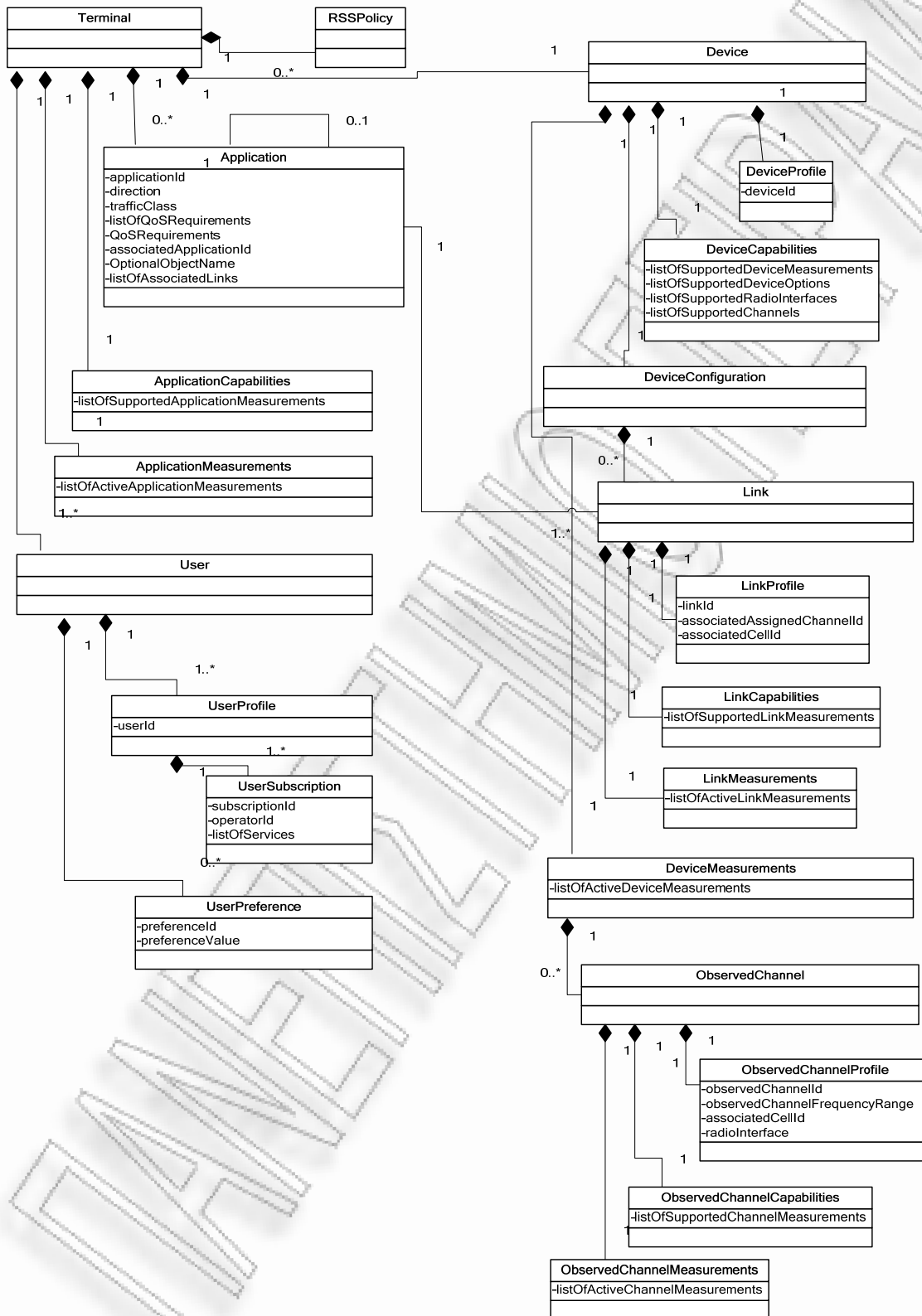


Εικόνα 8-2: UML class diagram: κλάσεις υλοποίησης της διαχείρισης του Σύνθετου  
Ασύρματου Δικτύου

Συσχέτιση μεταξύ των κλάσεων

- CWN: αντικείμενα της κλάσης αυτής περιέχουν αντικείμενα όλων των κλάσεων της οντότητας NRM όπως:
  - Operator: κάθε αντικείμενο της κλάσης CWN έχει ένα μόνο αντικείμενο της κλάσης Operator.
  - OperatorProfile: κάθε αντικείμενο της κλάσης Operator έχει ένα μόνο αντικείμενο της κλάσης OperatorProfile.
  - OperatorCapabilities: κάθε αντικείμενο της κλάσης Operator έχει ένα μόνο αντικείμενο της κλάσης OperatorCapabilities.
  - AssignedChannel: κάθε αντικείμενο της κλάσης Operator έχει ένα μόνο αντικείμενο της κλάσης AssignedChannel.
    - AssignedChannelProfile: κάθε αντικείμενο της κλάσης AssignedChannel έχει ένα μόνο αντικείμενο της κλάσης AssignedChannelProfile.
  - RegulatoryRule: κάθε αντικείμενο της κλάσης OperatorCapabilities έχει ένα ή περισσότερα αντικείμενα της κλάσης RegulatoryRule.
  - SAPolicy: κάθε αντικείμενο της κλάσης Operator έχει μηδέν ή περισσότερα αντικείμενα της κλάσης SAPolicy.
- RAN: κάθε αντικείμενο της κλάσης CWN έχει ένα ή περισσότερα αντικείμενα της κλάσης RAN.
  - RANProfile: κάθε αντικείμενο της κλάσης RAN έχει ένα μόνο αντικείμενο της κλάσης RANProfile.
  - RANConfiguration: κάθε αντικείμενο της κλάσης RAN έχει ένα μόνο αντικείμενο της κλάσης RANConfiguration.
  - BaseStation: κάθε αντικείμενο της κλάσης RAN έχει ένα ή περισσότερα αντικείμενα της κλάσης BaseStation.
    - BaseStationProfile: κάθε αντικείμενο της κλάσης BaseStation έχει ένα μόνο αντικείμενο της κλάσης BaseStationProfile.
    - BaseStationConfiguration: κάθε αντικείμενο της κλάσης BaseStation έχει ένα μόνο αντικείμενο της κλάσης BaseStationConfiguration.

- BaseStationMeasurements: κάθε αντικείμενο της κλάσης BaseStation έχει ένα μόνο αντικείμενο της κλάσης BaseStationMeasurements.
- Cell: κάθε αντικείμενο της κλάσης CWN έχει ένα ή περισσότερα αντικείμενα της κλάσης Cell.
  - CellProfile: κάθε αντικείμενο της κλάσης Cell έχει ένα μόνο αντικείμενο της κλάσης CellProfile.
  - CellCapabilities: κάθε αντικείμενο της κλάσης Cell έχει ένα μόνο αντικείμενο της κλάσης CellCapabilities.
  - CellConfiguration: κάθε αντικείμενο της κλάσης Cell έχει ένα μόνο αντικείμενο της κλάσης CellConfiguration.
  - CellMeasurements: κάθε αντικείμενο της κλάσης Cell έχει ένα μόνο αντικείμενο της κλάσης CellMeasurements.



Εικόνα 8-3: UML class diagram: κλάσεις υλοποίησης της διαχείρισης τερματικών

### Συσχέτιση μεταξύ των κλάσεων

- Terminal: αντικείμενα της κλάσης αυτής περιέχουν αντικείμενα όλων των κλάσεων της οντότητας TRM όπως:
- User: κάθε αντικείμενο της κλάσης Terminal έχει ένα ή περισσότερα αντικείμενα της κλάσης User.
  - UserProfile: κάθε αντικείμενο της κλάσης User έχει ένα μόνο αντικείμενο της κλάσης UserProfile.
  - UserSubscription: κάθε αντικείμενο της κλάσης UserProfile έχει ένα ή περισσότερα αντικείμενα της κλάσης UserSubscription.
  - UserPreference: κάθε αντικείμενο της κλάσης User έχει μηδέν ή περισσότερα αντικείμενα της κλάσης UserPreference.
- Application: κάθε αντικείμενο της κλάσης Terminal έχει μηδέν ή περισσότερα αντικείμενα της κλάσης Application.
  - ApplicationProfile: κάθε αντικείμενο της κλάσης Application έχει ένα μόνο αντικείμενο της κλάσης ApplicationProfile.
  - ApplicationCapabilities: κάθε αντικείμενο της κλάσης Application έχει ένα μόνο αντικείμενο της κλάσης ApplicationCapabilities.
  - ApplicationMeasurements: κάθε αντικείμενο της κλάσης Application έχει ένα μόνο αντικείμενο της κλάσης ApplicationMeasurements.
- Device: κάθε αντικείμενο της κλάσης Terminal έχει ένα μόνο αντικείμενο της κλάσης Device.
  - DeviceProfile: κάθε αντικείμενο της κλάσης Device έχει ένα μόνο αντικείμενο της κλάσης DeviceProfile.
  - DeviceConfiguration: κάθε αντικείμενο της κλάσης Device έχει ένα μόνο αντικείμενο της κλάσης DeviceConfiguration.
    - Link: κάθε αντικείμενο της κλάσης DeviceConfiguration έχει μηδέν ή περισσότερα αντικείμενα της κλάσης Link.
      - LinkProfile: κάθε αντικείμενο της κλάσης Link έχει ένα μόνο αντικείμενο της κλάσης LinkProfile.
      - LinkCapabilities: κάθε αντικείμενο της κλάσης Link έχει ένα μόνο αντικείμενο της κλάσης LinkCapabilities.
      - LinkMeasurements: κάθε αντικείμενο της κλάσης Link έχει ένα μόνο αντικείμενο της κλάσης LinkMeasurements.



- DeviceMeasurements: κάθε αντικείμενο της κλάσης Device έχει ένα μόνο αντικείμενο της κλάσης DeviceMeasurements.
  - ObservedChannel: κάθε αντικείμενο της κλάσης DeviceMeasurements έχει μηδέν ή περισσότερα αντικείμενα της κλάσης ObservedChannel.
    - ObservedChannelProfile: κάθε αντικείμενο της κλάσης ObservedChannel έχει ένα αντικείμενο της κλάσης ObservedChannelProfile.
    - ObservedChannelCapabilities: κάθε αντικείμενο της κλάσης ObservedChannel έχει ένα αντικείμενο της κλάσης ObservedChannelCapabilities.
    - ObservedChannelMeasurements: κάθε αντικείμενο της κλάσης ObservedChannel έχει ένα αντικείμενο της κλάσης ObservedChannelMeasurements.
- RSSPolicy: κάθε αντικείμενο της κλάσης Terminal έχει μηδέν ή περισσότερα αντικείμενα της κλάσης RSSPolicy.

### 8.3 Υλοποίηση κλάσεων του προτεινόμενου μοντέλου

#### Κλάση NRM

```
private Vector<Terminal> Terminals;  
private Vector<RAN> RANs;  
  
private Measurement GetMeasurements(RAN someRAN);  
private Context GetContext(RAN someRAN);  
private Measurement GetMeasurements(Terminal someTerminal);  
private Context GetContext(Terminal someTerminal);  
  
private Object OptimumAllocation(Vector<RAN> someRANs, Vector<Terminal>  
someTerminals);  
  
private Object CreatePolicy(Object OptimumAllocationResult);
```

```
private boolean SendPolicy(RAN someRAN);  
  
private void SendPolicyToRepository(Object CreatedPolicy, Date CurrentDateTime);  
  
private void SendNetworkRecalculationRequest(TRM someTRM);
```

#### Κλάση OSM

```
private Vector<RAN> RANs; //rans that have only one operator
```

#### Κλάση CWN

```
private Vector<RAN> RANs;  
  
private Vector<Terminal> Terminals;  
  
private Location Region;  
  
private Date OperationDateTime;  
  
private Vector<Operator> Operators; //the operator(s) that manage this CWN
```

#### Κλάση RRC

```
private void ConfigurationConfirmation(boolean Confirm, RAN someRAN);
```

#### Κλάση RMC

```
private Vector<RAN> RANs; // the rans this rmc manages  
  
private void SendContext(RAN SomeRAN, NRM someNRM);
```

#### Κλάση TRM

```
private Vector<Terminal> Terminals;
```



### Κλάση Operator

```
public string OperatorIP;

private Vector<Operator> Operators; //his fellow operators

private Vector<Object> Arrangements;

private Vector<CWN> CWNs; //the CWNs that this Operator manages

private Vector<RAN> RANs; //the RANs that this operator manages

private Vector<Object> GetArrangements(Operator someOperator);

private boolean SendArrangement(Operator someOperator);
```

### Κλάση Context

```
public Vector<Object> SpectrumUsageRights;

public Vector<Object> SpectrumUtilizationEvaluation;

public Vector<Object> FrequencyBands;
```

### Κλάση Terminal

```
private RAN TerminalRAN; //the RAN to which the terminal is connected

private Vector<RAN> RANs;

private Operator TerminalOperator;

private Timer UpdateNRM; //for periodic update of the NRM

private Location TerminalLocation;

private TRM TerminalTRM;

private int Channel;

private TerminalCapability thisTerminalCapability;

private User TerminalUser;
```

```
private Vector<Object> SupportedInterfaces;

private Location GetLocation();

private Vector<Operator> ScanForNetworks(Location MyLocation, Time MyTime);

private int RANRank(QoS PreferredQoS, string Application);

private RAN Connect(Vector<Operator> FoundOperators); //connects to the best RAN

public TerminalContext(Object RAT, Location CurrentLocation, User.thisUserPreference
Preferences, Object MobilityProfile);

private void SetChannel(int SomeChannel);

private SendNRMUpdate(Object TerminalContext, NRM someNRM);
```

#### **Κλάση Location**

```
private float Latitude;

private float Longitude;

private Date CurrentTime;

private float GetLatitude(string Element);

private float GetLongitude(string Element);

private int GetCellID(string Element);

private Time GetCurrentTime();

public Location GetLocation(boolean Absolute);
```

#### **Κλάση TerminalCapability**

```
public boolean MultiMode;

public boolean MultiHoming;
```

## Κλάση SAP

```
public Vector<Object> CreateMobilityProfile(Terminal SomeTerminal);  
public Vector<Object> EvaluateSpectrumUtilization(RAN SomeRAN);  
public float CalculateThroughput(RAN SomeRAN);  
public float CalculateThroughput(CWN SomeCWN);  
public float CalculateSNR(RAN SomeRAN);  
public float CalculateTrafficLoad(RAN SomeRAN);  
public float CalculateTrafficLoad(CWN SomeCWN);  
public int NumberOfTerminals(RAN SomeRAN);  
public int NumberOfTerminals(CWN SomeCWN);  
public float CalculateBIER(RAN SomeRAN);  
public float CalculateBIER(CWN SomeCWN);  
public float CalculatePacketLoss(RAN SomeRAN);  
public float CalculatePacketLoss(CWN SomeCWN);  
public float CalculatePacketLoss(Terminal SomeTerminal);  
public float CalculateLatency(RAN SomeRAN);  
public float CalculateLatency(CWN SomeCWN);  
public float GetBER(Terminal SomeTerminal);  
public Vector<Object> CreatePerformanceReport(Terminal SomeTerminal);  
public Vector<Object> CreateResourceReport(Terminal SomeTerminal);  
public float GetRadioPower(Terminal SomeTerminal);  
public float CalculateSINR(Terminal SomeTerminal);  
public float CalculateBER(Terminal SomeTerminal);  
public Object GetUserProfilesData(RAN someRAN);  
public float CalculateUtilizationDefree(RAN someRAN);  
public Object GetSpectrumHoles(RAN someRAN);
```

```
public float CalculateLoadChange(RAN someRAN, Time FromTime, Time ToTime);  
public Object CreatePolicy(Object CreatedPolicy);  
public int RANRank(RAN someRAN, string Application);
```

#### Κλάση QoS

```
public float BER;  
public Object ResourceReport;  
public float SNR;  
public float PacketLossRate;  
public float PacketErrorRate;
```

#### Κλάση Measurement

```
public float Throughput;  
public float SNR;  
public float TrafficLoad;  
public int NumberOfTerminals;  
public float BIER;  
public float PacketLossRate;  
public float Latency;  
public float RadioPower;  
public float SINR;  
public float BER;  
public Object UserProfilesData;  
public float UtilizationDegree;  
public Object SpectrumHoles;  
public float LoadChange;
```

### Κλάση User

```
public UserPreference thisUserPreference;
```

### Κλάση UserPreference

```
public Vector<Object> RATs;  
public Vector<Operator> Operators;  
public Vector<Object> Services;  
public Vector<QoS> ServicesQoS;  
public Vector<QoS> PerceivedServicesQoS;  
public List<String> Applications;  
public int CommunicationPriority;
```