

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ
ΤΜΗΜΑ ΨΗΦΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΔΙΑΧΕΙΡΙΣΗ ΤΕΡΜΑΤΙΚΩΝ ΣΕ ΠΕΡΙΒΑΛΛΟΝΤΑ ΔΙΚΤΥΩΝ ΥΨΗΛΩΝ ΤΑΧΥΤΗΤΩΝ B3G

Πτυχιακή Εργασία

Επιβλέπων: Δεμέστιχας Παναγιώτης
Αναπ. Καθηγητής Πα.Πει.

Σαρσεμπάγιεβα Αικατερίνη

E/ 04 147

Σεπτέμβριος 2008

Ευχαριστίες

Πρώτον από όλους θα ήθελα να ευχαριστήσω τον Απόστολο Κατιδιώτη, όχι μόνο επειδή του είχα πει ότι θα τον βάλω πρώτο, αλλά και επειδή αν δεν ήταν αυτός δε θα ήξερα ότι μπορώ να γράψω «Ευχαριστίες» και το τμήμα αυτό δε θα υπήρχε καν. Θέλω επίσης να τον ευχαριστήσω για όλη του τη βοήθεια και την υποστήριξη που μου προσέφερε, για τις πολύτιμες συμβουλές του και για την υπομονή του όταν του σπαταλούσα το χρόνο για τις όχι πάντα έξυπνες και σοβαρές απορίες μου.

Μέσα από την καρδιά μου ευχαριστώ και τον καθηγητή Παναγιώτη Δεμέστιχα. Εάν δεν ήταν αυτός και ο πραγματικά ενδιαφέρον τρόπος διδασκαλίας του, δε θα ανακάλυπτα ποτέ την αγάπη μου για τα δίκτυα. Το χιούμορ του και οι έξυπνες παρατηρήσεις του στο μάθημα ήταν αυτά που ανέπτυξαν την επιθυμία μου να προσέρχομαι συνεχώς στις διαλέξεις και να καταλάβω με τί θέλω να ασχοληθώ στη ζωή μου. Επίσης, θέλω να πω ένα τεράστιο ευχαριστώ για την ευκαιρία που μου έδωσε, αναθέτοντας μου την «καλύτερη πτυχιακή του σύμπαντος», και για το ότι πίστεψε στις δυνατότητές μου. Ακόμα και με μια απλή φράση όπως «Πιστεύω ότι θα τα καταφέρεις», με έκανε να συνεχίζω να προσπαθώ ακόμα και όταν τίποτα δεν έτρεχε, τα μάτια μου έκλειναν και ο καφές είχε εξαφανιστεί.

Θα ήθελα να ευχαριστήσω επίσης όλους τους φίλους μου οι οποίοι μου προσέφεραν στέγη και ασύρματο router, τους δύσκολους αυτούς μήνες που κάποιος από τους γείτονές μου έκλεισε το ασύρματό του και δεν μπορούσα να υλοποιήσω το πρόγραμμα λήψης wireless παραμέτρων. Τους ευχαριστώ πραγματικά για όλη την υποστήριξη και την υπομονή που υπέδειχναν σε όλο αυτό το διάστημα.

Τέλος, το μεγαλύτερο «ευχαριστώ» το χρωστάω στη μητέρα μου. Την ευχαριστώ από τα βάθη της ψυχής μου για το ότι επί 22 χρόνια ήταν ο μοναδικός άνθρωπος που πίστευε σε μένα. Ευχαριστώ, για όλη τη βοήθεια και την ψυχολογική υποστήριξη που μου προσέφερε σε όλο αυτό το έτος. Και την ευχαριστώ απλά, που είναι η καλύτερη μαμά σε όλο το σύμπαν.

ΠΕΡΙΕΧΟΜΕΝΑ

ΣΥΝΤΟΜΟΓΡΑΦΙΕΣ

1. ΕΙΣΑΓΩΓΗ	10
1.1 ΕΙΣΑΓΩΓΗ	10
1.2 ΜΗΔΕΝΙΚΗ ΓΕΝΙΑ ΤΕΧΝΟΛΟΓΙΑΣ (0G)	12
1.3 ΠΡΩΤΗ ΓΕΝΙΑ ΤΕΧΝΟΛΟΓΙΑΣ (1G)	14
1.4 ΔΕΥΤΕΡΗ ΓΕΝΙΑ ΤΕΧΝΟΛΟΓΙΑΣ (2G)	16
1.4.1 Μετάβαση από τα αναλογικά στα ψηφιακά συστήματα	16
1.4.2 GSM	17
1.4.3 HSCSD	17
1.4.4 cdmaONE	18
1.4.5 D – AMPS	18
1.4.6 PDC	18
1.5 2.5 ΓΕΝΙΑ ΤΕΧΝΟΛΟΓΙΑΣ (2.5G)	19
1.5.1 GPRS	19
1.6 ΤΡΙΤΗ ΓΕΝΙΑ ΤΕΧΝΟΛΟΓΙΑΣ (3G)	20
1.6.1 Στροφή στα Συστήματα Τρίτης Γενιάς	20
1.6.2 UMTS	21
1.6.2.1 Αρχιτεκτονική του UMTS	22
1.6.2.2 Τερματικά	23
1.6.2.3 Διαχείριση Ισχύος Τερματικών	24
1.6.2.3.1 Έλεγχος Ισχύος Ανοιχτού Βρόχου	25
1.6.2.3.2 Έλεγχος Ισχύος Ανοιχτού Βρόχου	25
1.6.2.3.3 Έλεγχος Ισχύος Εξωτερικού Βρόχου	25
1.6.2.4 Τυχαία Πρόσβαση Τερματικών	26

1.6.2.5 Εντοπισμός Τερματικών	26
1.6.2.5.1 Εντοπισμός Βασισμένος στην Ταυτότητα της Κυψέλης (Cell ID Based Positioning)	27
1.6.2.5.2 Εντοπισμός Βασισμένος στην Παρατηρούμενη Χρονική Διαφορά της Άφιξης (Observed Time Difference of Arrival, OTDOA)	27
1.6.2.5.3 Εντοπισμός Βασισμένος στη Συσχέτιση Πιλότων (Pilot Correlation Method)	27
1.6.2.5.4 Εντοπισμός Βασισμένος στη Συσχέτιση των Προφίλ Καθυστέρησης Ισχύος (Database Correlation Method using Power Delay Profiles)	28
1.6.2.5.5 Εντοπισμός Υποβοηθούμενος από το Global Positioning System (GPS)	28
1.6.2.6 Διαχείριση Μεταπομπής (Handover στο UMTS)	29
1.6.2.6.1 Hard Handover	29
1.6.2.6.2 Inter – System Handover	30
1.6.2.6.3 Soft, Softer Handover	30
1.6.2.7 Υπηρεσίες	33
1.6.2.7.1 Τηλεϋπηρεσίες	34
1.6.2.7.2 Υπηρεσίες Bearer	34
1.6.2.7.3 Συμπληρωματικές Υπηρεσίες	35
1.6.2.8 Ποιότητα Υπηρεσίας στο UMTS (QoS)	36
1.7 ΜΕΛΛΟΝΤΙΚΟΙ ΣΤΟΧΟΙ	38
2. ΕΥΦΥΗ ΔΙΚΤΥΑ ΠΕΡΑΝ ΤΗΣ ΤΡΙΤΗΣ ΓΕΝΙΑΣ	41
2.1 ΕΙΣΑΓΩΓΗ	41
2.2 ΣΥΝΘΕΤΗ Β3G ΡΑΔΙΟ – ΥΠΟΔΟΜΗ	43
2.3 ΑΡΧΙΤΕΚΤΟΝΙΚΗ Β3G ΔΙΚΤΥΩΝ	46
2.4 ΔΥΝΑΜΙΚΗ ΚΑΤΑΝΟΜΗ ΦΑΣΜΑΤΟΣ	50
2.4.1 Κίνητρα για Δυναμική Κατανομή Φάσματος	50

2.4.2 Αρχιτεκτονική Δικτύου για Υποστήριξη Δυναμικής Κατανομής Φάσματος	51
2.4.3 Λειτουργίες Οντοτήτων Δικτύων	53
2.4.4 Διαδικασία Κατανομής Φάσματος	54
2.5 ΔΙΑΧΕΙΡΙΣΗ ΘΕΣΗΣ ΚΑΙ PAGING ΤΕΡΜΑΤΙΚΩΝ	55
2.5.1 Περιορισμοί στην Τρέχουσα Διαχείριση Θέσης	55
2.5.2 Προταθείσες Μέθοδοι Ενημέρωσης Θέσης Χρήστη	56
2.5.3 Paging Τερματικών	58
2.6 ΔΙΑΧΕΙΡΙΣΗ ΚΙΝΗΤΙΚΟΤΗΤΑΣ	59
2.6.1 Γενικά	59
2.6.2 Αρχιτεκτονική Δικτύου για Υποστήριξη Διαχείρισης Κινητικότητας	60
2.6.3 Μεταπομπή Περιηγόμενου Τερματικού	61
2.6.4 Μεταπομπή Ενεργοποιημένη από το QoS	62
3. ΔΙΑΧΕΙΡΙΣΗ ΤΕΡΜΑΤΙΚΩΝ ΣΕ ΠΕΡΙΒΑΛΛΟΝΤΑ ΔΙΚΤΥΩΝ ΥΨΗΛΩΝ ΤΑΧΥΤΗΤΩΝ Β3G	64
3.1 RECONFIGURABILITY	64
3.2 ΕΥΦΥΗ ΑΣΥΡΜΑΤΑ ΤΕΡΜΑΤΙΚΑ	66
3.3 ΠΡΟΚΛΗΣΕΙΣ ΔΙΑΧΕΙΡΙΣΗΣ ΤΕΡΜΑΤΙΚΩΝ	68
3.4 ΠΑΡΑΜΕΤΡΟΙ ΚΑΙ ΠΟΛΙΤΙΚΕΣ	69
3.4.1 Παράμετροι Συσκευής	69
3.4.2 Προφίλ Χρήστη	69
3.4.3 Πολιτικές Δικτύων	70
3.4.4 Γνωστικές Πτυχές	71
3.4.5 Τεχνική Προσέγγιση	71
3.5 ΠΑΡΑΚΟΛΟΥΘΗΣΗ ΚΑΙ ΑΝΙΧΝΕΥΣΗ	73
3.5.1 Διαδικασία Παρακολούθησης	73
3.5.2 Διαδικασία Ανίχνευσης	73
3.5.3 Γνωστικές Πτυχές	74

3.5.4 Τεχνική Προσέγγιση	75
3.6 ΔΙΑΠΡΑΓΜΑΤΕΥΣΗ ΚΑΙ ΕΠΙΛΟΓΗ	76
3.6.1 Γνωστικές Πτυχές	77
3.6.2 Τεχνική Προσέγγιση	77
3.7 ΑΛΛΗΛΕΠΙΔΡΑΣΗ ΤΩΝ ΣΥΝΙΣΤΩΣΩΝ	78
4. ΕΥΦΥΕΣ ΕΠΑΝΑΔΙΑΜΟΡΦΩΣΙΜΟ ΣΥΣΤΗΜΑ ΔΙΑΧΕΙΡΙΣΗΣ ΤΕΡΜΑΤΙΚΩΝ	79
4.1 ΕΙΣΑΓΩΓΗ	79
4.2 ΔΙΕΓΕΡΣΗ ΤΟΥ ΣΥΣΤΗΜΑΤΟΣ	79
4.3 ΠΑΡΑΚΟΛΟΥΘΗΣΗ	80
4.4 ΑΝΙΧΝΕΥΣΗ	80
4.5 ΔΙΑΠΡΑΓΜΑΤΕΥΣΗ	81
4.5.1 Έναρξη Επαναδιαμορφωτικών Διαδικασιών από το Τερματικό	81
4.5.2 Έναρξη Επαναδιαμορφωτικών Διαδικασιών από το Δίκτυο	82
4.6 ΕΠΙΛΟΓΗ	82
4.6.1 Βελτιστοποίηση Αντικειμενικής Συνάρτησης Επιλογής	83
4.6.1.1 Αντικειμενική Συνάρτηση Επιλογής	84
4.6.1.2 Παράγοντας Ποιότητας	84
4.6.1.3 Παράγοντας Κόστους	85
4.6.1.4 Έξοδος	86
4.7 ΕΦΑΡΜΟΓΗ ΕΠΑΝΑΔΙΑΜΟΡΦΩΤΙΚΩΝ ΔΙΑΔΙΚΑΣΙΩΝ	86
4.8 ΕΛΕΓΧΟΣ ΕΠΑΝΑΔΙΑΜΟΡΦΩΣΗΣ	86
4.9 ΔΙΑΧΕΙΡΙΣΗ ΤΕΡΜΑΤΙΚΩΝ (ΣΤΗΝ ΠΡΑΞΗ)	87
4.9.1 Απόκριση του Συστήματος Διαχείρισης Τερματικών σε Αλλαγή Χρησιμοποιούμενων Εφαρμογών και Κινητικότητας	88
4.9.1.1 Πλαίσιο Εφαρμογής	88
4.9.1.2 Αποτελέσματα	89
5. ΛΟΓΙΣΜΙΚΟ ΛΗΨΗΣ ΠΑΡΑΜΕΤΡΩΝ ΑΣΥΡΜΑΤΗΣ ΚΑΡΤΑΣ ΔΙΚΤΥΟΥ	93

5.1 ΕΙΣΑΓΩΓΗ	93
5.2 IWTOOLS	95
5.2.1 IWConfig	96
5.3 D – BUS	102
5.4 ΜΑΕΜΟ	105
5.5 ΜΕΘΟΔΟΛΟΓΙΑ	106
5.5.1 Στόχος	106
5.5.2 Εργαλεία, τεχνολογίες	106
5.5.3 Υλοποίηση	107
5.5.3.1 Λήψη Μετρήσεων Ασύρματης Κάρτας	107
5.5.3.2 Γραφική Διεπαφή Χρήστη	109
5.5.4 Αποτελέσματα	111
5.5.4.1 Λήψη Παραμέτρων Ασύρματης Κάρτας	111
5.5.4.2 Εμφάνιση των Παραμέτρων στη Γραφική Διεπαφή Χρήστη	113
ΒΙΒΛΙΟΓΡΑΦΙΑ	129
ΠΑΡΑΡΤΗΜΑ Ι	132
ΠΑΡΑΡΤΗΜΑ ΙΙ	169

ΛΙΣΤΑ ΠΙΝΑΚΩΝ

Πίνακας 1. Χαρακτηριστικά Κυψελωτών Συστημάτων 1ης Γενιάς	1
	5
Πίνακας 2. Διαθέσιμα και Μέγιστα Επίπεδα QoS (του σεναρίου)	8
	8
Πίνακας 3. Διαθέσιμα Attachment Points	8
	9

ΛΙΣΤΑ ΕΙΚΟΝΩΝ

Εικόνα 1. Αρχιτεκτονική του UMTS	22
Εικόνα 2. Inter - System Handover	31
Εικόνα 3. Αρχιτεκτονική του ABC Concept	39
Εικόνα 4. Οντότητες Διαχείρισης σε B3G Δίκτυα	43
Εικόνα 5. Αρχιτεκτονική για Δυναμική Κατανομή Φάσματος	52
Εικόνα 6. Time και Movement Based Schemes Διαχείρισης Θέσης	57
Εικόνα 7. Αρχιτεκτονική Συστήματος Διαχείρισης Τερματικών	72
Εικόνα 8. Διαδικασία Επιλογής Reconfigurations	83
Εικόνα 9. Λειτουργία Ευφυούς Επαναδιαμορφώσιμου Συστήματος Διαχείρισης Τερματικών	87
Εικόνα 10. Λήψη Παραμέτρων Ασύρματης Κάρτας Δικτύου	112
Εικόνα 11. Έξοδος του iwconfig όταν λείπουν τιμές από τις παραμέτρους	113
Εικόνα 12. Αρχικό Παράθυρο του GUI	114
Εικόνα 13. Παράθυρο "OVERVIEW"	115
Εικόνα 14. Παράθυρο "MEASUREMENTS"	116
Εικόνα 15. Παράθυρο "LOSSES"	117
Εικόνα 16. Παράθυρο "THRESHOLDS"	118
Εικόνα 17. Παράθυρο "Info"	119
Εικόνα 18. Επιλογή Παραμέτρου για Θέαση Σχετικής Πληροφορίας	120
Εικόνα 19. Εμφάνιση Τρέχουσας Τιμής Παραμέτρου και Σχετικής Περιγραφής της	121
Εικόνα 20. Menu Πλοήγησης Στο Σύστημα	122

Στην καλύτερή μου φίλη, Κατερίνα..

ΕΙΣΑΓΩΓΗ

1.1 ΕΙΣΑΓΩΓΗ

Από τη δεκαετία του '30 έως και σήμερα, η εξέλιξη της τεχνολογίας στον τομέα των επικοινωνιών πραγματοποιήθηκε με ραγδαίους ρυθμούς και αλληπάλληλες καινοτομίες. Ξεκινώντας από αναλογικά, ευμεγέθη και δύσχρηστα τερματικά με αυτοματοποιημένες λειτουργίες και ελάχιστη υποστήριξη υπηρεσιών φτάσαμε σε κινητούς εξοπλισμούς χειρός όχι κατά πολύ κατώτερους από έναν ισχυρό υπολογιστή γραφείου, με πληθώρα λειτουργιών, ικανοποιητικά επίπεδα ποιότητας και ποικίλλες εφαρμογές.

Σήμερα, διανύοντας την Τρίτη Γενιά της τεχνολογίας, η επιστημονική κοινότητα κοιτάει ένα βήμα μπροστά και αναπτύσσει την Επόμενη Γενιά. Ο στόχος είναι η διατήρηση της υπάρχουσας ραδιοϋποδομής και η σύνθεσή της σε μία ενιαία, με από άκρο σε άκρο υποστήριξη της Reconfigurability έννοιας. Κρατώντας την υποδομή, λοιπόν, οι έρευνες πλέον στρέφονται στον εξοπλισμό του χρήστη. Το όραμα είναι η δημιουργία ολοκληρωμένων, ευφυών, πλήρως προσαρμόσιμων τερματικών τα οποία θα μπορούν να εκμεταλλευτούν πλήρως τις ποικίλλες δυνατότητες που προσφέρει η σύνθετη ραδιοϋποδομή των B3G δικτύων και θα προσφέρουν σε οποιαδήποτε ώρα και μέρος τα καλύτερα επίπεδα ποιότητας υπηρεσιών στο χρήστη ώστε ο τελευταίος να είναι «Always Best Connected».

Προς την κατεύθυνση αυτή, στοχεύει και η προκείμενη εργασία η οποία ερευνά την B3G τεχνολογία εστιάζοντας στα reconfigurable τερματικά και υλοποιεί μία από τις λειτουργίες τους. Αρχικά, στην εργασία θα δούμε μία ιστορική ανασκόπηση δίνοντας το απαραίτητο θεωρητικό υπόβαθρο των προηγούμενων τεχνολογιών. Προχωρώντας, αναλύεται η σύνθετη ραδιοϋποδομή των Πέραν της Τρίτης Γενιάς δικτύων και παρουσιάζονται οι καινοτομίες που αυτή εισάγει. Στο Τρίτο Κεφάλαιο, παρουσιάζεται αναλυτικά, σε θεωρητικό πλαίσιο, ένα σύστημα διαχείρισης ευφυών τερματικών της B3G, οι απαιτήσεις και οι λειτουργίες του. Συνεχίζοντας στο Τέταρτο, θα δούμε τη διαχείριση τερματικών από τεχνικής και λειτουργικής σκοπιάς, αναλύοντας σε κάθε φάση της λειτουργίας του, τις διεργασίες και τους αλγορίθμους που αυτό υλοποιεί και εκτελεί.

Το τελευταίο, Πέμπτο κεφάλαιο, πραγματεύεται την σχεδίαση και υλοποίηση ενός λογισμικού, στόχος του οποίου είναι η λήψη ασύρματων

παραμέτρων και μετρήσεων και η εμφάνιση αυτών σε ένα γραφικό περιβάλλον χρήστη. Αφού δοθεί το απαραίτητο θεωρητικό υπόβαθρο και εξηγηθούν οι λειτουργίες που υλοποιεί το υπό συζήτηση λογισμικό, παρουσιάζονται αναλυτικά οι διεργασίες τις οποίες αυτό εκτελεί.

1.2 ΜΗΔΕΝΙΚΗ ΓΕΝΙΑ ΤΕΧΝΟΛΟΓΙΑΣ (0G)

Αρχικά, τα πρώτα συστήματα επικοινωνιών χρησιμοποιήθηκαν από αστυνομικά τμήματα σε πόλεις των Η.Π.Α. για λόγους δημόσιας ασφάλειας. Το 1934 είχαν εγκατασταθεί, σε 252 αστυνομικά τμήματα, τα πρώτα συστήματα κινητών επικοινωνιών, που χρησιμοποιούσαν αναλογική διαμόρφωση πλάτους (AM), ενώ 5.000 συσκευές εγκαταστάθηκαν σε αυτοκίνητα της αστυνομίας για την επικοινωνία τους με τα τμήματα. Με την εισαγωγή της διαμόρφωσης συχνότητας, από τον Edwin Armstrong, το 1935, όλα τα συστήματα κινητών επικοινωνιών υιοθέτησαν την FM διαμόρφωση.

Το 1946, εγκαταστάθηκαν για πρώτη φορά συστήματα κινητών επικοινωνιών σε 25 πόλεις των Η.Π.Α. Κάθε σύστημα χρησιμοποιούσε έναν πομπό σε υψηλό πύργο για να καλύπτει αποστάσεις μέχρι 50 χιλιομέτρων. Ένα χρόνο αργότερα, το Δεκέμβριο του 1947, οι μηχανικοί της Bell Labs Douglas H. Ring και W. Rae Young, πρότειναν τη χρήση εξαγωνικών κυψελών για τις κινητές επικοινωνίες. Παράλληλα, προτάθηκε από τον Philip T. Porter, επίσης μηχανικό της Bell Labs, η τοποθέτηση των σταθμών βάσης στις γωνίες των εξαγώνων αντί του κέντρου τους, καθώς και η χρήση κατευθυντικών κεραιών που θα λάμβαναν από τρεις διαφορετικές τοποθεσίες και θα εξέπεμπαν σε τρεις γειτονικές κυψέλες. Την περίοδο εκείνη οι τεχνολογίες κυψελωτών δικτύων ήταν ουσιαστικά ανύπαρκτες, μέχρι τη δεκαετία του '60 όταν ο Richard H. Frenkiel και ο Joel S. Engel της Bell Labs ανέπτυξαν την απαιτούμενη ηλεκτρονική υποδομή.

Στην Ευρώπη, η ραδιοτηλεφωνία χρησιμοποιήθηκε αρχικά μεταξύ του Βερολίνου και του Αμβούργου στα πρώτης τάξεως επιβατικά τρένα. Την ίδια περίοδο, εισήχθη και στα επιβατικά αεροπλάνα ως μέτρο ασφάλειας της εναέριας κυκλοφορίας, ενώ αργότερα έγινε εκτεταμένη χρήση της στα γερμανικά άρματα μάχης κατά τη διάρκεια του Δεύτερου Παγκοσμίου Πολέμου. Μετά τον πόλεμο, η Γερμανική αστυνομία, στη ελεγχόμενη από τους Βρετανούς ζώνη, αξιοποίησε το πλέον μη χρησιμοποιούμενο τηλεφωνικό εξοπλισμό των αρμάτων για να δημιουργήσει τα πρώτα ραδιο-περιπολικά αυτοκίνητα. Αξίζει να τονιστεί ότι σε όλες τις αναφερθείσες περιπτώσεις, η υπηρεσία της ραδιοτηλεφωνίας και η διαχείριση των τερματικών ήταν προσιτή μόνο σε ειδικούς που εκπαιδεύονταν για τη χρήση των εν λόγω συστημάτων. Η πρώτη επαφή των απλών πολιτών με τις κινητές επικοινωνίες έγινε στις αρχές της δεκαετίας του '50, όταν τα πλοία του Ρήνου έδωσαν τη δυνατότητα σε ανειδίκευτους πελάτες να χρησιμοποιήσουν την τεχνολογία αυτή υπό το ρόλο των τελικών χρηστών.

Το 1956 αναπτύχθηκε από την Ericsson και κυκλοφόρησε στη Σουηδία το πρώτο πλήρως αυτοματοποιημένο κινητό τηλεφωνικό σύστημα, υπό την ονομασία MTA (Mobile Telephone system A). Ήταν το πρώτο σύστημα για το οποίο δεν απαιτούνταν οποιοδήποτε είδος χειρωνακτικού ελέγχου, αλλά είχε το σοβαρό μειονέκτημα του μεγάλου βάρους των κινητών τερματικών τα

οποία ζύγιζαν 40 κιλά. Το MTB, που αποτέλεσε την αναβαθμισμένη έκδοση του MTA, χρησιμοποιούσε transistors, DTMF (Dual – tone multi – frequency) σηματοδότηση και ζύγιζε μόλις 9 κιλά. Αρχικά εξυπηρετούσε 150 πελάτες ενώ μέχρι το 1983 (έτος διακοπής του) ο αριθμός συνδρομητών είχε φτάσει τους 600.

Το 1967 κάθε κινητό τηλέφωνο έπρεπε να παραμένει μέσα στην περιοχή κάλυψης των κυψελών, εξυπηρετούμενο από μόνο ένα σταθμό βάσης κατά τη διάρκεια μιας τηλεφωνικής κλήσης, με αποτέλεσμα να μην παρέχεται η έννοια της συνέχειας των τηλεφωνικών υπηρεσιών στα τερματικά που περιάγονταν σε διαφορετικές κυψέλες. Έτσι, το 1970 ο Amos E. Joel, Jr. , ακόμα ένας μηχανικός της Bell Labs, εφήυρε ένα αυτόματο σύστημα μεταφοράς που μετέφερε μια τρέχουσα κλήση από ένα κανάλι συνδεδεμένο στο δίκτυο κορμού σε άλλο, επιτρέποντας έτσι στα κινητά τερματικά να κινούνται ανάμεσα στις διάφορες περιοχές των κυψελών χωρίς τον κίνδυνο της διακοπής επικοινωνίας.

Ένα από τα πρώτα πραγματικά επιτυχή εμπορικά κινητά τηλεφωνικά δίκτυα ήταν το ARP στη Φινλανδία που προωθήθηκε το 1971. Το ARP μπορεί να θεωρηθεί ως κυψελωτό δίκτυο μηδενικής γενιάς (0G, zero generation), το οποίο είναι λίγο πιο ψηλά σε ιεραρχία από τα προηγούμενα, περιορισμένης κάλυψης, δίκτυα. Την ίδια χρονιά, η AT&T υπέβαλε μια πρόταση για κυψελωτή υπηρεσία στην FCC (Federal Communications Commission) η οποία έγινε δεκτή τελικά το 1982 με την ονομασία «AMPS» (Advanced Mobile Phone Service) ενώ παράλληλα της αποδόθηκαν συχνότητες στην περιοχή των 824-894 MHz. Το 1990 το αναλογικό AMPS εκτοπίστηκε από το ψηφιακό D-AMPS (Digital AMPS).

1.3 ΠΡΩΤΗ ΓΕΝΙΑ ΤΕΧΝΟΛΟΓΙΑΣ (1G)

Τα Πρώτης Γενιάς (1G) δίκτυα κινητών επικοινωνιών ήταν από τα πρώτα αναλογικά κυψελωτά συστήματα που στηρίχθηκαν σε δίκτυα καταναμημένων πομποδεκτών για την επικοινωνία κινητών τερματικών, τα οποία ήταν επίσης αναλογικά και χρησιμοποιούνταν μόνο για μετάδοση φωνής με Διαμόρφωση Συχνότητας (FM). Στα συστήματα αυτά, συνήθως, αποδιδόταν ζώνη συχνοτήτων στα 25 MHz για τη μετάδοση σημάτων από το σταθμό βάσης στο τερματικό (downlink) και μια διαφορετική ζώνη επίσης στα 25 MHz για την επικοινωνία του τερματικού με το σταθμό (uplink). Οι ζώνες αυτές, τελικά, χωρίστηκαν σε κανάλια επικοινωνίας, το κάθε ένα από τα οποία χρησιμοποιούνταν από έναν χρήστη. Στην περίπτωση του AMPS, το κάθε κανάλι απήχε από τα γειτονικά του κατά διάστημα 30KHz, πράγμα μη αποδοτικό από την άποψη του διαθέσιμου φάσματος, με αποτέλεσμα ο αριθμός των κλήσεων που μπορούσαν να πραγματοποιηθούν να είναι πολύ περιορισμένος. Εντούτοις, το σύστημα ήταν FDMA (Frequency Division Multiple Access), οπότε ένας δεύτερος χρήστης μπορούσε να χρησιμοποιήσει το ίδιο κανάλι, μόλις ο πρώτος τερμάτιζε την επικοινωνία.

Στην Πρώτη Γενιά τεχνολογίας εισήχθη και η έννοια των Συστημάτων Επαναχρησιμοποίησης Συχνότητας. Λόγω της περιορισμένης ισχύος εξόδου των πομπών, οι οποίοι σχεδιάστηκαν έτσι ώστε να καλύπτουν συγκεκριμένες περιοχές, έγινε εφικτή η επαναχρησιμοποίηση των ίδιων συχνοτήτων σε άλλες κυψέλες οι οποίες απήχαν αρκετά έτσι ώστε να μη δημιουργούνται παρεμβολές. Χάρη στα συστήματα αυτά αυξήθηκαν οι δυνατότητες των δικτύων, ανάμεσα στις οποίες είναι και η συνέχιση μιας τηλεφωνικής κλήσης ενός τερματικού το οποίο μεταβαίνει από μία κυψέλη σε μία άλλη (μεταπομπή).

Το πρώτο παγκοσμίως κυψελωτό σύστημα αναπτύχθηκε και λειτούργησε στην Ιαπωνία το 1979, από τη Nippon Telephone and Telegraph (NTT). Παράλληλα, στην Ευρώπη το πρώτο κυψελωτό τηλεφωνικό δίκτυο ξεκίνησε να λειτουργεί το 1981 υπό την ονομασία NMT450 (Nordic Mobile Telephone System), που χρησιμοποιούσε μπάντα συχνοτήτων στα 450 MHz. Αργότερα, το 1985, ακολούθησε και το Ηνωμένο Βασίλειο, στο οποίο λειτούργησε το TACS (Total Access Communications System). Στον πίνακα που ακολουθεί, απεικονίζονται συνοπτικά τα βασικά χαρακτηριστικά των κυψελωτών συστημάτων 1^{ης} Γενιάς για τις πιο σημαντικές προδιαγραφές:

Προδιαγραφή	Ζώνη Συχνοτήτων Ευθεία/ Αντίστροφη Ζεύξη (MHz)	Απόσταση Φερόντων (KHz)	Αριθμός Διαύλων	Περιοχή Ανάπτυξης
NTT	925 - 940/ 870 - 885	25	600	Ιαπωνία
NMT450	453 - 457,5/ 463 - 467,5	25	180	Σουηδία
NMT900	890 - 915/ 935 - 960	12.5	1999	Σουηδία
AMPS	824 - 849/ 869 - 894	30	832	ΗΠΑ
ETACS	872 - 905/ 917 - 950	25	1240	Αγγλία
JTACS	915 - 925/ 860 - 870	25	400	Ιαπωνία
C-450	450 - 455,74/ 460 - 465,74	10	573	Γερμανία

Πίνακας 1. Χαρακτηριστικά Κυψελωτών Συστημάτων 1ης Γενιάς

Με την εμφάνιση των δικτύων Δεύτερης Γενιάς (2G), τα 1G τερματικά έγιναν ξεπερασμένα, αφ' ενός επειδή δεν μπορούσαν να προσαρμοστούν στα πρότυπα της Δεύτερης γενιάς, και αφ' ετέρου επειδή παρείχαν πολύ μικρά επίπεδα ασφάλειας. Το γεγονός ότι ο οποιοσδήποτε δέκτης που συντονιζόταν στην σωστή συχνότητα μπορούσε να κρυφακούσει την εκάστοτε συνομιλία, λόγω έλλειψης κρυπτογράφησης, ήταν ένας από τους λόγους της μετάβασης της τεχνολογίας στη Δεύτερη Γενιά κυψελωτών συστημάτων.

1.4 ΔΕΥΤΕΡΗ ΓΕΝΙΑ ΤΕΧΝΟΛΟΓΙΑΣ (2G)

1.4.1 Μετάβαση από τα αναλογικά στα ψηφιακά συστήματα

Στις αρχές του 1980, είχε γίνει εμφανής η αδυναμία των αναλογικών κυπελωτών συστημάτων να ανταπεξέλθουν στις απαιτήσεις των επερχόμενων χρόνων, λόγω: του εξαιρετικά περιορισμένου φάσματος προς απόδοση που σήμαινε χαμηλή χωρητικότητα συστημάτων, της αντίληψης των χρηστών ότι ήταν περιορισμένης χρησιμότητας λόγω της χαμηλής ποιότητας υπηρεσιών, της αδυναμίας μείωσης του κόστους των τερματικών και της υποδομής των δικτύων, καθώς και της ασυμβατότητας μεταξύ των διάφορων αναλογικών συστημάτων.

Έτσι, ο καθορισμός του μέγιστου ρυθμού σηματοδοσίας που μπορεί να μεταδοθεί από τηλεγραφικό δίαυλο συγκεκριμένου εύρους ζώνης, από τον Nyquist (1924), και του μέγιστου πλήθους δεδομένων που μπορούν να μεταδοθούν αξιόπιστα, από τον Hartley (1928), καθώς και η ανάλυση διάφορων ψηφιακών συστημάτων με βάση τη γεωμετρική τους προσέγγιση, από τον Kotelnikov (1947) και οι τεχνικές αναγνώρισης και διόρθωσης σφαλμάτων από τον Hamming (1950), έθεσαν τα θεμέλια των σημερινών ψηφιακών επικοινωνιών.

Ενώ οι ψηφιακές τεχνικές γρήγορα υιοθετήθηκαν στα ενσύρματα συστήματα επικοινωνιών, έπρεπε να προηγηθεί η ραγδαία εξέλιξη στην τεχνολογία της μικροηλεκτρονικής, μέχρι να βρουν εφαρμογή στα ασύρματα συστήματα. Τα πλεονεκτήματα των ψηφιακών συστημάτων έναντι των αναλογικών είναι αρκετά και περιλαμβάνουν:

- Την αυξημένη ανοσία στο θόρυβο
- Τις περισσότερες αποδοτικές τεχνικές μετάδοσης και την καλύτερη ποιότητα υπηρεσιών, κάνοντας χρήση κωδίκων διόρθωσης σφαλμάτων, φασματικά αποδοτικών τεχνικών διαμόρφωσης, αποδοτική κωδικοποίηση πληροφορίας και κωδικοποίηση διαύλου
- Τη δυνατότητα εφαρμογής τεχνικών κρυπτογράφησης για την ασφάλεια της μετάδοσης
- Την ευλυγισία στην ανάπτυξη και επέκταση των δικτύων
- Τη χαμηλότερη κατανάλωση ισχύος
- Επιτρέπουν την εφαρμογή διαφορετικών επιπέδων ποιότητας υπηρεσίας, παρέχοντας ταυτόχρονα υπηρεσίες φωνής και δεδομένων
- Δίνουν τη δυνατότητα επεξεργασίας του σήματος
- Παρέχουν τη δυνατότητα για επιπλέον τεχνικές πολλαπλής πρόσβασης (TDMA / CDMA / SDMA) και duplexing (TDD)
- Επιτρέπουν την υλοποίηση software δεκτών

Η στροφή της βιομηχανίας προς τις ψηφιακές τεχνικές, οι οποίες εκμεταλλεύόμενες τα εμφανή πλεονεκτήματα, αποδείχτηκαν ιδιαίτερα αποτελεσματικές τόσο στα ασύρματα όσο και στα ενσύρματα δίκτυα, ήταν αναμενόμενη. Στις παρακάτω υποενότητες θα δούμε τις πιο σημαντικές από τις τεχνολογίες της 2^{ης} γενιάς, δίνοντας μία σύντομη περιγραφή των χαρακτηριστικών και των καινοτομιών που εισήγαγαν.

1.4.2 **GSM**

Το GSM (Global System for Mobile communications) είναι το ευρωπαϊκό πρότυπο κινητής τηλεφωνίας και το πλέον πετυχημένο κυψελωτό σύστημα παγκοσμίως (Α. Κανάτας et al, 2008), εφαρμοσμένο σε πάνω από 80 χώρες πλέον, με ραγδαία αυξανόμενο πλήθος συνδρομητών. Είναι ένα ψηφιακό, circuit switched σύστημα που λειτουργεί εμπορικά από το 1991. Οι βασικοί στόχοι του GSM συνίστανται:

- Στην παροχή δυνατότητας περιαγωγής (roaming) οπουδήποτε στην Ευρώπη,
- Στην εγγύηση ποιότητας υπηρεσίας τουλάχιστον εφάμιλλης αυτής των συστημάτων πρώτης γενιάς,
- Στη μέγιστη επαναχρησιμοποίηση του φάσματος, με δυνατότητες κλιμάκωσης του συστήματος,
- Στην κρυπτογράφηση της μεταδιδόμενης πληροφορίας του χρήστη,
- Στην ευελιξία κλιμάκωσης των ρυθμών μετάδοσης, άρα και των παρεχόμενων υπηρεσιών και
- Στη δυνατότητα εφαρμογής ευέλικτων συστημάτων χρέωσης.

Σήμερα το GSM είναι ένα συνεχώς εξελισσόμενο πρότυπο μέσα στο Ευρωπαϊκό Ινστιτούτο Τυποποίησης Τηλεπικοινωνιών (European Telecommunications Standards Institute, ETSI) επιδεικνύοντας μια διαρκή πορεία εξέλιξης προς υπηρεσίες πολυμέσων 3^{ης} γενιάς. Από τον Ιούνιο 2000, οι εργασίες εξέλιξης της τεχνολογίας στη ραδιοεπαφή έχουν μεταφερθεί στο 3GPP (3rd Generation Partnership Project).

1.4.3 **HSCSD**

Το HSCSD (High-Speed Circuit-Switched Data) προσφέρει μεγαλύτερες ταχύτητες από το GSM, έως 57.6 Kbps, αλλά είναι μεταγωγής κυκλώματος. Επομένως, είναι εξ' ορισμού ανεπαρκές σε συνθήκες ραγδαίας αύξησης κινητικότητας στο δίκτυο, λαμβάνοντας υπ' όψιν ότι χρησιμοποιεί μέχρι τέσσερα ραδιοκανάλια. Εξ' αιτίας αυτών των αδυναμιών, μόλις 30 σταθμοί χρησιμοποίησαν το HSCSD, ενώ οι περισσότεροι στράφηκαν στο GPRS, το οποίο θα δούμε παρακάτω.

1.4.4 **cdmaONE**

Η τεχνολογία διεύρυνσης φάσματος χρησιμοποιούνταν από στρατιωτικές εφαρμογές για πολύ καιρό. Στα μέσα της δεκαετίας του '80, ο στρατός των Η.Π.Α. κατακερμάτισε την τεχνολογία αυτή και εξέτασε τη χρήση της στις κυψελωτές εφαρμογές τηλεφωνίας. Έτσι, τον Ιούλιο του 1993, εγκρίθηκε το βασισμένο σε διεύρυνση φάσματος CDMA (Code Division Multiple Access) πρότυπο από την TIA (Telecommunications Industry Association).

Τα εμπορικά CDMA δίκτυα άνοιξαν το 1995 προσελκύοντας μόλις 9 εκατομμύρια χρήστες ως το 1998. Έκτοτε, τα πράγματα παρουσίασαν βελτίωση, και έτσι πλέον εξυπηρετούνται 100 εκατομμύρια χρήστες κυρίως στην Αμερική (55 εκατ.) και την Ασία (40 εκατ.). Σήμερα, το CDMA ονομάζεται cdmaOne, για να διαφοροποιείται από τα CDMA συστήματα 3^{ης} γενιάς.

1.4.5 **D – AMPS**

Με τα αναλογικά κυψελωτά συστήματα, όπως τα AMPS, ένας συνδρομητής τη φορά χρησιμοποιεί ένα κανάλι 30 KHz. Το D-AMPS (TDMA σύστημα που σχεδιάστηκε για να συνυπάρξει με τα συστήματα AMPS) διαιρεί αυτά τα κανάλια 30 KHz σε 3 κανάλια, δίνοντας τη δυνατότητα σε 3 χρήστες να χρησιμοποιούν ένα ραδιοκανάλι, αποδίδοντας μοναδικές χρονοσχισμές στον καθέναν.

1.4.6 **PDC**

Το PDC (Personal Digital Cellular) αναπτύχθηκε στην Ιαπωνία από το 1989 και στηρίζεται στις αρχές του IS-54, με TDMA τεχνική πολλαπλής πρόσβασης (3 χρονοσχισμές ανά φέρον). Αποτελεί το πιο πειστικό παράδειγμα Mobile Internet (iMode). Το iMode έχει ήδη κερδίσει πάνω από 30 εκατομμύρια συνδρομητές, χάρη σε μια μεγάλη προσφορά υπηρεσιών και ένα άριστο επιχειρηματικό μοντέλο (ογκοχρέωση, διακανονισμοί διαμοιρασμού κερδών με τους ιδιοκτήτες περιεχομένου και άλλα).

1.5 **2.5 ΓΕΝΙΑ ΤΕΧΝΟΛΟΓΙΑΣ (2.5G)**

1.5.1 **GPRS**

Το GSM προσφέρει παγκόσμια διαθεσιμότητα και σχεδόν απεριόριστη κινητικότητα τερματικού κυρίως για τις υπηρεσίες φωνής. Ωστόσο, εξαιτίας της περιορισμένης ποιότητας υπηρεσιών που παρέχει, όσον αφορά στη μετάδοση δεδομένων, κρίθηκε απαραίτητη η εισαγωγή ενός νέου συστήματος που θα υποστηρίζει μεταφορά δεδομένων. Οι τωρινές υπηρεσίες δεδομένων του GSM, των 9.6 Kbps, επιφέρουν μια ιδιαίτερα υψηλή πολυπλοκότητα στην ασύρματη διεπαφή και στις διαδικασίες σηματοδότησης του δικτύου. Κατά συνέπεια, είναι προφανές ότι η ανάπτυξη ενός συστήματος ικανού να παρέχει ταχύτερη μετάδοση δεδομένων και εξασφαλισμένη υψηλή ποιότητα και αξιοπιστία στη μεταφορά των δεδομένων, αποτελούσε επιτακτική ανάγκη για το παγκόσμιο τηλεπικοινωνιακό στερέωμα.

Στην κατεύθυνση αυτή στοχεύει το Σύστημα Γενικής Ασύρματης Υπηρεσίας Μεταγωγής Πακέτου (General Packet Radio Service, GPRS) το οποίο παρέχει αποτελεσματική χρησιμοποίηση των ασύρματων πόρων για υπηρεσίες μεταγωγής πακέτου που χαρακτηρίζονται από ασυνεχή ρυθμό μετάδοσης bit, ενώ οι ρυθμοί μετάδοσης που δύναται να προσεγγίσει το GPRS, θεωρητικά, μπορούν να φτάσουν μέχρι και τα 160 Kbps περίπου. Πρέπει να τονιστεί, στο σημείο αυτό, ότι το GPRS δεν αποτελεί ουσιαστικά ένα εξ' ολοκλήρου νέο σύστημα, αλλά μια προέκταση του συστήματος GSM στο δίκτυο κορμού (core network), για την υποστήριξη υπηρεσιών μεταγωγής πακέτου, και για αυτό έχει χαρακτηριστεί ως σύστημα 2, 5 G.

Αφού, λοιπόν, το GPRS είναι packet switched network, προφανές είναι ότι δεν απαιτείται η εγκατάσταση σύνδεσης από άκρο σε άκρο κατά την αποστολή δεδομένων. Ως αποτέλεσμα, υφίσταται μεγάλη οικονομία στους διαθέσιμους πόρους του δικτύου και επιτυγχάνονται υψηλοί ρυθμοί μετάδοσης δεδομένων (160 Kbps θεωρητικά), δίνοντας έτσι τη δυνατότητα υλοποίησης και εφαρμογής νέων υπηρεσιών οι οποίες προστίθενται στις ήδη υπάρχουσες από το GSM. Οι υπηρεσίες αυτές είναι γενικότερα προσανατολισμένες προς τις γνωστές υπηρεσίες του Διαδικτύου, και οι σημαντικότερες από αυτές είναι:

- Υπηρεσίες Διαδικτύου
- Υπηρεσίες Θέσης
- Υπηρεσία MMS (Multimedia Messaging Service)
- Υπηρεσία Προηγμένων MMS (Advanced MMS)

1.6 ΤΡΙΤΗ ΓΕΝΙΑ ΤΕΧΝΟΛΟΓΙΑΣ (3G)

1.6.1 Στροφή στα Συστήματα Τρίτης Γενιάς

Η ανάγκη για μεγαλύτερη χωρητικότητα φωνής, οι υψηλών ταχυτήτων υπηρεσίες δεδομένων (384 Kbps για κινητή και 2 Mbps για σταθερή τηλεφωνία), η δυνατότητα ασύρματης πρόσβασης στο διαδίκτυο και η παγκόσμια περιαγωγή εξώθησαν τα όρια των υπαρχόντων 2G και 2.5 G δικτύων. Δημιουργήθηκε, λοιπόν, η ανάγκη για νέες (3G) τεχνολογίες οι οποίες να μπορούν να υποστηρίξουν τις νέες, πιο απαιτητικές εφαρμογές και να προσφέρουν:

- Ποιότητα φωνής συγκρίσιμη με το Δημόσιο Τηλεφωνικό Δίκτυο Μεταγωγής (Public Switched Telephone Network – PSTN)
- Μεταβαλλόμενο ρυθμό μετάδοσης για δυνατότητα προσφοράς εύρους ζώνης κατά απαίτηση
- Πολυπλεξία υπηρεσιών, με διαφορετικές απαιτήσεις ως προς την ποιότητα (π.χ. φωνή, video, πακέτα δεδομένων), στην ίδια σύνδεση
- Συνύπαρξη 2^{ης} και 3^{ης} Γενιάς συστημάτων και υποστήριξη μεταπομπής μεταξύ τους, για βελτίωση της κάλυψης και εξισορρόπηση του τηλεπικοινωνιακού φορτίου
- Ρυθμό δεδομένων στα 144 Kbps για χρήστες που κινούνται σε ιδιωτικά ή δημόσια οχήματα, μέσα σε μεγάλες περιοχές
- Ρυθμό δεδομένων στα 384 Kbps για πεζούς που κινούνται σε μικρές περιοχές
- Υποστήριξη εφαρμογών μεταγωγής πακέτου και μεταγωγής κυκλώματος
- Μια προσαρμόσιμη ραδιοδιεπαφή που θα ταιριάζει στην υψηλά ασυμμετρική φύση των περισσότερων Διαδικτυακών επικοινωνιών – πολύ μεγαλύτερο εύρος ζώνης για το downlink από το uplink
- Πιο αποτελεσματική χρήση του διαθέσιμου φάσματος
- Συνύπαρξη FDD και TDD συστημάτων
- Υποστήριξη πολλών και διαφορετικών εξοπλισμών (τηλέφωνα, κινητά, PDAs και άλλα)
- Ευκαμψία στην εισαγωγή νέων υπηρεσιών και τεχνολογιών

Έτσι λοιπόν, την ίδια χρονιά που κυκλοφόρησε εμπορικά το GSM, το ETSI ξεκίνησε την προτυποποίηση του δικτύου κινητών επικοινωνιών της επόμενης γενιάς. Το νέο σύστημα ονομάστηκε Universal Mobile Telecommunications System (UMTS). Η ανάπτυξη των τεχνολογιών της 3^{ης} Γενιάς, βέβαια, δεν έγινε μόνο από το ETSI. Υπήρχαν και άλλες οργανώσεις και ερευνητικά προγράμματα που είχαν τον ίδιο σκοπό. Η Ευρωπαϊκή Επιτροπή (European Commission) χρηματοδότησε ερευνητικά προγράμματα όπως την Έρευνα στις Προηγμένες Τηλεπικοινωνιακές Τεχνολογίες της Ευρώπης (Research on Advanced Communication Technologies in Europe –

RACE I και II) και τις Προηγμένες Τηλεπικοινωνιακές Τεχνολογίες και Υπηρεσίες (Advanced Communication Technologies and Services – ACTS). Παράλληλα με την Ευρώπη, υπήρχαν άλλα πολυάριθμα προγράμματα Τρίτης Γενιάς στις Ηνωμένες Πολιτείες, στην Ιαπωνία και την Κορέα, ενώ αρκετές ιδιωτικές εταιρίες τηλεπικοινωνιών διεξήγαγαν επίσης τις δικές τους ερευνητικές δραστηριότητες. Επιπλέον, το 1996 δημιουργήθηκε το UMTS forum για να επιταχύνει τη διαδικασία προσδιορισμού των απαιτούμενων προτύπων.

Ένα πολύ σημαντικό βήμα έγινε το 1996 – 1997, όταν το ETSI και η ARIB (Association of Radio Industries and Businesses) επέλεξαν το W-CDMA ως την υποψήφια διεπαφή 3^{ης} γενιάς. Επιπλέον, ο μεγαλύτερος Ιαπωνικός τηλεπικοινωνιακός πάροχος, NTT DoCoMo, διένειμε ένα δοκιμαστικό W-CDMA σύστημα στους μεγαλύτερους κατασκευαστές τηλεπικοινωνιακών συστημάτων. Το γεγονός αυτό ανάγκασε τους υπόλοιπους κατασκευαστές να λάβουν μια στρατηγική απόφαση, το οποίο σήμαινε είτε την αύξηση των δικών τους ερευνών πάνω στο W-CDMA είτε την αποχή τους από την ιαπωνική αγορά συστημάτων 3^{ης} γενιάς.

Αργότερα, οι πιο σημαντικές εταιρίες τηλεπικοινωνιών ένωσαν τις δυνάμεις τους στο 3GPP πρόγραμμα, στόχος του οποίου ήταν η παραγωγή προδιαγραφών για ένα 3G σύστημα βασισμένο στη ραδιοδιεπαφή UTRA (Universal Terrestrial Radio Access) του ETSI και στο αναβαθμισμένο GSM/GPRS Mobile Application Part (MAP) δίκτυο κορμού. Προς το παρόν, τη μέγιστη ευθύνη για το έργο ανάπτυξης της 3G τεχνολογίας τη φέρει το 3GPP.

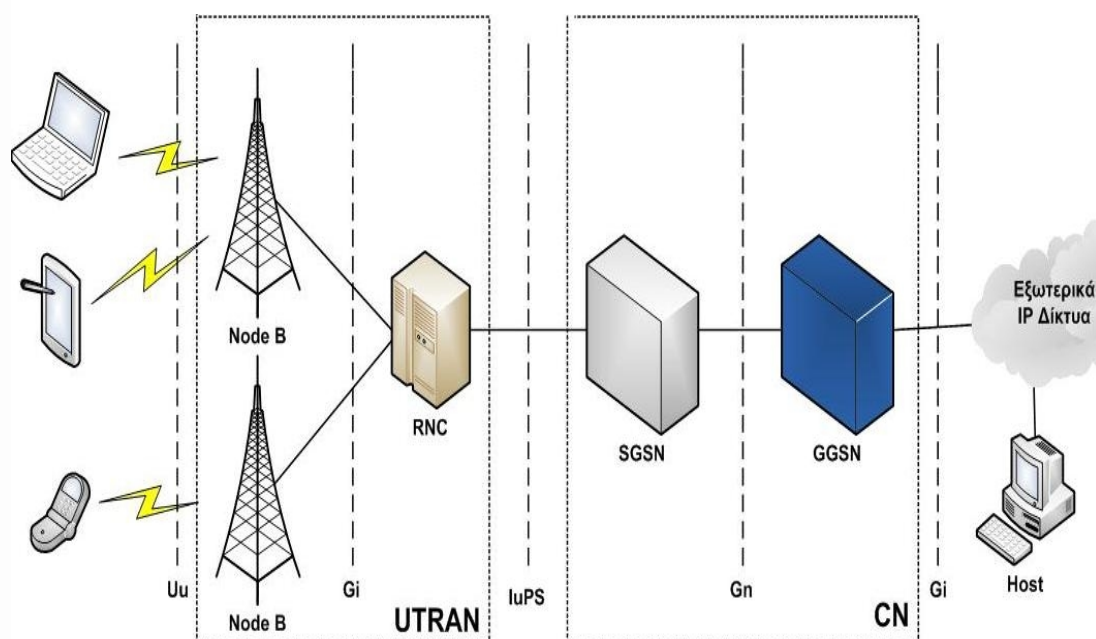
Παρακάτω θα δούμε την πιο αντιπροσωπευτική και σημαντική τεχνολογία 3^{ης} γενιάς, το UMTS, εστιάζοντας περισσότερο στον εξοπλισμό χρήστη και στις διαχειρίσεις στις οποίες αυτός υπόκειται.

1.6.2 UMTS

Η λογική ανάπτυξης των δικτύων 2,5G ήταν η όσο το δυνατόν μεγαλύτερη εκμετάλλευση της υποδομής της 2^{ης} γενιάς, διατηρώντας το υποδίκτυο ραδιοπρόσβασης για παροχή ραδιοκάλυψης και αναβαθμίζοντας το δίκτυο κορμού, ώστε να μπορεί να υποστηριχθεί η μεταγωγή πακέτων. Αντίθετα, κατά τη σχεδίαση των δικτύων 3^{ης} γενιάς, με δεδομένη την ανάπτυξη του δικτύου κορμού που υποστηρίζει μεταγωγή πακέτων και IP τεχνολογίες, οι αλλαγές έγιναν στο υποδίκτυο ραδιοπρόσβασης, τουλάχιστον κατά την πρώτη φάση ανάπτυξης των δικτύων. Η τεχνολογία της ραδιοεπαφής, συνεπώς, είναι ριζικά διαφορετική από τις προηγούμενες της, αφού επικράτησε η τεχνολογία διασποράς φάσματος και η αντίστοιχη CDMA τεχνική.

1.6.2.1 Αρχιτεκτονική του UMTS

Όπως φαίνεται και από την Εικόνα 1, το δίκτυο στο UMTS αποτελείται από τον εξοπλισμό του χρήστη (UE), το Δίκτυο Ραδιοπρόσβασης (UMTS Terrestrial Radio Access Network, UTRAN) που προσφέρει τη ραδιοεπαφή Uu στον Κινητό Σταθμό του χρήστη, και το Δίκτυο Κορμού (Core Network, CN) που είναι υπεύθυνο για τις συνδέσεις στο UMTS.



Εικόνα 1. Αρχιτεκτονική του UMTS

Το UTRAN (UMTS Terrestrial Radio Access Network) είναι ένα νέο δίκτυο ραδιοπρόσβασης που σχεδιάστηκε για το UMTS. Τα όριά του είναι από την πλευρά του UE, η Uu interface, δηλαδή η ραδιοεπαφή, ενώ από την πλευρά του CN είναι η Iu interface. Το Δίκτυο Ραδιοπρόσβασης του UMTS αποτελείται από πολλά υποδίκτυα RNS (Radio Network Subsystems), τα οποία περιέχουν έναν Ελεγκτή Σταθμών Βάσης (Radio Network Controller, RNC) και ένα ή περισσότερα Node B.

Το δίκτυο κορμού, από την άλλη, των συστημάτων 3^{ης} Γενιάς δε διαφέρει κατά πολύ από το αντίστοιχο αυτού των συστημάτων της 2^{ης} (GSM).

Συνδέεται με το UTRAN μέσω της διεπαφής Iu, ενώ αποτελείται από τους παρακάτω τέσσερις τομείς:

- Τομέας Μεταγωγής Κυκλώματος (Circuit Switched Domain)
- Τομέας Εγγραφής (Register Domain)
- Τομέας Υπηρεσιών (Service Domain)
- Τομέας Μεταγωγής Πακέτων (Packet Switched Domain)

1.6.2.2 Τερματικά

Τα τερματικά στο UMTS είναι και πολλαπλού τρόπου λειτουργίας (multi mode) και πολλαπλών ζωνών συχνοτήτων (multi band) (Α. Κανάτας et al, 2008). Τα ακόλουθα χαρακτηριστικά θεωρούνται απολύτως απαραίτητα και πρέπει να υποστηρίζονται:

- Διεπαφή κάρτας SIM
- Δυνατότητα εγγραφής και διαγραφής από το δίκτυο και τον ISP
- Ενημέρωση θέσης
- Εγκατάσταση connectionless και connection – oriented υπηρεσιών
- Διακριτή ταυτότητα εξοπλισμού
- Δυνατότητα αναγνώρισης των δυνατοτήτων του UE
- Πραγματοποίηση κλήσεων ανάγκης (π.χ. 166) ακόμα και χωρίς κάρτα SIM
- Εκτέλεση αλγορίθμων για έλεγχο πιστοποίησης και κρυπτογράφηση

Τα τερματικά πρέπει, επίσης, να υποστηρίζουν δυνατότητα μεταφόρτωσης πληροφοριών που αφορούν τις υπηρεσίες νέων πρωτοκόλλων, νέων διεπαφών των εφαρμογών (API), νέων δυνατοτήτων και λειτουργιών. Επιπλέον, πρέπει να υποστηρίζουν το Virtual Home Environment, αλλά και την ταυτόχρονη χρήση περισσότερων της μίας SIM κάρτας.

Υπάρχουν διάφοροι τύποι τερματικών στο UMTS:

- Απλού τρόπου λειτουργίας (single – mode) FDD
- Απλού τρόπου λειτουργίας TDD
- Διπλού τρόπου λειτουργίας (dual – mode) FDD / TDD
- Διπλού τρόπου λειτουργίας FDD / άλλο ραδιοσύστημα (π.χ. GSM)
- Διπλού τρόπου λειτουργίας TDD / άλλο ραδιοσύστημα
- Τριπλού τρόπου λειτουργίας (tri – mode) FDD / TDD / GSM

Αναφορικά με τις δυνατότητες του εξοπλισμού, διακρίνονται 4 τύποι τερματικών:

Τύπος 1: Είναι στην πράξη δύο τερματικά με ένα κέλυφος, που χρησιμοποιούν την ίδια διεπαφή ανθρώπου μηχανής (Man Machine Interface, MMI). Δυνατότητα ταυτόχρονης σύνδεσης στα δύο δίκτυα GSM/ GPRS και UMTS δεν υπάρχει.

Τύπος 2: Τα τερματικά του τύπου 2, όπως και αυτά του τύπου 1, δεν μπορούν να συνδεθούν ταυτόχρονα στα 2 αναφερθέντα δίκτυα. Μπορούν, όμως, να διεξάγουν μετρήσεις στο ένα δίκτυο, την ώρα που είναι συνδεδεμένα στο άλλο.

Τύπος 3: Το τερματικό μπορεί να διατηρεί πολλές συνδέσεις με διαφορετικά δίκτυα, αλλά μόνο μία σύνδεση είναι ενεργή κάθε φορά. Για παράδειγμα, αν υπάρχει μία ενεργή σύνδεση με τον UMTS, το τερματικό μπορεί ταυτόχρονα να λαμβάνει paging μηνύματα από ένα δίκτυο GSM. Ταυτόχρονη μεταφορά τηλεπικοινωνιακής κίνησης σε δύο δίκτυα δεν είναι δυνατή, αλλά μπορεί να γίνει σειριακά.

Τύπος 4: Υποστηρίζονται ταυτόχρονα πολλά δίκτυα, τόσο για τη σηματοδότηση όσο και για τηλεπικοινωνιακή κίνηση χρηστών.

1.6.2.3 Διαχείριση Ισχύος Τερματικών

Στο WCDMA όλοι οι χρήστες χρησιμοποιούν τον ίδιο συχνοτικό δίαυλο, με κάθε σύνδεση να αναγνωρίζεται από ορθογωνικούς κώδικες διασποράς. Συνεπώς, κάθε χρήστης φαίνεται σαν θόρυβος για τους υπόλοιπους χρήστες. Αν η ισχύς εκπομπής των χρηστών είναι ανεξέλεγκτη, τότε η προκύπτουσα παρεμβολή θα είναι αυξημένη. Συνεπώς, απαιτείται ένας Έλεγχος Ισχύος (Power Control) ο οποίος θα διαχειρίζεται την ισχύ των τερματικών ώστε να αποφεύγονται προβλήματα αυξημένων παρεμβολών και φαινομένων όπως το near – far. Υπάρχουν τρεις τρόποι ελέγχου ισχύος:

- Έλεγχος Ισχύος Ανοικτού Βρόχου (Open Loop Power Control)
- Έλεγχος Ισχύος Κλειστού Βρόχου (Closed Loop Power Control)
- Έλεγχος Ισχύος Εξωτερικού Βρόχου (Outer Loop Power Control)

1.6.2.3.1 Έλεγχος Ισχύος Ανοικτού Βρόχου

Στον Open Loop Power Control, το Node B χρησιμοποιώντας συγκεκριμένα κανάλια, μεταδίδει με broadcasting την πληροφορία για την ισχύ εκπομπής του UE, κατά την πρώτη του προσπάθεια πρόσβασης στο δίκτυο. Αφού, όπως είδαμε, κάθε χρήστης φαίνεται σαν θόρυβος στους υπόλοιπους, κατά την τυχαία πρόσβασή του στο δίκτυο (για την οποία θα μιλήσουμε πιο κάτω) η ισχύς εκπομπής του πρέπει να είναι η μικρότερη δυνατή. Συνεπώς, ο πομπός πρέπει να μετρά τα επίπεδα παρεμβολών και να προσαρμόζει την ισχύ του αντίστοιχα.

1.6.2.3.2 Έλεγχος Ισχύος Ανοιχτού Βρόχου

Στον Closed Loop Power Control, από τη στιγμή που εγκατασταθεί η σύνδεση, το Node B αναλαμβάνει τη διαχείριση της ισχύος, το οποίο έχοντας μετρήσεις από το λόγο σήματος προς παρεμβολή (SIR) και από τη λαμβανόμενη ισχύ στο Node B, ενημερώνει το UE να προσαρμόζει την ισχύ εκπομπής του κάθε 0,667 msec (χρόνος ίσος με τη διάρκεια μίας χρονοσχισμής). Εναλλακτικά, το Node B μπορεί να χρησιμοποιεί τις τιμές του ρυθμού εσφαλμένων πλαισίων FER (Frame Error Rate). Η διαδικασία που προκύπτει για τον έλεγχο της ισχύος, με αυτό τον τρόπο, είναι πιο αργή αλλά γίνεται με μεγαλύτερη αξιοπιστία, καθώς ο FER είναι πιο αξιόπιστος δείκτης από τον SIR.

1.6.2.3.3 Έλεγχος Ισχύος Εξωτερικού Βρόχου

Στον Outer Loop Power Control, το RNC προσαρμόζει την τιμή του επιθυμητού SIR (target SIR) στο Node B. Αυτή η ρύθμιση αφορά κάθε cell και γίνεται προκειμένου να διατηρηθεί σταθερή η ποιότητα μετάδοσης, με δεδομένη τιμή BER ή FER. Ο σταθερός έλεγχος του επιθυμητού SIR κάθε κυψέλης είναι πολύ σημαντικός για την απόδοση των CDMA networks. Αν δεν επιτυγχάνεται το target SIR, τότε μειώνεται το μέγεθος και, συνεπώς, η χωρητικότητα της κυψέλης. Αν, λοιπόν, το RNC εντοπίσει μία επιδεινούμενη σύνδεση, η επιθυμητή τιμή του SIR στο Node B θα αυξηθεί κατά ένα σταθερό ποσό. Το τερματικό, τότε, θα αυξήσει σταδιακά την ισχύ εκπομπής του, μέχρι να πετύχει τη νέα τιμή του SIR.

1.6.2.4 Τυχαία Πρόσβαση Τερματικών

Κάθε UE προκειμένου να συνδεθεί στο UMTS δίκτυο, εκτελεί μία διαδικασία τυχαίας πρόσβασης. Η τυχαία πρόσβαση του UE είναι δυνατή σε κανονικά διαστήματα που ονομάζονται Access Slots (AS). Κάθε 2 πλαίσια περιέχουν 15 τέτοια AS. Στην αρχή της τυχαίας πρόσβασης, το UE θα πρέπει πιθανώς να ανταγωνιστεί με άλλα UEs. Για το λόγο αυτό, το τερματικό μεταδίδει μία από τις 16 διαφορετικές διαθέσιμες επικεφαλίδες (preambles). Συνεπώς, κάθε AS παρέχει πρόσβαση χωρίς σύγκρουση σε 16 το πολύ UEs. Σε περίπτωση σύγκρουσης, το UE μεταδίδει δεύτερη επικεφαλίδα με μεγαλύτερη ισχύ. Αν η διαδικασία γίνει με επιτυχία, το τερματικό εκπέμπει το επιθυμητό μήνυμα, με διάρκεια 10 ή 20 msec.

Πριν την τυχαία πρόσβαση, το τερματικό επιλέγει τυχαία ένα υποκανάλι RACH το οποίο καθορίζει τα διαθέσιμα AS στο τρέχον πλαίσιο. Μετά την τυχαία επιλογή μίας υπογραφής, η τυχαία διασπορά της οποίας παράγει τις επικεφαλίδες, παράγεται η επικεφαλίδα η οποία και εκπέμπεται στο αποδιδόμενο AS με συγκεκριμένη ισχύ εκπομπής. Στη συνέχεια το UE περιμένει για επιβεβαίωση. Εάν δε τη λάβει, επιλέγεται μία νέα υπογραφή και η επικεφαλίδα εκπέμπεται σε άλλο AS με υψηλότερη ισχύ. Οι προσπάθειες συνεχίζονται με υψηλότερη ισχύ κάθε φορά. Μετά από έναν αριθμό ανεπιτυχών προσπαθειών, οι προσπάθειες σταματούν και ενημερώνονται τα ανώτερα στρώματα του δικτύου για την ανεπιτυχή πρόσβαση. Εάν δε, το τερματικό λάβει επιβεβαίωση επιτυχούς μετάδοσης, εκπέμπεται το μήνυμα με καθυστέρηση 3 ή 4 χρονοσχημάτων και ενημερώνονται τα ανώτερα στρώματα για την επιτυχή πρόσβαση.

1.6.2.5 Εντοπισμός Τερματικών

Μία από τις υπηρεσίες που προσφέρει το UMTS είναι η γνώση της θέσης του Κινητού Σταθμού μέσα στο δίκτυο. Αυτή η πληροφορία, εκτός ως υπηρεσία προστιθέμενης αξίας, μπορεί να χρησιμοποιηθεί και από το δίκτυο για εσωτερικούς σκοπούς διαχείρισης, από τον ίδιο τον συνδρομητή και από κάποιον πάροχο υπηρεσιών που βασίζονται στη θέση του UE. Η πληροφορία που μεταδίδεται περιλαμβάνει αυτήν καθ'αυτήν τη θέση του UE τη χρονική στιγμή της εκτίμησης, καθώς και ένα σφάλμα εκτίμησης. Οι μέθοδοι που χρησιμοποιούνται στο UMTS για την εκτίμηση της θέσης του τερματικού είναι πέντε και περιγράφονται συνοπτικά παρακάτω:

1.6.2.5.1 **Εντοπισμός Βασισμένος στην Ταυτότητα της Κυψέλης (Cell ID Based Positioning)**

Με τη μέθοδο αυτή δεν απαιτείται καμία απολύτως μέτρηση στο ραδιοδίκτυο. Το μόνο που χρειάζεται είναι η ταυτότητα της κυψέλης στην οποία έγινε η τελευταία ενημέρωση θέσης του UE. Η θέση είναι πλέον γνωστή είτε ως ταυτότητα κυψέλης (Cell ID) είτε ως συντεταγμένες του Node B. Η ακρίβεια της μεθόδου αυτής είναι πολύ μικρή, αλλά χρησιμεύει σε περιπτώσεις που απαιτείται η γνώση του σε ποια κυψέλη βρίσκεται το τερματικό. Σε ειδική περίπτωση του soft handover, στην οποία το UE έχει ταυτόχρονα επικοινωνία με πολλά Node Bs και άρα αναφέρει τις ταυτότητες πολλών κυψελών, απαιτείται επιπλέον πληροφορία στο Serving RNC ώστε να επιλέξει τη σωστή κυψέλη. Αυτό γίνεται συνήθως επιλέγοντας είτε το Node B με το πιο ισχυρό σήμα, είτε το πρώτο Node B στο οποίο έγινε η εγκατάσταση σύνδεσης, με την ακρίβεια της μεθόδου, προφανώς, να μειώνεται ακόμα περισσότερο.

1.6.2.5.2 **Εντοπισμός Βασισμένος στην Παρατηρούμενη Χρονική Διαφορά της Άφιξης (Observed Time Difference of Arrival, OTDOA)**

Με τη μέθοδο της παρατηρούμενης χρονικής διαφοράς της άφιξης (OTDOA), το UE αναλαμβάνει να μετρήσει τη χρονική διαφορά άφιξης μεταξύ των σημάτων πιλότων (τα οποία μεταφέρουν τον πρωτεύοντα κώδικα περίπλεξης που είναι μοναδικός για κάθε κυψέλη) από γειτονικά Node Bs. Το UE, ουσιαστικά μετρά τις χρονικές διαφορές άφιξης από κάθε Node B σε σχέση με το Node B που το εξυπηρετεί. Αφού μετρήσει όλες τις διαφορές στις αποστάσεις, και γνωρίζοντας τις συντεταγμένες των Node Bs, μπορεί να υπολογίσει μέσω μίας εξίσωσης υπερβολής τη θέση του.

1.6.2.5.3 **Εντοπισμός Βασισμένος στη Συσχέτιση Πιλότων (Pilot Correlation Method)**

Κατά τη μέθοδο συσχέτισης των πιλότων, όλη η περιοχή χωρίζεται σε περιοχές εντοπισμού και δημιουργείται μια βάση δεδομένων με καταχωρήσεις που περιλαμβάνουν την περιοχή εντοπισμού, τα κανάλια πιλότους CPICH που είναι ορατά στην περιοχή και την ισχύ του κώδικα του κάθε CPICH. Το

μέγεθος της περιοχής εντοπισμού εξαρτάται από την επιθυμητή ακρίβεια του εντοπισμού. Το UE καταγράφει τα κανάλια και μετρά την ισχύ του καθενός. Έτσι, μπορεί πλέον να κάνει σύγκριση με τα δεδομένα της βάσης δεδομένων, ώστε να αποφασίσει για την περιοχή εντοπισμού στην οποία ανήκει. Η θέση του UE, τελικά, υπολογίζεται ως το σημείο στο κέντρο της περιοχής εντοπισμού με το ελάχιστο μέσο τετραγωνικό σφάλμα.

1.6.2.5.4 ***Εντοπισμός Βασισμένος στη Συσχέτιση των Προφίλ Καθυστέρησης Ισχύος (Database Correlation Method using Power Delay Profiles)***

Η μέθοδος της συσχέτισης των προφίλ, δημιουργεί μία βάση δεδομένων στο δίκτυο, στην οποία αποθηκεύονται οι συντεταγμένες του σημείου που έχει μετρηθεί, η ταυτότητα του Node B που εξυπηρετεί την περιοχή και το προφίλ καθυστέρησης ισχύος από το Node B στο συγκεκριμένο σημείο. Το UE μετρά την καθυστέρηση ισχύος στη θέση που βρίσκεται και το συγκρίνει με τα δεδομένα της βάσης, με αποτέλεσμα η θέση του UE να είναι αυτή με τη μεγαλύτερη συσχέτιση κατά τη σύγκριση.

1.6.2.5.5 ***Εντοπισμός Υποβοηθούμενος από το Global Positioning System (GPS)***

Στην μέθοδο εντοπισμού με τη χρήση GPS υπάρχουν δύο εναλλακτικές: το UE είναι εφοδιασμένο με πλήρη δέκτη GPS αλλά επιπλέον λαμβάνει και δεδομένα υποβοήθησης από το δίκτυο, ή το UE περιλαμβάνει μια έκδοση GPS δέκτη με μειωμένη πολυπλοκότητα και δυνατότητες. Ανάλογα με την υλοποίηση, είναι δυνατό ο υπολογισμός της θέσης να γίνεται είτε εξ'ολοκλήρου στο UE είτε στο δίκτυο όπου στέλνονται και κάποια δεδομένα από το τερματικό.

1.6.2.6 Διαχείριση Μεταπομπής (Handover στο UMTS)

Στο UMTS υπάρχουν διάφοροι τύποι handover για την αντιμετώπιση τόσο της κινητικότητας όσο και άλλων απαιτήσεων, όπως ο έλεγχος του φορτίου, η παροχή κάλυψης και η προσφορά ποιότητας υπηρεσιών. Ένας αποτελεσματικός αλγόριθμος handover μπορεί να εφαρμοστεί με τη βοήθεια των κατάλληλων πόρων και της διαχείρισης θέσης του χρήστη. Η διαχείριση πόρων σημαίνει ότι υπάρχει τρόπος να εδραιωθεί, συντηρηθεί, απελευθερωθεί και ελεγχθεί μία σύνδεση στο επίπεδο ραδιοπρόσβασης.

Στα UMTS συστήματα, το μεγαλύτερο μέρος της σηματοδότησης ελέγχου μεταξύ του τερματικού και του UTRAN γίνεται μέσω του RRC (Radio Resource Control) πρωτοκόλλου. Κάποιες σημαντικές για τα handovers λειτουργίες του RRC πρωτοκόλλου είναι η επιλογή κυψέλης, οι μετρήσεις του τερματικού, η μετάθεση του SRNS και ο έλεγχος των διαφόρων καναλιών. Πρέπει να τονιστεί ότι οι περισσότερες από τις λειτουργίες του RRC εφαρμόζονται στο RNC.

Ανάλογα με τις απαιτήσεις, υπάρχουν τέσσερα είδη Handovers: Hard, Inter-system, Soft και Softer. Τα δύο τελευταία υποστηρίζονται μόνο από UTRA FDD mode, ενώ τα δύο πρώτα υποστηρίζονται τόσο σε TDD όσο και σε FDD συστήματα. Παρακάτω θα δούμε αναλυτικά τους τέσσερις αυτούς τύπους μεταπομπής, περιγράφοντας τον τρόπο εκτέλεσής τους, καθώς και τις οντότητες δικτύου που λαμβάνουν μέρος κατά τη διάρκεια διεξαγωγής τους.

1.6.2.6.1 *Hard Handover*

Το Hard HO (HandOver) είναι ο τύπος HO κατά τον οποίο η παλιά σύνδεση πέφτει πριν το τερματικό να συνδεθεί εκ νέου με το εκάστοτε δίκτυο ραδιοπρόσβασης. Είναι το είδος του HO που χρησιμοποιείται και στα GSM δίκτυα, όπου κάθε σύνδεση χρησιμοποιεί διαφορετική μπάντα συχνοτήτων. Η μετακίνηση ενός χρήστη σε νέα κυψέλη, για παράδειγμα, συνεπάγεται την αποσύνδεσή του πριν εγκαταστήσει νέα σύνδεση σε διαφορετική target κυψέλη. Ο αλγόριθμος του HO αυτού του τύπου, λοιπόν, είναι πολύ απλός: το κινητό τερματικό εκτελεί ένα HO μόλις η ισχύς του σήματος μίας γειτονικής κυψέλης υπερβεί το κατώφλι της ισχύος σήματος της τρέχουσας κυψέλης.

Στο UMTS τα Hard HOs χρησιμοποιούνται όταν απαιτείται η αλλαγή της μπάντας συχνοτήτων μεταξύ του τερματικού και του UTRAN. Παράλληλα, το Hard HO επιλέγεται όταν ένας κινητός σταθμός σε συγκεκριμένο κανάλι παριέρχεται σε νέα κυψέλη την ώρα που τα Soft ή Softer HOs είναι αδύνατο να

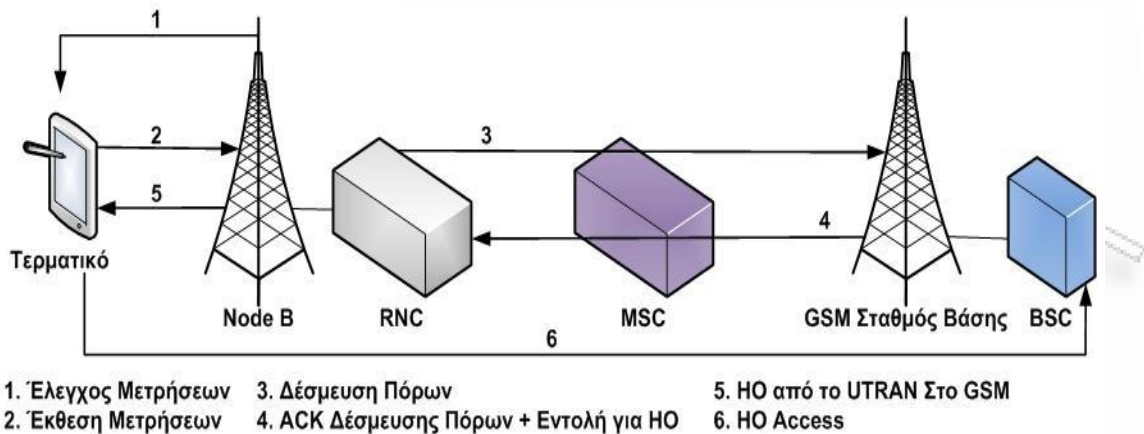
εφαρμοστούν. Το Hard HO είναι, επίσης, ο τύπος HO που χρησιμοποιείται όταν απαιτείται αλλαγή από FDD σε TDD UTRA Mode. Στην περίπτωση αυτή, το εν λόγω HO θεωρείται υποτύπος Hard Handover, γνωστός ως Inter-system HO, τον οποίο θα δούμε πιο κάτω.

Το κυριότερο πρόβλημα που υφίσταται στα Hard HOs είναι τα υψηλά blocking – probabilities για χρήστες που εισέρχονται σε νέα κυψέλη. Η πιθανότητα αυτή μπορεί να μειωθεί αν δοθεί προτεραιότητα σε χρήστες που περιφέρονται όντας ήδη σε κατάσταση επικοινωνίας με το δίκτυο, δεσμεύοντάς τους συγκεκριμένο μέρος της χωρητικότητας μίας κυψέλης. Από την άλλη, αυτό συχνά συνεπάγεται υψηλότερες blocking probabilities και λιγότερο αποτελεσματική χρήση της χωρητικότητας για τους νέους χρήστες. Τα προβλήματα αυτά οδήγησαν στη δημιουργία αλγορίθμων Soft και Softer HO χάρη στους οποίους αυξάνεται η συνολική απόδοση του δικτύου. Πλέον, τα Hard HO χρησιμοποιούνται μόνο για λόγους κάλυψης και φορτίου, ενώ η κινητικότητα υποστηρίζεται από τα Soft και Softer HOs.

Αξίζει, τέλος, να επισημανθεί ότι τα HHOs είναι αρκετά δύσκολα για ένα τερματικό σε CDMA συστήματα, καθώς αυτό εκπέμπει και λαμβάνει συνεχώς, χωρίς να έχει τα απαραίτητα χρονικά κενά για μετρήσεις. Έτσι, οι μετρήσεις λαμβάνονται από το δίκτυο, το οποίο αποφασίζει ποια κυψέλη είναι πιο κατάλληλη για το UE. Μόλις ληφθούν και γνωστοποιηθούν οι απαραίτητες μετρήσεις στο δίκτυο, το τερματικό ξεκινά τη διαδικασία της μεταπομπής.

1.6.2.6.2 Inter – System Handover

Τα Inter-system HOs είναι απαραίτητα για την υποστήριξη της συμβατότητας με άλλες αρχιτεκτονικές συστημάτων. Η διαδικασία σηματοδότησης για τη μεταφορά ενός χρήστη από το UTRAN σε άλλο σύστημα, φαίνεται στην παρακάτω εικόνα, η οποία απεικονίζει ένα παράδειγμα HO από UMTS σε GSM σύστημα, ενώ η ίδια διαδικασία εφαρμόζεται και κατά το HO σε άλλα συστήματα. Σε γενικές γραμμές, η διαδικασία αυτή αποτελείται από τη λήψη μετρήσεων, τη δέσμευση πόρων και την εκτέλεση του HO αλγορίθμου.

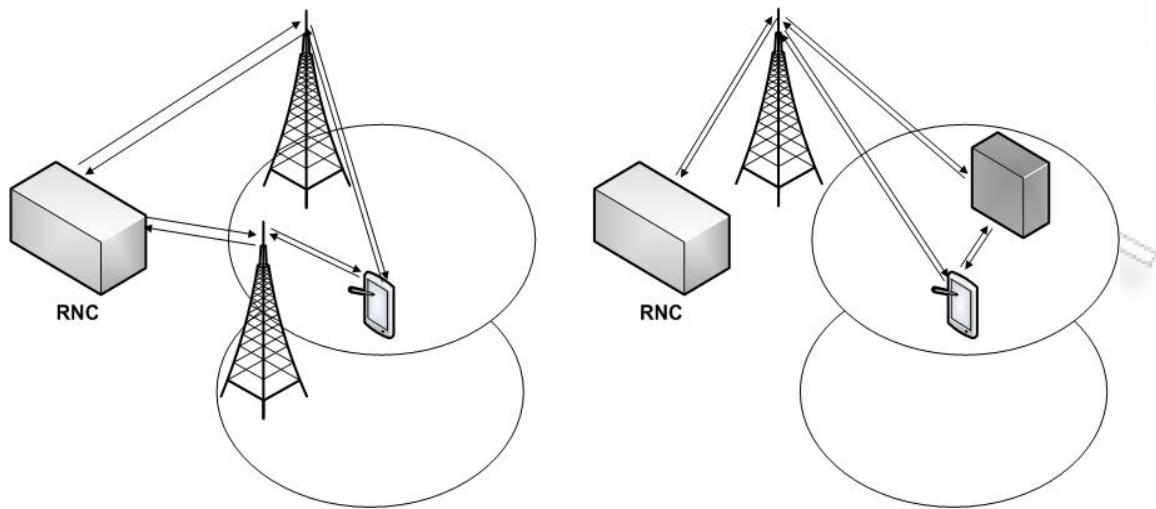


Εικόνα 2. Inter - System Handover

Ας δούμε, λίγο πιο αναλυτικά, τί συμβαίνει κατά το Inter-System HO. Όταν απαιτηθεί η μεταγωγή ενός χρήστη από το UTRAN στο GSM, το Node B του τερματικού αιτεί από το τελευταίο τις μετρήσεις του. Το τερματικό του τις παρέχει, και το Node B στέλνει αίτηση στο RNC ώστε αυτό με τη σειρά του να ζητήσει τη δέσμευση πόρων από το σταθμό βάσης του GSM δικτύου. Εάν όλα πάνε καλά, ο BS του GSM επιστρέφει στον RNC επιβεβαίωση (ACK) δέσμευσης των πόρων, και μία εντολή για Handover. Στο σημείο αυτό, ο RNC ενημερώνει καταλλήλως το Node B, ο οποίος εκτελεί το HO από το UTRAN στο GSM για το τερματικό. Μετά από τη διαδικασία αυτή, το τερματικό έχει πλέον πρόσβαση στο BSC του GSM συστήματος, και το Handover έχει ολοκληρωθεί.

1.6.2.6.3 Soft, Softer Handover

Τα Soft και Softer HOs πραγματοποιούνται όταν ο κινητός σταθμός βρίσκεται στην περιοχή επικάλυψης δύο γειτονικών κυψελών. Εκεί ο χρήστης έχει δύο ταυτόχρονες συνδέσεις στο UTRAN μέρος του δικτύου χρησιμοποιώντας διαφορετικά κανάλια ραδιοεπαφής την ίδια χρονική στιγμή. Στην περίπτωση του Soft HO (SHO) το τερματικό βρίσκεται σε περιοχή που ανήκει σε δύο διαφορετικούς σταθμούς βάσης, ενώ σε αυτήν του Softer HO, λαμβάνει δύο σήματα από έναν BS ο οποίος εξυπηρετεί δύο τομείς μίας κυψέλης.



Εικόνα 3. Soft και Softer Handovers

Αν και υπάρχει μεγάλος βαθμός ομοιότητας ανάμεσα στα δύο αυτά είδη HOs, οι διαφορές είναι πολύ σημαντικές. Κατά το Softer HO ο BS λαμβάνει 2 ξεχωριστά σήματα μέσω πολυδιαδρομικής διάδοσης. Εξ' αιτίας των ανακλάσεων πάνω σε κτήρια ή σε φυσικά εμπόδια, το σήμα που στέλνει το τερματικό φτάνει στο σταθμό βάσης από δύο διαφορετικούς τομείς και αντιμετωπίζεται από τον τελευταίο ως πολυδιαδρομικές συνιστώσες οι οποίες δρομολογούνται σε έναν δέκτη RAKE. Στην downlink κατεύθυνση, ο BS χρησιμοποιεί διαφορετικούς κώδικες διασποράς για κάθε διαφορετικό τομέα που εξυπηρετεί. Έτσι, τα διαφορετικά fingers του δέκτη RAKE του τερματικού εφαρμόζουν τον κατάλληλο de-spreading κώδικα στα σήματα που λαμβάνει από διαφορετικούς τομείς, πριν τους συνδιάσει.

Για το SHO η διαδικασία στο downlink είναι αρκετά όμοια με εκείνη του Softer HO. Στο τερματικό, τα σήματα που λαμβάνονται από διαφορετικούς σταθμούς βάσης συνδιάζονται χρησιμοποιώντας επεξεργασία MRC (Maximum Ratio Combining) RAKE. Στην άνω ζεύξη, από την άλλη, υπάρχουν αρκετές διαφορές. Τα λαμβανόμενα σήματα δε συνδυάζονται στο σταθμό βάσης, αλλά δρομολογούνται προς τον RNC. Εκεί, τα δύο σήματα συγκρίνονται σε frame-by-frame βάση και επιλέγεται το βέλτιστο από αυτά μετά από συγκεκριμένες χρονικές περιόδους (π.χ. κάθε 10, 20, 40, 80 msec). Επειδή ο αλγόριθμος ελέγχου ισχύος μετρά το SNR των λαμβανόμενων uplink σημάτων σε ρυθμούς μεταξύ 10 και 100 Hz, η πληροφορία αυτή χρησιμοποιείται για την επιλογή του frame με την καλύτερη ποιότητα κατά τη διάρκεια του SHO.

Σε γενικές γραμμές, το SHO αποτελείται από δύο βασικές λειτουργίες:

- Λήψη και επεξεργασία μετρήσεων
- Εκτέλεση του HO αλγορίθμου

Όσον αφορά στην πρώτη λειτουργία, τρεις παράμετροι μπορούν να υπολογιστούν: ο λόγος ενέργειας σήματος προς παρεμβολή (E/I), η RSCP (Received Signal Code Power) και η RSSI (Received Signal Strength Indicator). Η RSCP είναι η ισχύς που μεταφέρεται από το αποκωδικοποιημένο κανάλι, ενώ η RSSI είναι η συνολική ευρυζωνική λαμβανόμενη ισχύς. Η σχέση που συνδέει τα τρία αυτά μεγέθη είναι:

$$E/I = RSCP / RSSI.$$

Βασισμένο στις E/ I μετρήσεις των κυψελών που παρακολουθεί, το τερματικό αποφασίζει ποιες από τις τρεις βασικές λειτουργίες να εκτελέσει: προσθήκη (Radio Link Addition), αφαίρεση (Radio Link Removal) ή αντικατάσταση (Combined Radio Link Addition and Removal) ενός Node B στην κυψέλη, και ανάλογα με το αποτέλεσμα της απόφασης, προχωρά στην εκτέλεση του HO αλγορίθμου.

Κατά την προσθήκη ενός Node B, ο RNC πρώτα το ελέγχει και αφού λάβει τις κατάλληλες επιβεβαιώσεις από αυτό, προχωρά στην αποστολή ενημερωτικού μηνύματος στο RRC (Radio Resource Control) του UE. Το μήνυμα αυτό περιέχει όλες τις απαραίτητες για το τερματικό πληροφορίες ώστε αυτό να ξεκινήσει να λαμβάνει τα εκπεμπόμενα σήματα μέσω του νέου Node B. Κατά την αφαίρεση ενός Node B από την άλλη, διαγράφεται το τρέχον Node B, και στέλνεται στο τερματικό εντολή να απενεργοποιήσει την λήψη των σημάτων του. Μόλις γίνει αυτό, σταματά και το Node B να εκπέμπει προς και να λαμβάνει από το UE.

1.6.2.7 Υπηρεσίες

Ένα μοντέρνο τηλεπικοινωνιακό δίκτυο όπως το UMTS μπορεί να παρέχει μια μεγάλη ποικιλία από υπηρεσίες. Τα βασικά χαρακτηριστικά του UMTS έχουν στο μεγαλύτερο μέρος αντιγραφεί από τον κόσμο του GSM. Παρ' όλα αυτά, στο GSM οι παράμετροι είναι συνήθως σταθερές ενώ στο UMTS μπορούν να αλλάζουν δυναμικά όποτε αυτό απαιτείται.

Οι υπηρεσίες που παρέχονται από το UMTS χωρίζονται σε:

- Τηλευπηρεσίες (Teleservices)
- Υπηρεσίες Bearer (Bearer Services)
- Συμπληρωματικές υπηρεσίες (Supplementary and Value Added Services)

1.6.2.7.1 Τηλεϋπηρεσίες

Οι τηλεϋπηρεσίες εξασφαλίζουν από άκρο σε άκρο επικοινωνία μεταξύ χρηστών με βάση προσυμφωνημένα και προτυποποιημένα πρωτόκολλα, χρησιμοποιώντας όλη τη στοίβα πρωτοκόλλων του OSI και συμπεριλαμβάνοντας τις σχετικές λειτουργίες του τερματικού εξοπλισμού. Κάποιες τηλεϋπηρεσίες προτυποποιούνται έτσι ώστε να μπορούν να συνεργάζονται με τηλεϋπηρεσίες που παρέχονται από άλλα δίκτυα, ενώ άλλες χρησιμοποιούνται μόνο από ένα δίκτυο με αποτέλεσμα η προτυποποίησή τους να μην είναι αναγκαία. Κάποιες από τις τηλεϋπηρεσίες είναι οι παρακάτω:

- Φωνητικές κλήσεις
- Κλήσεις έκτακτης ανάγκης (Emergency Calls)
- Μηνύματα κειμένου (SMS)
- Υπηρεσίες cell broadcast προς όλα τα κινητά της κυψέλης
- Τηλεφωνητής (Voice Main)
- Υπηρεσίες κλήσεων ομάδας (Voice Group Call Service)
- Πρόσβαση στο διαδίκτυο

1.6.2.7.2 Υπηρεσίες Bearer

Οι υπηρεσίες bearer είναι βασικές τηλεπικοινωνιακές υπηρεσίες που προσφέρουν πλήρη μετάδοση σημάτων ανάμεσα σε σημεία πρόσβασης. Οι υπηρεσίες αυτές μπορούν να είναι είτε CS είτε PS και αφορούν μόνο τα τρία χαμηλότερα επίπεδα του μοντέλου OSI.

Μια υπηρεσία bearer ορίζεται χρησιμοποιώντας μια ομάδα χαρακτηριστικών που την κάνουν διαφορετική από όλες τις άλλες υπηρεσίες bearer. Αυτά τα χαρακτηριστικά της υπηρεσίας περιλαμβάνουν τον τύπο της κίνησης, τα χαρακτηριστικά της κίνησης και τους υποστηριζόμενους ρυθμούς bit. Πιο συγκεκριμένα, τα χαρακτηριστικά αυτά περιλαμβάνουν:

Τον τύπο της τηλεπικοινωνιακής κίνησης

- Connection – Oriented ή Connectionless
- Constant Bit Rate (CBR)
- Dynamically Variable Bit Rate (VBR)

Τα χαρακτηριστικά της τηλεπικοινωνιακής κίνησης

- Point – to – Point

- Point – to – Multipoint

Την ποιότητα της πληροφορίας

- Μέγιστη Καθυστέρηση Μεταφοράς (Maximum Transfer Delay)
- Ρυθμός Μετάδοσης (Data Rate)
- Μεταβολή στην Καθυστέρηση (Delay Variation)
- Bit Error Rate (BER)

Αυτό που κάνει το UMTS τόσο διαφορετικό από τα άλλα κινητά τηλεπικοινωνιακά συστήματα είναι το ότι επιτρέπει την διαπραγμάτευση αυτών των παραμέτρων ανάμεσα στην εφαρμογή και το δίκτυο. Υπάρχει, δηλαδή, μια ρουτίνα διαπραγμάτευσης στην οποία η εφαρμογή ζητάει μια συγκεκριμένη bearer υπηρεσία, το δίκτυο ελέγχει τις διαθέσιμες πηγές και μετά παρέχει τη ζητούμενη υπηρεσία ή προτείνει ένα χαμηλότερο επιπέδο της υπηρεσίας. Η εφαρμογή στη μεριά του χρήστη είτε δέχεται είτε απορρίπτει την πρόταση του δικτύου. Είναι επίσης πιθανό να επαναδιαπραγματευθούν οι ιδιότητες μιας υπηρεσίας bearer κατά τη διάρκεια μιας ενεργής σύνδεσης στο UMTS. Αυτή η ιδιότητα κάνει την υπηρεσία bearer του UMTS πολύ πιο εύκαμπτη και επιτρέπει ευελιξία στη διαχείριση των πόρων του συστήματος.

1.6.2.7.3 Συμπληρωματικές Υπηρεσίες

Οι Συμπληρωματικές υπηρεσίες (Supplementary Services, SS) υλοποιούν και βελτιώνουν τις Bearer υπηρεσίες και τηλευπηρεσίες. Εξαρτώνται από τον πάροχο υπηρεσιών ή του δικτύου και προστίθενται στις αναφερθείσες υπηρεσίες. Παραδείγματα συμπληρωματικών υπηρεσιών ακολουθούν παρακάτω:

- Προώθηση κλήσεων
- Φραγή εξερχόμενων/ εισερχόμενων κλήσεων
- Πληροφορίες χρέωσης
- Αναμονή κλήσης
- Κράτηση κλήσης
- Τηλεδιάσκεψη
- Αναγνώριση καλούντος
- Κλειστές ομάδες χρηστών
- Δυνατότητα ταυτόχρονων πολλαπλών κλήσεων

1.6.2.8 Ποιότητα Υπηρεσίας στο UMTS (QoS)

Στο UMTS, όπως είπαμε, κάθε κινητός σταθμός μπορεί να διαπραγματεύεται τις παραμέτρους της ποιότητας υπηρεσίας για ένα Radio Bearer. Έχουν καθοριστεί, λοιπόν, 4 κλάσεις υπηρεσιών:

Conversational Real Time Class: Πρόκειται για μια κλάση υπηρεσίας εξαιρετικά ευαίσθητη στην καθυστέρηση μετάδοσης και σχεδόν συμμετρική στις δύο κατευθύνσεις. Παραδείγματα υπηρεσιών είναι η φωνή, η video τηλεφωνία και τα video games. Αποτελεί τη μεγαλύτερη πρόκληση στις υπηρεσίες του UMTS, απαιτώντας μικρές τιμές καθυστέρησης, μικρότερες των 400 msec, καθώς και μικρές αυξομειώσεις στην καθυστέρηση (jitter). Συνεπώς, δεν πρέπει να χρησιμοποιούνται τεχνικές αναμετάδοσης, αλλά ούτε και καταχωρητές στον αποδέκτη, οι οποίοι θα εξασφάλιζαν ελάχιστες μεταβολές στην καθυστέρηση.

Streaming Class: Πρόκειται για μια κλάση λιγότερο ευαίσθητη στην καθυστέρηση μετάδοσης και εξαιρετικά ασύμμετρη, αφού είναι κυρίως προς μια κατεύθυνση και συνεχής. Αφορά κυρίως πολυμεσικές εφαρμογές, όπου σημαντική είναι η σύγχρονη μετάδοση των διαφορετικών μονάδων πληροφορίας π.χ. video και audio. Παραδείγματα είναι το video on demand και το audio on demand. Αξίζει να σημειωθεί ότι εδώ δεν απαιτείται λήψη όλου του προς μετάδοση αρχείου προκειμένου να ξεκινήσει η αναπαραγωγή στο χρήστη. Οι απαιτήσεις ως προς την καθυστέρηση δεν είναι μεγάλες, καθώς είναι της τάξης των 10 sec, αλλά δεν ισχύει το ίδιο για τις αυξομειώσεις της καθυστέρησης και την ασυμμετρία καθυστέρησης μεταξύ των διαφορετικών μονάδων πληροφορίας. Για την εξασφάλιση των απαιτήσεων υπηρεσιών της κλάσης αυτής χρησιμοποιούνται καταχωρητές στο σημείο λήψης. Πρέπει να τονιστεί ότι οι υπηρεσίες αυτές μπορούν να μεταδοθούν από δίκτυα PS.

Interactive Class: Στην κλάση αυτή ανήκουν υπηρεσίες όπου ο χρήστης ζητά δεδομένα από έναν απομακρυσμένο εξυπηρετή και η απάντηση περιλαμβάνει τα επιθυμητά data. Χαρακτηριστικές υπηρεσίες είναι η πλοήγηση στο διαδίκτυο, οι τηλε-αγορές και τα διαδραστικά παιχνίδια. Σημαντική διαφορά μεταξύ αυτής της κλάσης και της conversational είναι η μεγαλύτερη ασυμμετρία που τη χαρακτηρίζει, με το μεγαλύτερο όγκο των δεδομένων να αφορά στην κάτω ζεύξη. Οι απαιτήσεις καθυστέρησης μετάδοσης είναι μικρές, της τάξης μερικών δευτερολέπτων, συνήθως μικρότερες των 4. Στην κλάση αυτή, η πιο σημαντική είναι η ακεραιότητα των δεδομένων.

Background Class: Στην κλάση αυτή ανήκουν υπηρεσίες χωρίς σαφείς περιορισμούς στην καθυστέρηση μετάδοσης, όπως το ηλεκτρονικό

ταχυδρομείο και το SMS. Και εδώ είναι σημαντική η μετάδοση χωρίς σφάλματα, απαίτηση που ικανοποιείται σχετικά εύκολα με δεδομένη τη δυνατότητα χρήσης αναμετάδοσης των πακέτων. Οι αυξομειώσεις στην καθυστέρηση δεν έχουν καμία σημασία, αφού τα δεδομένα παρουσιάζονται στο χρήστη μετά την ολοκληρωμένη λήψη του μεταδιδόμενου αρχείου. Τέλος, και οι απαιτήσεις σε ραδιοπόρους του συστήματος είναι πολύ μικρές.

1.7 ΜΕΛΛΟΝΤΙΚΟΙ ΣΤΟΧΟΙ

Είδαμε ότι η ανάπτυξη και η εξέλιξη των κινητών τηλεπικοινωνιακών συστημάτων και τεχνολογιών πέρασε από αρκετά διακριτά στάδια και φάσεις. Ξεκινώντας από αναλογικές διαμορφώσεις, μικρό πλήθος χρηστών, αυτοματοποιημένες λειτουργίες διαχείρισης τερματικών και δικτύων, καθώς και πολύ βασικές υπηρεσίες σε χαμηλά επίπεδα ποιότητας, φτάνουμε σε ολοκληρωμένα συστήματα με πληθώρα νέων υπηρεσιών και εφαρμογών, συνδυασμό διαμορφώσεων για την καλύτερη επίτευξη ποιότητας και χωρητικότητας δικτύων και σύνθετους μηχανισμούς διαχείρισης οντοτήτων των δικτυακών υποδομών.

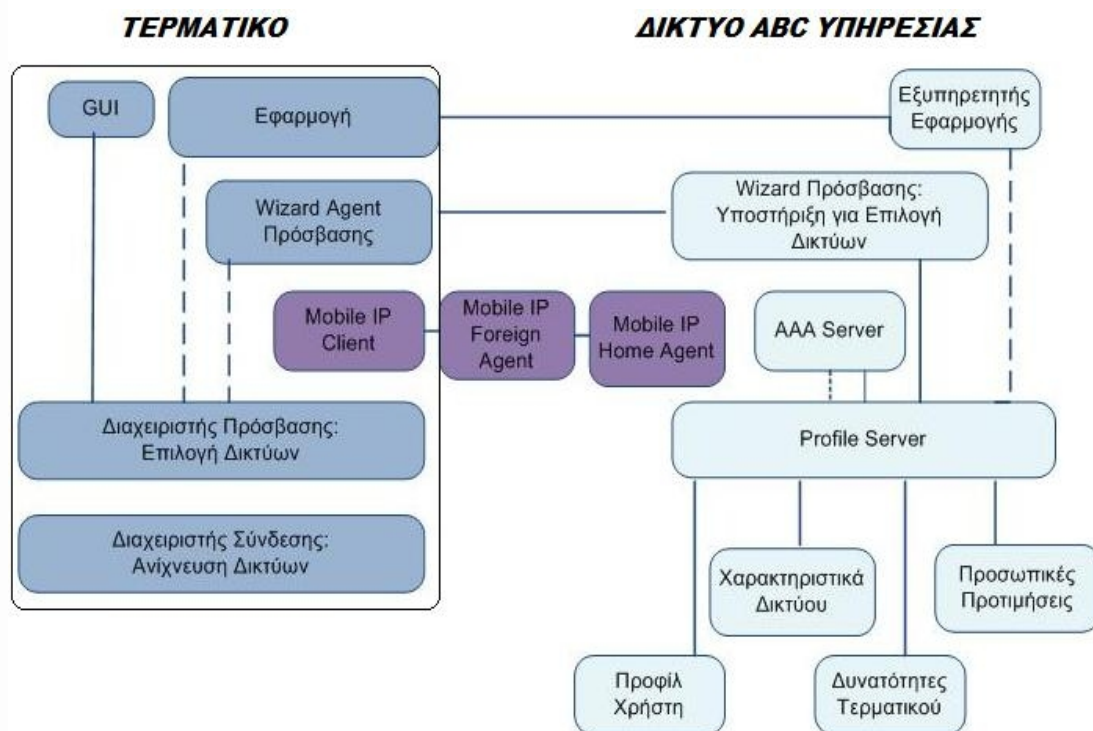
Οι εφαρμογές κινητής τηλεφωνίας των γενεών τεχνολογίας που περιγράφηκαν στις παραπάνω υποενότητες, βέβαια, εστιάζουν στην εξασφάλιση της συνέχειας των υπηρεσιών, υπό την προϋπόθεση ότι οι συσκευές θα συνδέονται στην υποδομή χρησιμοποιώντας μόνο μία τεχνολογία πρόσβασης. Παρ' όλο, δηλαδή, που η αρραγής κινητικότητα και η μεταπομπή από μία τεχνολογία σε άλλη είναι γεγονός, δεν ισχύει το ίδιο για την παράλληλη εκμετάλλευση των δυνατοτήτων που προσφέρουν οι διαφορετικές τεχνολογίες ραδιοπρόσβασης, ταυτόχρονα.

Έτσι, οι επόμενες γενιές τεχνολογίας, στρέφονται στα ετερογενή, πλέον, δίκτυα πρόσβασης (Wi-Fi, Bluetooth, κυψελωτά) με διαφορετικά χαρακτηριστικά που θα συνυπάρχουν με αλληλοσυμπληρούμενο τρόπο. Οι τεχνολογίες αυτές, όπως είναι προφανές, θα ποικίλλουν σε εύρος ζώνης, καθυστέρηση, κάλυψη, απαιτήσεις σε ισχύ, ασφάλεια, αξιοπιστία, πολυπλοκότητα υλοποίησης, κόστος τελικού χρήστη και σε αρκετές άλλες πτυχές, προσφέροντας έτσι η κάθε μία τα δικά της πλεονεκτήματα και χαρακτηριστικά.

Η ύπαρξη πολλών δικτύων πρόσβασης, θα προσφέρει και δυνατότητες επιλογής τους, έτσι ώστε όλες οι εφαρμογές να λαμβάνουν αποδεκτά επίπεδα QoS. Για παράδειγμα, όταν η κατάσταση ενός δικτύου αλλάξει, οι εφαρμογές μπορούν να ανατεθούν σε άλλη καλύτερη τεχνολογία, διαφανώς. Παράλληλα, από την πλευρά της σύνθετης ραδιοϋποδομής, θα γίνει δυνατή η συνεργασία των δικτύων και ISP, με αποτέλεσμα να μοιράζεται το τηλεπικοινωνιακό φορτίο ανάμεσα σε αρκετά δίκτυα, αυξάνοντας έτσι την απόδοση και την χωρητικότητά τους.

Μέσω της σύνθεσης και της συνεργασίας των διαφορετικών δικτύων και τεχνολογιών, εισάγεται μία νέα έννοια στον τηλεπικοινωνιακό χώρο, η Always Best Connected (ABC), χάρη στην οποία οι χρήστες θα μπορούν να παραμένουν αρραγώς συνδεδεμένοι στο δίκτυο με τρόπο που να ταιριάζει

καλύτερα στις ανάγκες των εφαρμογών που τρέχουν. Η αρχιτεκτονική των ABC δικτύων, σε ένα περιβάλλον στο οποίο οι συνθήκες αλλάζουν συνεχώς, θα δώσει τη δυνατότητα στις εφαρμογές να προσαρμόζονται δυναμικά στις αλλαγές αυτές ούτως ώστε η ροή της πληροφορίας, από την οπτική γωνία του χρήστη, να είναι διαφανής ως προς την αλλαγή του δικτύου πρόσβασης ή του γενικού περιβάλλοντος επικοινωνίας. Έτσι, ο χρήστης δε θα αντιλαμβάνεται καμία αλλαγή στις τεχνολογίες πρόσβασης και θα λαμβάνει, οποιαδήποτε στιγμή, την υπηρεσία που επιθυμεί με τον καλύτερο δυνατό τρόπο.



Εικόνα 3. Αρχιτεκτονική του ABC Concept

Μέσω της αρχιτεκτονικής του ABC Concept, η χρήση των ασύρματων τεχνολογιών πρόσβασης περνά από την παθητική λήψη των σχετικά σταθερών δυνατοτήτων μίας τεχνολογίας, στην ενεργητική, πλέον, επιλογή των επιθυμητών χαρακτηριστικών, επιπέδων ποιότητας και κόστους. Δίνεται, δηλαδή, η δυνατότητα στο χρήστη, μέσω ενός Προφίλ Χρήστη, να καταχωρήσει τα επιθυμητά επίπεδα QoS και τα χρηματικά ποσά που επιθυμεί να διαθέσει για την εκάστοτε υπηρεσία, έτσι ώστε το τερματικό, αν εντοπίσει αλλαγή στις συνθήκες του τρέχοντος δικτύου, να αναζητήσει ένα καλύτερο, το οποίο να συμφωνεί με τις επιλογές και απαιτήσεις του χρήστη.

Πριν περάσουμε, όμως, στην ανάλυση των τεχνολογιών και συστημάτων διαχείρισης τερματικών που θα προσφέρουν τις παραπάνω δυνατότητες, θα δούμε την αρχιτεκτονική, τις λειτουργίες και τα επαγόμενα πλεονεκτήματα που θα προσφέρει η σύνθετη ραδιοϋποδομή των επόμενων

γενεών τεχνολογίας τόσο στην πλευρά των δικτύων/ ISPs όσο και στην πλευρά των τελικών χρηστών.

ΕΥΦΥΗ ΔΙΚΤΥΑ ΠΕΡΑΝ ΤΗΣ ΤΡΙΤΗΣ ΓΕΝΙΑΣ

2.1 ΕΙΣΑΓΩΓΗ

Στα υπάρχοντα δίκτυα, ένας συνδρομητής που περιάγεται χρησιμοποιώντας ένα access – specific τερματικό, δεν μπορεί να αποκτήσει υπηρεσίες από άλλα ετερογενή δίκτυα. Εάν η σύνδεσή του επιδεινωθεί λόγω ανεπάρκειας πόρων, χαμηλής ποιότητας υπηρεσίας, μικρής χωρητικότητας δικτύου ή γενικής μεταβολής των συνθηκών της υποδομής, ο χρήστης δεν μπορεί παρά να παραμείνει στην τρέχουσα σύνδεσή του αντιμετωπίζοντας προβλήματα όπως η καθυστέρηση, το jitter, το φαινόμενο near – far, η χαμηλή ποιότητα φωνής και δεδομένων, η χαμηλή στάθμη του σήματος και άλλα. Παράλληλα, ως συνδρομητής σε συγκεκριμένο πάροχο δικτύου, δεν έχει την δυνατότητα να ορίζει την τιμή κόστους που επιθυμεί να διαθέσει για τις υπηρεσίες που λαμβάνει, καθώς κάθε ISP διατηρεί έναν δικό του τιμοκατάλογο ορίζοντας στατικά τις χρεώσεις των συνδρομητών που είναι συνδεδεμένοι σε αυτόν. Μην μπορώντας να μεταβεί σε άλλη τεχνολογία ραδιοπρόσβασης με άλλα χαρακτηριστικά ποιότητας και τιμών, ο χρήστης απαλλάσσεται της δυνατότητας να επιλέξει το ανώτατο ποσό που προτίθεται να δώσει για συγκεκριμένα επίπεδα ποιότητας υπηρεσιών, και λαμβάνει τις εφαρμογές του από το δίκτυο στο οποίο είναι συνδρομητής, με τα QoS επίπεδα και τις τιμές που αυτό ορίζει.

Τη λύση στους περιορισμούς των υπάρχοντων τεχνολογιών ασύρματης πρόσβασης, που περιγράφηκαν παραπάνω, έρχεται να δώσει η Πέραν της Τρίτης Γενιά Τεχνολογίας (B3G), τα δίκτυα της οποίας θα κάνουν δυνατή την αρραγή κινητικότητα ανάμεσα σε ετερογενή δίκτυα πρόσβασης, επιτρέποντας στον συνδρομητή να χρησιμοποιεί οποιουδήποτε τύπου υπηρεσίες, σε οποιοδήποτε τόπο, χρόνο και τρόπο μέσω της διαλειτουργικότητας μεταξύ δικτύων και τερματικών, καθώς και της συνεργασίας των ISPs.

Η στόχος της B3 γενιάς, ουσιαστικά, είναι η ενοποίηση των ετερογενών περιβαλλόντων και δικτύων σε ένα σύνθετο ραδιοπεριβάλλον κάτω από το Internet Protocol Version 6 (IPv6). Η πολυσύνθετη αυτή ραδιο – υποδομή, θα

επιτρέπει καλύτερη διαχείριση πόρων, κινητικότητας και υπηρεσιών μέσω της συνεργασίας των RANs τα οποία πλέον δε θα λειτουργούν ανταγωνιστικά αλλά αλληλοσυμπληρωματικά. Τα διάφορα δίκτυα, δηλαδή, θα μπορούν να μοιράζονται τους πόρους, την χωρητικότητα και το φορτίο τους, λύνοντας έτσι το σημερινούς περιορισμούς που αντιμετωπίζουν.

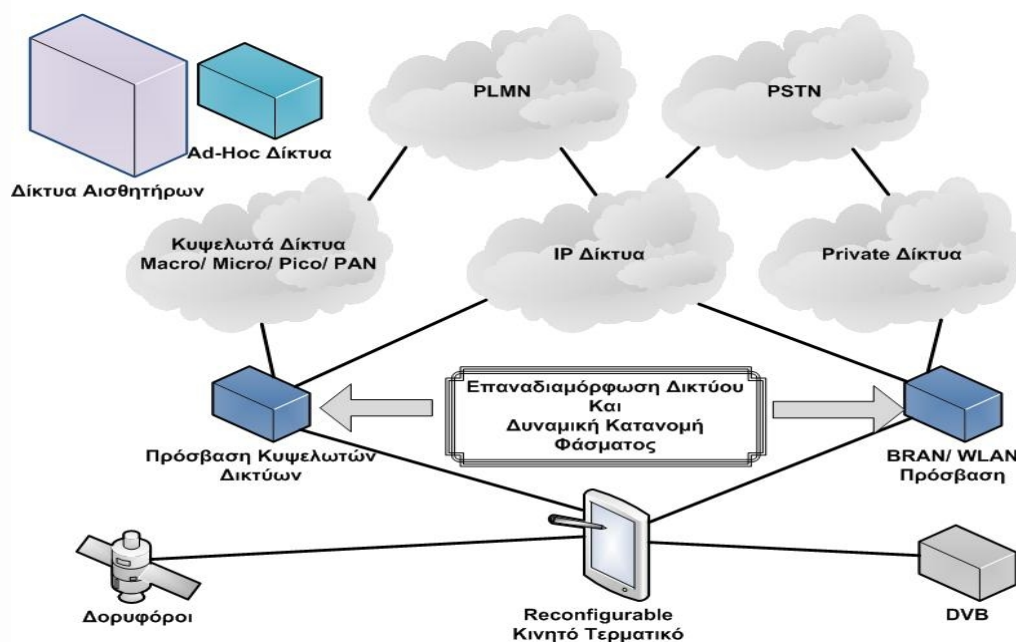
Τα B3G δίκτυα και οι οντότητές τους, λοιπόν, θα διαχειρίζονται συνεργατικά τα handovers ανάμεσα σε ετερογενή περιβάλλοντα για διάφορους βαθμούς κινητικότητας, δεσμεύοντας πόρους μεταξύ των ομότιμων τερματικών για υποστήριξη τόσο RT όσο και NRT εφαρμογών. Παράλληλα, θα υποστηρίζουν από άκρο σε άκρο διαχείριση της ποιότητας υπηρεσιών, κάτω από τις μεταβαλλόμενες συνθήκες των δικτύων. Για να γίνουν δυνατά τα παραπάνω, βέβαια, χρειάζεται και ο κατάλληλος σχεδιασμός των τερματικών έτσι ώστε αυτά να μπορούν να υποστηρίξουν αυτό το reconfigurability concept. Συνεπώς, τα κινητά τερματικά θα είναι προσαρμόσιμα και θα έχουν πρόσβαση σε ετερογενείς υπηρεσίες ασχέτως από τις γεωγραφικές περιοχές στις οποίες βρίσκονται.

Το κεφάλαιο αυτό πραγματεύεται το σύνθετο ραδιο – περιβάλλον της Πέραν της Τρίτης Γενιάς τεχνολογίας. Αναλύεται η αρχιτεκτονική της ενοποιημένης ραδιο – υποδομής, καθώς και τα νέα στοιχεία που προστίθενται για την υποστήριξη της διαχείρισής της και της έννοιας της επαναδιαμόρφωσης. Επιπρόσθετα, εξηγείται ο τρόπος με τον οποίο γίνεται η εκμετάλλευση αυτού του πολυσύνθετου περιβάλλοντος, μέσω της δυναμικής κατανομής φάσματος και την αποτελεσματική διαχείριση της κινητικότητας και των πόρων.

2.2 ΣΥΝΘΕΤΗ Β3G ΡΑΔΙΟ – ΥΠΟΔΟΜΗ

Στην Πέραν της Τρίτης Γενιά της τεχνολογίας, η κοινωνία των ασύρματων επικοινωνιών στρέφεται στην ένωση των τεχνολογιών πρόσβασης σε ένα σύνθετο ραδιοπεριβάλλον (Composite Radio Environment, CRE) και κατά συνέπεια στην έννοια της επαναδιαμόρφωσης (reconfigurability concept). Το CRE προϋποθέτει ότι διαφορετικά ραδιο – δίκτυα θα μπορούν να λειτουργούν ως συνεργαζόμενα μέρη μίας ετερογενούς υποδομής ασύρματης πρόσβασης, μέσω των οποίων οι πάροχοι δικτύων θα επιτυγχάνουν πιο αποτελεσματικά την απαιτούμενη χωρητικότητα και τα επιθυμητά επίπεδα ποιότητας υπηρεσιών (QoS). Η reconfigurability, από την άλλη, θα επιτρέπει στα τερματικά και στα στοιχεία των δικτύων να επιλέγουν δυναμικά και να προσαρμόζονται στις πλέον κατάλληλες τεχνολογίες ραδιοπρόσβασης, αντιμετωπίζοντας έτσι τις διάφορες καταστάσεις που συναντώνται σε διάφορες περιοχές και χρονικές ζώνες της ημέρας.

Σήμερα, υπάρχει μία μεγάλη ποικιλία τεχνολογιών ραδιο – πρόσβασης (Radio Access Technologies, RATs) των οποίων τα πρότυπα χρησιμοποιούνται στις ασύρματες επικοινωνίες. Όπως φαίνεται και στην Εικόνα 4, οι τεχνολογίες αυτές μπορούν να κατηγοριοποιηθούν σε τέσσερις ομάδες:



Εικόνα 4. Σύνθετη Ραδιοϋποδομή Β3G Δικτύων

- Κυβελωτά Δίκτυα

Στην ομάδα αυτή ανήκουν τα συστήματα κινητών επικοινωνιών Δεύτερης (GSM), 2.5 (EDGE, GPRS, IS 136) και Τρίτης Γενιάς (UMTS, WCDMA, cdma2000). Οι δύο πρώτες γενιές, είναι βασισμένες στην TDMA τεχνολογία, ενώ η Τρίτη στην CDMA παρέχοντας ταχύτητες έως 2 Mbps. Οι αρχιτεκτονικές δικτύων περιλαμβάνουν επίσης macro-, micro- και picocellular δίκτυα, HANs (Home Area Network) και PANs (Personal Area Networks).

- BRANs ή WLANs

Στην 4G τεχνολογία τα Broadband Radio Access Networks (BRANs) ή Wireless Local Area Networks (WLANs) αναμένεται να παρέχουν ταχύτητες έως και 1Gbps. Οι τεχνολογίες αυτές βασίζονται στην πολλαπλή πρόσβαση μέσω διαίρεσης ορθογωνικής συχνότητας (Orthogonal Frequency Division Multiple Access, OFDMA) και στην time – space κωδικοποίηση.

- Digital Video Broadcasting (DVB) και Δορυφορικές Επικοινωνίες
- Ad Hoc Δίκτυα και Δίκτυα Αισθητήρων

Παρ' όλο που η 4G κοινωνία είναι ανοιχτή σε προτάσεις για νέες τεχνολογίες ραδιοπρόσβασης, η έννοια του CRE έγκειται στην εκμετάλλευση των δυνατοτήτων των ήδη υπάρχοντων RATs. Η κύρια ιδέα είναι ότι τα διάφορα ασύρματα δίκτυα όπως τα GPRS, UMTS, BRAN/ WLAN, DVB και άλλα, μπορούν να αποτελούν μέρη μίας ετερογενούς υποδομής ασύρματης πρόσβασης. Ένας πάροχος δικτύου (Network Provider, NP) θα μπορεί να κατέχει αρκετές από τις συνιστώσες του CRE (δηλαδή, θα έχει άδειες για να χρησιμοποιεί και να λειτουργεί διάφορα RATs) και να συνεργάζεται με άλλους NPs. Σε κάθε περίπτωση, ο NP θα μπορεί να βασιστεί σε ποικίλλα εναλλακτικά ασύρματα δίκτυα και τεχνολογίες για να φτάσει την επιθυμητή χωρητικότητα και τα επίπεδα QoS, αποτελεσματικά και οικονομικά. Την ίδια στιγμή, οι χρήστες θα κατευθύνονται στα πιο κατάλληλα δίκτυα και τεχνολογίες, σε διαφορετικές περιοχές υπηρεσίας και χρονικές ζώνες, βάσει των απαιτήσεων των προφίλ τους και τα κριτήρια επίδοσης των δικτύων. Έτσι, τα διάφορα RATs θα τρέχουν με έναν αλληλοσυμπληρούμενο τρόπο, αντί να ανταγωνίζονται το ένα το άλλο, δίνοντας τη δυνατότητα για ευέλικτη διαχείριση συνδέσεων, βελτίωση των QoS επιπέδων και αύξηση της συνολικής χωρητικότητας του συστήματος.

Τα δίκτυα θα παρέχουν ασύρματη πρόσβαση σε IP – based εφαρμογές καθώς και συνέχεια ροής υπηρεσιών μέσω διασυστημικής κινητικότητας. Η ενοποίηση των συστατικών δικτύου στην CR υποδομή θα επιτυγχάνεται μέσω ενός συστήματος διαχείρισης του σύνθετου ραδιοπεριβάλλοντος το οποίο θα ενσωματώνεται σε κάθε δίκτυο. Το σύστημα διαχείρισης του κάθε δικτύου θα διαχειρίζεται μία συγκεκριμένη ραδιο – τεχνολογία, παρ’ όλο που οι πλατφόρμες θα μπορούν να συνεργάζονται. Το σταθερό δίκτυο (core και backbone) θα αποτελείται από δημόσια και ιδιωτικά τμήματα βασισμένα στην IPv4 και IPv6 – based υποδομή, με το Mobile IP να επιτρέπει τη διατήρηση της συνδετικότητας IP επιπέδου, ασχέτως από τις ενδεχόμενες αλλαγές στις ραδιο – τεχνολογίες της σύνθετης υποδομής.

2.3 ΑΡΧΙΤΕΚΤΟΝΙΚΗ Β3G ΔΙΚΤΥΩΝ

Για την υποστήριξη της επαναδιαμόρφωσης των διαφόρων δικτυακών οντοτήτων (όπως σημεία πρόσβασης, σταθμοί βάσης, τερματικά και άλλα) προς διαφορετικές ραδιο – τεχνολογίες, απαιτείται μία αντίστοιχη reconfigurable αρχιτεκτονική δικτύου που θα παρέχει αρκετά πλεονεκτήματα λόγω της ολοκλήρωσης, καθώς και ώθηση προς νέες τοπολογίες δικτύων ραδιο – πρόσβασης.

Η δικτυακή δομή, δηλαδή, θα πρέπει να υποστηρίζει την έννοια του terminal reconfiguration (π.χ. ανίχνευση και εντοπισμό RATs, διαδικασίες λήψης αποφάσεων για handover, μεταφορτώσεις λογισμικού), απελευθερώνοντας έτσι το τερματικό από υπολογιστικές διαδικασίες που ενδεχομένως να επιδρούν αρνητικά στην απόδοση των ενεργών υπηρεσιών. Η αρχιτεκτονική των δικτύων που θα υποστηρίζει τα παραπάνω, θα πρέπει, συνεπώς, να είναι ανοιχτή και ευέλικτη, και να παρέχει μία κοινή πλατφόρμα για πολλαπλές ραδιο – επαφές, διαφορετικά πρωτόκολλα και ποικίλες εφαρμογές, κάτω από τον έλεγχο ενός κοινού software περιβάλλοντος.

Για την υποστήριξη της reconfigurability έννοιας έχουν προταθεί αρκετές και διαφορετικές αρχιτεκτονικές δικτύων με οντότητες υποδομής που θα διευκολύνουν τις inter – operator διαπραγματεύσεις, όσον αφορά στην ανταλλαγή πληροφοριών που απαιτούνται για το terminal reconfiguration και τη διαχείριση μεταπομπών. Οι Dillinger et al στην εργασία τους «Network Functions Supporting Reconfigurations in B3G Environments» κινούνται γύρω από την ιδέα της διατήρησης της υπάρχουσας υποδομής των δικτύων, προσθέτοντας κάποιες νέες οντότητες τις οποίες θα δούμε παρακάτω.

Στα Β3G δίκτυα, όταν αυτό απαιτείται, τα τερματικά θα μεταφορτώνουν λογισμικό ώστε να μπορούν να υποστηρίξουν νέα RATs και νέες υπηρεσίες που ενδέχεται να προκύψουν. Συνεπώς, τα δίκτυα θα πρέπει να έχουν εξυπηρετητές και διακομιστές που θα ελέγχουν τα προφίλ των χρηστών και θα εκτελούν τις εκάστοτε λειτουργίες μεταφόρτωσης του software. Οι οντότητες αυτές, καθώς και οι λειτουργίες τις οποίες εκτελούν συνοψίζονται ως εξής:

- Proxy Reconfiguration Manager (PRM)

Ο PRM τοποθετείται σε πολλές, διαφορετικές περιοχές του δικτύου ραδιοπρόσβασης (RAN) και λειτουργεί ως διακομιστής μεταφόρτωσης (download proxy) εκτελώντας βραχυπρόθεσμη caching. Υποστηρίζει τόσο τις ξεχωριστές, ενεργοποιημένες από το τερματικό, μεταφορτώσεις λογισμικού, όσο και τις ενεργοποιημένες από τους προμηθευτές, μαζικές software αναβαθμίσεις λογισμικού, χρησιμοποιώντας μηχανισμούς πολυεκπομπής. Ο

PRM, επιπρόσθετα, πρέπει να υποστηρίζει την επανάληψη των μεταφορτώσεων λογισμικού σε περιπτώσεις που αυτές διακοπούν.

- Serving Reconfiguration Manager (SRM)

Ο SRM τοποθετείται σε διάφορες περιοχές του δικτύου κορμού (CN) και λειτουργεί ως σύστημα κατανεμημένων εξυπηρετών μεταφόρτωσης λογισμικού. Παρέχει το απαιτούμενο λογισμικό τους PRMs (το οποίο ανακτά από τον HRM) και εκτελεί μακροπρόθεσμη caching για λόγους κλιμάκωσης.

- Home Reconfiguration Manager (HRM)

Ο HRM διαχειρίζεται το CSR και τα SDBs (βλ. παρακάτω). Παρέχει υποστήριξη ατομικών μεταφορτώσεων λογισμικού και συντονίζει τις μαζικές software αναβαθμίσεις. Παράλληλα, επικοινωνεί με εξωτερικούς προμηθευτές λογισμικού. Σε αντίθεση με τους αναφερθέντες διαχειριστές, ο HRM δεν τοποθετείται σε διαφορετικά μέρη ενός δικτύου, καθώς κάθε πάροχος κινητού δικτύου μπορεί να τρέξει μόνο έναν HRM.

- Central Software Repository (CSR)

Το CSR συσχετίζεται με κάθε χρήστη ξεχωριστά και λειτουργεί ως διατηρούμενη μακροπρόθεσμη μνήμη για το λογισμικό και το προφίλ που έχουν «κατέβει» ή «ανέβει» από εξωτερικούς προμηθευτές λογισμικού.

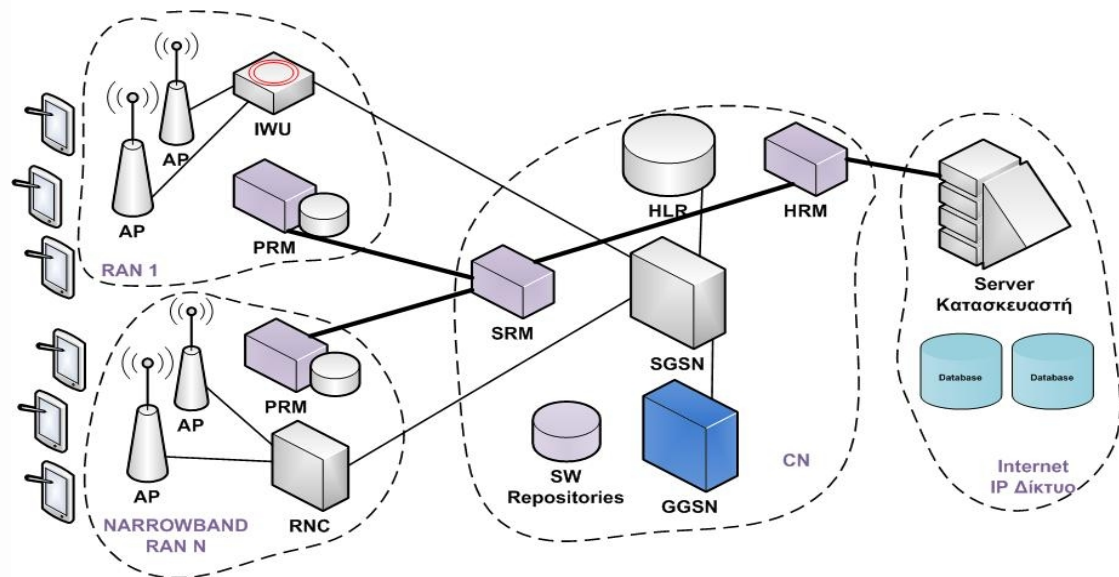
- Software Download Box (SDB)

Τα SDBs είναι υπεύθυνα για το φιλτράρισμα και τον έλεγχο καθυστέρησης των αναβαθμίσεων λογισμικού, για λογαριασμό του τερματικού χρήστη. Διαχειρίζονται τη μεταφόρτωση των αποθηκευμένων στο CSR αναβαθμίσεων λογισμικού, με ασύγχρονο τρόπο. Σε κάθε δίκτυο, διατηρείται ένα SDB ανά χρήστη.

- Gateway Reconfiguration Manager (GRM)

Ο GRM τοποθετείται σε αρκετά σημεία ενός domain κινητού δικτύου και είναι υπεύθυνος για την inter – operator επικοινωνία που σχετίζεται με τη μεταφόρτωση λογισμικού και τη reconfiguration διαχείριση γενικότερα. Πιο

συγκεκριμένα, οι GRMs αντιμετωπίζουν θέματα ιδιωτικότητας και επιλαμβάνονται της ανταλλαγής της σχετικής με το χρήστη πληροφορίας.



Εικόνα 4. Οντότητες Διαχείρισης σε B3G Δίκτυα

Προκειμένου η μεταφόρτωση και ο συγχρονισμός λογισμικού και προφίλ να γίνει αποτελεσματικά, κάθε διαχειριστής έχει τα δικά του αποθηκευτικά μέσα. Ενώ, λοιπόν, ο HRM μπορεί να αποθηκεύει λογισμικό με διατηρούμενο τρόπο, οι SRMs εκτελούν μακροπρόθεσμη caching του λογισμικού που ανέκτησαν από τον HRM. Οι PRMs, από την άλλη, εκτελούν μόνο βραχυπρόθεσμη caching, επιτρέποντας την επανάληψη των μεταφορτώσεων που διεκόπησαν (είτε είναι ατομικές μεταφορτώσεις λογισμικού είτε είναι αναβαθμίσεις λογισμικού), χωρίς να απαιτείται η εκ νέου επαναμεταφόρτωσή τους από το αντίστοιχο SRM. Έτσι, ανάλογα με την περίπτωση και την απαίτηση σε λειτουργίες μεταφόρτωσης, επιλέγεται ο κατάλληλος manager ώστε η έννοια του reconfigurability να υποστηρίζεται συνεχώς, με τον καλύτερο δυνατό τρόπο. Στην Εικόνα 11, μπορούμε να δούμε τις αναφερθείσες οντότητες που εισάγονται στα B3G δίκτυα, καθώς και τις συνδέσεις μεταξύ τους.

Όσον αφορά τη συνολική αρχιτεκτονική των B3G δικτύων, πρέπει να αναφερθεί ότι εξαιτίας των διαφορετικών φυσικών χαρακτηριστικών και πρωτοκόλλων, κάθε δίκτυο ραδιοπρόσβασης αποτελείται από διαφορετικούς σταθμούς βάσης και κόμβους ραδιοελέγχου. Για να γίνει αποτελεσματικά η ενοποίησή τους, οι οντότητες αυτές συνδέονται στο κοινό δίκτυο κορμού μέσω κόμβων υποστήριξης οι οποίοι μπορούν να είναι τα MSCs για τα κυψελωτά δίκτυα, τα IP L1/L1 switches για τα WLANs ή τα FESs (Fixed Earth Station)

για τις δορυφορικές επικοινωνίες. Παράλληλα, η πρόσβαση τόσο σε PSTN/ISDN δίκτυα όσο και στο Διαδίκτυο, παρέχεται μέσω διαδικτυακών μονάδων, πυλών φωνής, firewalls και Gateway Support Nodes. Επιπλέον εξυπηρετητές (π.χ. DNS) παρέχουν συμπληρωματικές υπηρεσίες στο IP domain, ενώ οι mobility servers εκτελούν την αντιστοίχιση μεταξύ των Προσωπικών Αριθμών Τηλεπικοινωνίας και των δυναμικών IP διευθύνσεων.

2.4 ΔΥΝΑΜΙΚΗ ΚΑΤΑΝΟΜΗ ΦΑΣΜΑΤΟΣ

Έχοντας αναλύσει την αρχιτεκτονική των δικτύων της Πέραν της Τρίτης Γενιάς, σκόπιμο είναι, στο σημείο αυτό, να δούμε με ποιόν τρόπο θα γίνεται πραγματικότητα η από άκρο σε άκρο reconfigurability των δικτύων, μέσα από τις λειτουργίες τόσο των νέων όσο και των υπαρχόντων δικτυακών οντοτήτων. Στην παρούσα ενότητα, λοιπόν, θα εξετάσουμε ένα από τα πιο βασικά πλεονεκτήματα που εισάγει η Β3 γενιά της τεχνολογίας από τη σκοπιά των δικτύων: την δυναμική κατανομή φάσματος.

2.4.1 Κίνητρα για Δυναμική Κατανομή Φάσματος

Ως γνωστόν, τα περισσότερα δίκτυα τηλεπικοινωνιών υπόκεινται σε εναλλαγές όσον αφορά τις απαιτήσεις φορτίου, ανάλογα με την ώρα και την περιοχή λειτουργίας, πράγμα που οδηγεί σε αυξομειώσεις του χρησιμοποιούμενου φάσματος. Συνεπώς, ένα φάσμα συχνοτήτων μίας υπηρεσίας μπορεί να χρησιμοποιείται ελάχιστα σε συγκεκριμένες περιοχές ή ώρες, την ώρα που μία άλλη υπηρεσία στον ίδιο τόπο/ χρόνο αντιμετωπίζει έλλειψη. Γνωρίζοντας την υψηλή οικονομική αξία του φάσματος συχνοτήτων και τη μεγάλη σημασία της φασματικής επάρκειας γενικότερα, γίνεται εμφανής η ανάγκη για αποφυγή της σπατάλης του.

Τα παραπάνω ζητήματα έγιναν το κίνητρο για τη δημιουργία μίας μεθόδου που ονομάζεται Δυναμική Κατανομή Φάσματος (Dynamic Spectrum Allocation, DSA), η οποία στοχεύει στη διαχείριση του χρησιμοποιούμενου φάσματος μέσω ενός ασύρματου συστήματος, και στο διαμοιρασμό του ανάμεσα στα ασύρματα δίκτυα ώστε να αυξηθεί η συνολική φασματική επάρκεια. Τα σύνθετα ραδιοσυστήματα και η έννοια της επαναδιαμόρφωσης, που συζητήθηκαν στην προηγούμενη ενότητα, είναι πιθανό να υποβοηθήσουν στη δημιουργία και τη λειτουργία DSA συστημάτων, αφού η συνεργασία γειτονικών δικτύων μπορεί να οδηγήσει στο μερισμό όχι μόνο υπηρεσιών (όπως θα δούμε παρακάτω) αλλά και φάσματος.

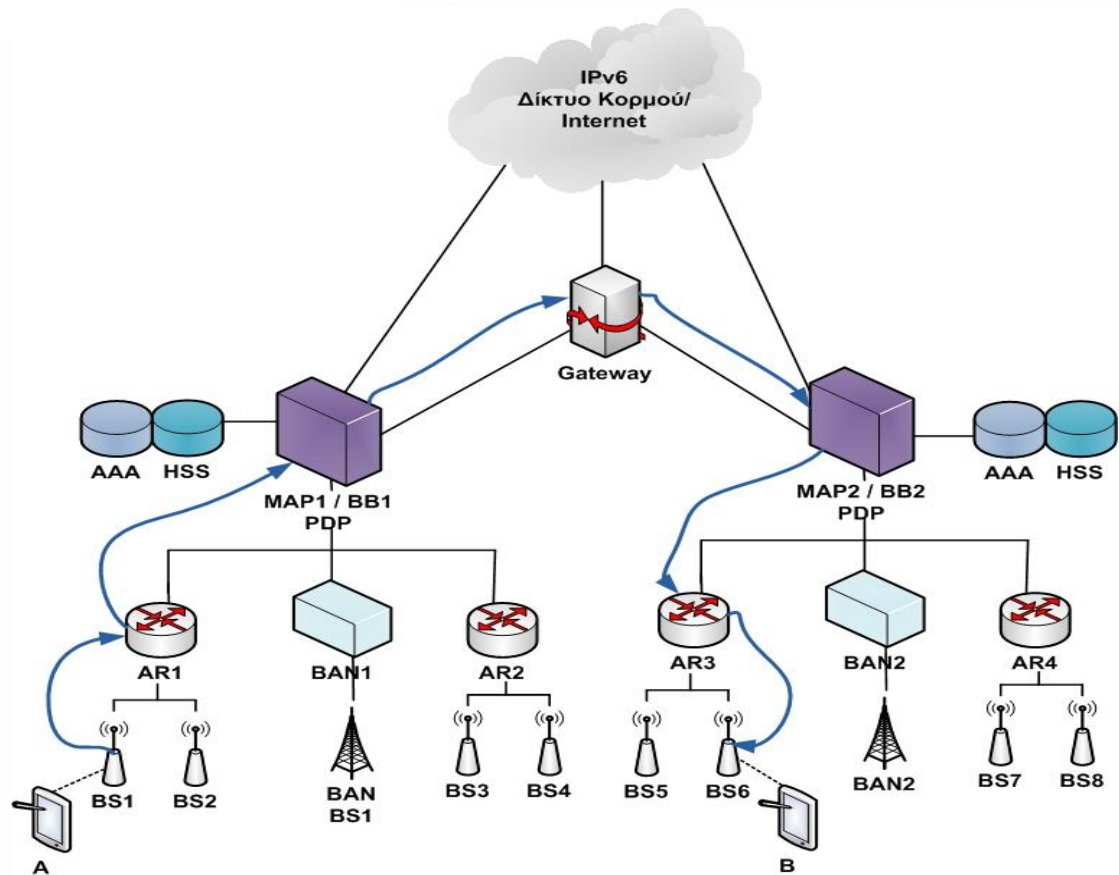
Παράλληλα, η reconfigurability αποτελεί επίσης πολύ σημαντικό παράγοντα στο υπό συζήτηση θέμα, καθώς με ένα DSA σύστημα, σε ένα δίκτυο ασύρματης πρόσβασης θα μπορούσε να αποδοθεί οποιαδήποτε συχνότητα σε οποιοδήποτε τόπο και χρόνο. Πρέπει να σημειωθεί, βέβαια, ότι το επίπεδο εφαρμογής θα πρέπει να αναβαθμιστεί έτσι ώστε να μπορεί να

συγχρονίζει τις ποικίλες ροές πληροφορίας της ίδιας εφαρμογής, η οποία θα μπορούσε να μεταφέρεται ταυτόχρονα μέσω διαφορετικών RATs.

2.4.2 Αρχιτεκτονική Δικτύου για Υποστήριξη Δυναμικής Κατανομής Φάσματος

Όπως είπαμε και προηγουμένως, η διαχείριση φάσματος έχει να κάνει με την κατανομή/ ανακατανομή του φάσματος ώστε να διατηρηθούν οι τρέχουσες σύνοδοι επικοινωνίας. Αυτή η κατανομή/ ανακατανομή γίνεται μέσα στον υπηρετούντα domain και ανάμεσα στη διαδρομή επικοινωνίας μεταξύ ομότιμων τερματικών, πριν και μετά από μία σύνοδο. Στη βιβλιογραφία έχουν προταθεί πολλές μέθοδοι διαχείρισης φάσματος με τη χρήση ενός συγκεντρωτικού bandwidth broker (BB) τον οποίον θα μοιράζονται αρκετά domains. Οι μέθοδοι αυτοί, όμως, αυξάνουν το φορτίο επεξεργασίας καθώς και το ρίσκο, μιας και εάν υποστεί κάποια ζημιά ο BB, θα αποτύχει συνολικά το spectrum management system.

Έτσι, προτείνεται μία αρχιτεκτονική κατανενημένων BBs εντός του domain, με το κάθε router να είναι ικανό να πάρει αυτό το ρόλο (αν και ο βασικός υποψήφιος για αυτή τη δουλειά φαίνεται να είναι το Σημείο Πρόσδεσης Κινητικότητας (Mobility Anchor Point, MAP)). Στο χαμηλότερο στρώμα της δομής αυτής, όπως φαίνεται και στην Εικόνα 5, βρίσκονται οι Base Stations (BSs) οι οποίοι ελέγχονται από τους Access Routers (ARs). Όλα τα ετερογενή δίκτυα μέσα στο domain προσπελαίνουν το διαδίκτυο μέσω του MAP, με το Gateway (GW) να τα συνδέει, επιτυγχάνοντας έτσι ταχύτερη συνδεσιμότητα και μεταφορά δεδομένων. Επιπρόσθετα, κάθε MAP διατηρεί έναν Home Subscriber Server (HSS) ο οποίος αποθηκεύει τα user profiles και την πολιτική υπηρεσιών, καθώς και έναν AAA server που κρατά τις πληροφορίες αυθεντικοποίησης και πιστοποίησης.



Εικόνα 5. Αρχιτεκτονική για Δυναμική Κατανομή Φάσματος

Το MAP, τα ARs και το BAN (Basic Area Network) επικοινωνούν μεταξύ τους μέσω ποικίλων πρωτοκόλλων σηματοδότησης επιπέδου 2, όπως το Point – to – Point Protocol (PPP) και Layer Two Tunneling Protocol (L2TP), με σκοπό να δημιουργήσουν μία λίστα πόρων που προσφέρονται από τα ετερογενή δίκτυα. Το BAN στέλνει περιοδικά με πολυεκπομπή τη λίστα των διαθέσιμων τεχνολογιών πρόσβασης στην περιοχή κάλυψής του, μέσω ενός αξιόπιστου, χαμηλού ρυθμού δεδομένων, καναλιού. Η λίστα αυτή περιέχει τα ID επιπέδου 2 των Σταθμών Βάσης, τις IP διευθύνσεις των αντίστοιχων ARs καθώς και έναν περιορισμένο αριθμό QoS παραμέτρων (π.χ. υποστηριζόμενος ρυθμός δεδομένων, ζώνη συχνοτήτων, ρυθμός κωδικοποίησης βίντεο και άλλα).

2.4.3 Λειτουργίες Οντοτήτων Δικτύων

Στην αρχιτεκτονική καταμεμημένων BBs που περιγράφηκε στην προηγούμενη ενότητα, κάθε router μπορεί να λάβει το ρόλο του Bandwidth Broker. Η διαδικασία επιλογής για το ποιο router μπορεί να γίνει BB βασίζεται στο πόσο χρόνο ο υποψήφιος δρομολογητής μπορεί να παραμείνει online ή στο αν είναι edge router. Για να συμπεράνει ένας δρομολογητής που βρίσκεται στις άκρες ενός domain ότι είναι edge router, συγκρίνει τη διεύθυνσή του με τις ιεραρχικές διευθύνσεις πηγής των γειτονικών δρομολογητών. Σε περίπτωση που ο κύριος BB αποτύχει, τη θέση του παίρνει ένας backup BB router, αφού περάσει την αναφερθείσα διαδικασία επιλογής.

Τα λειτουργικά στοιχεία του BB router θα πρέπει να περιλαμβάνουν τα παρακάτω:

- Συμφωνητικό επιπέδου υπηρεσίας (Service Level Agreement, SLA)
- Προδιαγραφές επιπέδου υπηρεσίας (Service Level Specifications, SLS)
- Βάση δεδομένων του γειτονικού BB
- Υπηρεσία ελέγχου αποδοχής (Admission Control Service, ACS)
- Αυθεντικοποίηση
- Εξουσιοδότηση
- Σημείο απόφασης πολιτικής (Policy Decision Point, PDP)

Στη διαδρομή ανταλλαγής πληροφοριών ανάμεσα σε δύο routers που λειτουργούν ως BBs μπορεί να χρησιμοποιηθεί ένα τούνελ εικονικού ιδιωτικού δικτύου (Virtual Private Network, VPN) το οποίο θα κρυπτογραφεί τη σύνδεση έτσι ώστε να παρέχεται πρόσθετη ασφάλεια.

Στις άκρες του δικτύου θα χρησιμοποιείται το Resource Reservation Protocol (RSVP), ενώ στο δίκτυο κορμού για τις σταθερές συνδέσεις θα χρησιμοποιείται το DiffServ. Επίσης, ένας Διαχειριστής Εύρους Ζώνης Υποδικτύου (Subnetwork Bandwidth Manager, SBM) που θα βρίσκεται στο AR, θα παρέχει έλεγχο αποδοχής για τις ασύρματες συνδέσεις.

2.4.4 Διαδικασία Κατανομής Φάσματος

Στην υποενότητα αυτή θα δούμε ποιά είναι η διαδικασία κατανομής φάσματος για δύο τερματικά που επικοινωνούν από διαφορετικούς domains. Έστω ότι ένα τερματικό A είναι συνδεδεμένο στο σταθμό βάσης BS1 και επιθυμεί να επικοινωνήσει με το τερματικό B που συνδέεται με το BS6 (Εικόνα 12). Αρχικά, ο Bandwidth Broker BB1 θα εξακριβώσει μέσω ενός AAA server το αν ο χρήστης του τερματικού A έχει άδεια πρόσβασης στους πόρους του domain του και το αν η αίτηση είναι για inter-domain ή intra-domain προορισμό. Σε περίπτωση inter-domain αίτησης, ο BB1 θα δημιουργήσει έναν χάρτη (πίνακα δρομολόγησης) με τους γειτονικούς BBs οι οποίοι ελέγχουν διαφορετικά δίκτυα και θα ενσωματώσει ένα Policy Decision Point (PDP) στο οποίο θα γίνεται ο έλεγχος αποδοχής στο domain του, βασισμένος σε προηγούμενες διαχειριστικές ή δυναμικές πολιτικές. Σε μία απλή περίπτωση κάτι τέτοιο θα περιλάμβανε τον έλεγχο για διαθεσιμότητα φάσματος στους edge routers (π.χ. AR1 και AR2 για το domain του MAP1).

Κατόπιν, ο BB1 θα συντονιστεί με το BB2 για να εξακριβώσει το αν ο προορισμός υπάρχει στο domain του BB2 και το αν επαρκεί το φάσμα στο domain του BB2 ώστε να υποστηρίξει το φορτίο από το A στο B. Εάν όχι, ο BB2 θα ξεκινήσει διαπραγματεύσεις με το BB1, αναφορικά με τις απαιτήσεις του τερματικού B. Στην περίπτωση που ανάμεσα στην πηγή και τον προορισμό υπάρχουν αρκετοί BBs, ο συντονισμός του bandwidth θα πρέπει να γίνει σε κάθε έναν από τους ενδιάμεσους Bandwidth Brokers. Εάν η αίτηση για πόρους από τον BB2 γίνει δεκτή, οι δρομολογητές πρόσβασης θα πρέπει να ελέγχουν το τερματικό A ώστε αυτό να μην υπερβεί το ποσό του φάσματος για το οποίο πληρώνει. Το AR μπορεί επίσης να πληροφορεί το AAA server για τα επίπεδα χρήσης φάσματος του A, πράγμα που είναι ακόμα πιο σημαντικό στην περίπτωση των συνδρομητών που έχουν προπληρώσει για τις υπηρεσίες που τους παρέχονται.

Για το παραπάνω παράδειγμα, η διαδρομή της λειτουργίας διαχείρισης φάσματος, απεικονίζεται με την μπλε γραμμή της εικόνας 5.

2.5 ΔΙΑΧΕΙΡΙΣΗ ΘΕΣΗΣ ΚΑΙ PAGING ΤΕΡΜΑΤΙΚΩΝ

2.5.1 Περιορισμοί στην Τρέχουσα Διαχείριση Θέσης

Τα σημερινά δίκτυα χωρίζουν την επιφάνειά τους σε ένα πλήθος από Location Areas (LA). Κάθε LA περιέχει έναν αριθμό κυψελών και κάθε τερματικό που εισέρχεται σε νέα LA εκτελεί ένα location update. Όταν, λοιπόν, καταφθάνει μία εισερχόμενη κλήση, το δίκτυο εντοπίζει το τερματικό στέλνοντας μικρά paging μηνύματα ταυτόχρονα σε όλες τις κυψέλες μέσα στο συγκεκριμένο LA. Τα κυριότερα μειονεκτήματα αυτής της διαδικασίας εντοπισμού και paging συνοψίζονται ως εξής:

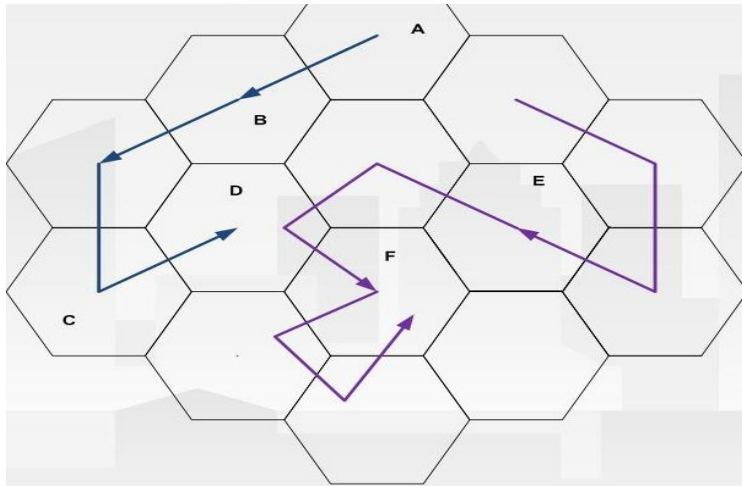
- Η απαίτηση από το δίκτυο να στέλνει μηνύματα σε όλες τις κυψέλες κάθε φορά που καταφθάνει μία κλήση, μπορεί να προκαλέσει μεγάλη αύξηση του φορτίου στο δίκτυο.
- Οι μορφές κινητικότητας και άφιξης κλήσεων μπορούν να ποικίλλουν από τερματικό σε τερματικό, με αποτέλεσμα η επιλογή μεγέθους μίας LA που να ικανοποιεί όλους τους χρήστες, να είναι δύσκολη.
- Σε περιπτώσεις που τα τερματικά βρίσκονται δίπλα στα σύνορα κυψελών και κάνουν συχνές αλλαγές από μία κυψέλη στην άλλη και πίσω, μπορούν να προκαλέσουν την αποστολή πολύ μεγάλου αριθμού paging μηνυμάτων, αυξάνοντας το συνολικό φορτίο στο δίκτυο.

Παράλληλα, η βασισμένη σε LAs μορφή ενημέρωσης της θέσης του χρήστη είναι στατική, αφού δεν μπορεί να προσαρμόζεται στις μεταβαλλόμενες παραμέτρους ενός τερματικού. Πρόσφατες έρευνες για τα B3G συστήματα, έχουν προσπαθήσει να μειώσουν τα παραπάνω μειονεκτήματα, προτείνοντας αρκετές λύσεις όπως timer – based paging στρατηγικές, εντοπισμούς τερματικών βασισμένους στην τοπολογία των κυψελών, δημιουργία διεπαφών ανάμεσα σε paging στοιχεία δικτύου και άλλες. Πολλές από τις προσπάθειες εστιάζουν σε δυναμικούς μηχανισμούς εντοπισμού θέσης, οι οποίοι εκτελούν ενημέρωση της θέσης του χρήστη βασισμένη στην κινητικότητα του τερματικού και τη συχνότητα των εισερχόμενων κλήσεων. Παρακάτω, θα δούμε κάποιους από αυτούς τους μηχανισμούς.

2.5.2 Προταθείσες Μέθοδοι Ενημέρωσης Θέσης Χρήστη

Η σημερινές μέθοδοι ενημέρωσης της θέσης δεν επιτρέπουν την προσαρμογή στα χαρακτηριστικά κινητικότητας του τερματικού. Οι λύσεις που προτείνονται για την B3G, πρέπει να επιτρέπουν τη δυναμική επιλογή των παραμέτρων ενημέρωσης θέσης, καταλήγοντας σε χαμηλότερο κόστος. Η δυναμική διαχείριση LA (Dynamic LA Management) παρουσιάζει μία μέθοδο για τον υπολογισμό του βέλτιστου μεγέθους μίας LA, λαμβάνοντας υπ' όψιν το κόστος για τα Location Updates (LUs) και το πλήθος των paging μηνυμάτων που απαιτούνται. Προτείνεται πλεγματοειδής διάρθρωση, με τετραγωνικές κυψέλες από $k \times k$ κυψέλες για κάθε LA. Η τιμή του k επιλέγεται ανά χρήστη, σύμφωνα με την κινητικότητά του, το μοτίβο των εισερχόμενων κλήσεων και τις παραμέτρους κόστους. Ο μηχανισμός αυτός είναι πιο αποδοτικός από τον σημερινό, κατά τον οποίο το μέγεθος των LAs είναι σταθερό. Εντούτοις, ανακύπτει και ένα σημαντικό πρόβλημα, καθώς δεν είναι εύκολη η χρήση διαφορετικών μεγεθών LAs για το κάθε τερματικό, μιας και τα τελευταία θα δυσκολεύονται να αναγνωρίζουν τα όρια των LAs τα οποία θα αλλάζουν συνεχώς. Η πραγματοποίηση αυτής της μεθόδου γίνεται ακόμα πιο περίπλοκη, αν οι κυψέλες είναι εξαγωνικές ή, ακόμα χειρότερα, αν είναι ακανόνιστες.

Μία άλλη μέθοδος που έχει προταθεί για την Ενημέρωση Θέσης Χρήστη, ως λύση για το παραπάνω πρόβλημα, είναι μία *time based scheme* κατά την οποία το τερματικό εκτελεί LUs περιοδικά σε χρόνο ΔT . Εάν για παράδειγμα ένα τερματικό βρίσκεται στην περιοχή A σε χρόνο 0, τότε θα ακολουθήσουν ενημερώσεις στις περιοχές B, C και D, εφ' όσον το τερματικό μετακινηθεί εκεί στους χρόνους ΔT , $2\Delta T$ και $3\Delta T$ αντίστοιχα (Εικόνα 6, μπλε γραμμή).



Εικόνα 6. Time και Movement Based Schemes Διαχείρισης Θέσης

Σε μία *movement-based scheme*, το τερματικό πραγματοποιεί μία ενημέρωση θέσης, κάθε φορά που ολοκληρώνει έναν προκαθορισμένο αριθμό κινήσεων δια μέσου των ορίων των κυψελών. Αν υποθέσουμε ότι το κατώφλι κινήσεων έχει την τιμή 3, τότε το τερματικό (Εικόνα 6, μωβ γραμμή) θα κάνει ένα LU στην περιοχή E και δύο LUs στην περιοχή F, έχοντας περάσει από 3 όρια για να φτάσει σε κάθε μία από αυτές.

Στην *distance-based scheme*, το τερματικό ενημερώνει τη θέση του κάθε φορά που η απόστασή του από την κυψέλη στην οποία έκανε την τελευταία ενημέρωση είναι μεγαλύτερη από μία προκαθορισμένη τιμή (κατώφλι απόστασης).

Κατά την επαναληπτική, βασισμένη στην απόσταση, ενημέρωση θέσης του τερματικού (*iterative distance – based location update scheme*), χρησιμοποιείται ένας επαναληπτικός αλγόριθμος ο οποίος επιστρέφει το βέλτιστο κατώφλι απόστασης το οποίο συνεπάγεται το ελάχιστο κόστος. Όταν καταφθάνει μία εισερχόμενη κλήση, δηλαδή, στέλνονται paging μηνύματα πρώτα στις κυψέλες που απέχουν τη μικρότερη απόσταση από την κυψέλη στην οποία έγινε η τελευταία ενημέρωση θέσης. Έτσι, η καθυστέρηση του εντοπισμού του τερματικού είναι ανάλογη με την απόσταση κατά την οποία αυτό μετακινήθηκε από την τελευταία LU του. Τα αποτελέσματα της εφαρμογής του μηχανισμού αυτού δείχνουν ότι η εξάρτηση από την κινητικότητα και τις παραμέτρους της άφιξης των κλήσεων συνεπάγεται μεταβολή του κατωφλίου απόστασης, αποδεικνύοντας έτσι ότι οι ενημερώσεις θέσης αυτού του τύπου πρέπει να είναι *user – based*, και να προσαρμόζονται δυναμικά σύμφωνα με την τρέχουσα κινητικότητα και το μοτίβο των εισερχομένων κλήσεων. Ένα πολύ σημαντικό μειονέκτημα που προκύπτει από την εφαρμογή αυτής της μεθόδου έχει να κάνει με τους πόρους του τερματικού. Επειδή ο αλγόριθμος που χρησιμοποιείται είναι επαναληπτικός, ο προσδιορισμός του βέλτιστου κατωφλίου απόστασης θα απαιτεί αρκετά σημαντική υπολογιστική ισχύ από το τερματικό.

2.5.3 Paging Τερματικών

Αρκετή έρευνα έχει γίνει και στην κατεύθυνση του Terminal Paging, καθώς οι σημερινές μέθοδοι Paging καταλήγουν σε μεγάλο κόστος και σημαντικές καθυστερήσεις. Οι C. Rose και R. Yates στην εργασία τους «Minimizing the average cost of paging under delay constraints», υποθέτουν ότι η περιοχή κάλυψης ενός δικτύου είναι χωρισμένη σε LAs, με γνωστή την πιθανότητα ένα τερματικό να βρίσκεται σε δεδομένη LA. Έστω P_{LAx} ($x = 1, 2, \dots$), η πιθανότητα το τερματικό να βρίσκεται στην περιοχή LAx. Η έρευνα έδειξε ότι αν δεν τίθεται περιορισμός στην Paging καθυστέρηση, και αν η αναζήτηση στις LAs γίνεται σειριακά ξεκινώντας από αυτήν με τη μεγαλύτερη P_{LA} και προχωρώντας με τη σειρά σε περιοχές με μικρότερη πιθανότητα, τότε ελαχιστοποιείται το κόστος. Εάν, δε, τίθεται περιορισμός στην καθυστέρηση, οι συγγραφείς προτείνουν μία βέλτιστη ακολουθία αναζήτησης η οποία καταλήγει στο ελάχιστο κόστος. Κατά την έρευνα, όμως, έγινε η παραδοχή ότι είναι γνωστή η πιθανότητα P_{LA} . Πρέπει, λοιπόν, να ληφθεί υπ' όψιν το γεγονός ότι η πιθανότητα αυτή είναι άμεσα εξαρτημένη από το χρήστη, με αποτέλεσμα ο αλγόριθμος τον οποίο πρότειναν οι Rose και Yates να περιλαμβάνει τις κατανομές πιθανοτήτων που προκύπτουν από την κινητικότητα του χρήστη.

Ένα άλλο Paging μοντέλο που προτάθηκε βασίζεται πάλι στην απόσταση, μόνο που εδώ η Paging καθυστέρηση είναι περιορισμένη, με το χρόνο που απαιτείται για τον εντοπισμό ενός τερματικού να είναι μικρότερος ή ίσος από μία προκαθορισμένη μέγιστη τιμή. Όταν καταφθάνει μία εισερχόμενη κλήση, η περιοχή στην οποία βρίσκεται το τερματικό χωρίζεται σε κάποιο πλήθος υποπεριοχών, στις οποίες στέλνονται paging messages με σειριακό τρόπο. Μειώνοντας τον αριθμό των υποπεριοχών αυτών σε μία δοθείσα τιμή N , ο χρόνος που απαιτείται για τον εντοπισμό του τερματικού θα είναι μικρότερος ή ίσος από το χρόνο που απαιτείται για το paging N περιοχών. Με γνωστή την κινητικότητα, τις παραμέτρους άφιξης των κλήσεων, το κατώφλι απόστασης και τη μέγιστη καθυστέρηση, μπορεί να υπολογιστεί το αναμενόμενο κόστος της προτεινόμενης μεθόδου. Χρησιμοποιώντας τα παραπάνω ως δεδομένα εισόδου, μπορεί να χρησιμοποιηθεί ένας επαναληπτικός αλγόριθμος ο οποίος θα επιστρέφει το βέλτιστο κατώφλι απόστασης που θα καταλήγει στο μικρότερο κόστος. Αποδεικνύεται ότι το ελάχιστο κόστος προκύπτει όταν δεν τίθενται περιορισμοί στην καθυστέρηση. Παράλληλα, η αύξηση του κατωφλίου καθυστέρησης κατά μία χρονική μονάδα τη φορά, συνεπάγεται σημαντική μείωση του κόστους για τον εντοπισμό θέσης του τερματικού.

2.6 ΔΙΑΧΕΙΡΙΣΗ ΚΙΝΗΤΙΚΟΤΗΤΑΣ

2.6.1 Γενικά

Όπως έχει ήδη αναφερθεί, τα B3G ασύρματα δίκτυα θα ενοποιούν υπηρεσίες διαφορετικών τεχνολογιών ραδιοπρόσβασης, χρησιμοποιώντας αρκετά εναλλακτικά δίκτυα. Ασχέτως από το δίκτυο, όμως, ένα από τα πιο σημαντικά προβλήματα στις ασύρματες επικοινωνίες είναι η διαχείριση κινητικότητας (mobility management) η οποία δίνει τη δυνατότητα στα δίκτυα τηλεπικοινωνιών να εντοπίζουν περιηγόμενα τερματικά για να τους μεταφέρουν τις κλήσεις, και να διατηρούν τις συνδέσεις τους όταν αυτά μετακινούνται σε νέα περιοχή υπηρεσιών.

Τα handovers μπορούν να εκτελεστούν τόσο ανάμεσα σε διαφορετικές κυψέλες του ίδιου συστήματος όσο και σε διαφορετικά cells. Η διαδικασία αυτή γίνεται συνήθως σε δύο περιπτώσεις: όταν επιδεινωθεί η τρέχουσα ραδιοσύνδεση ή όταν το σύστημα αποφασίσει να ανακατανείμει τα ραδιοκανάλια προς αποφυγή συμφόρησης. Για παράδειγμα, ένας χρήστης μπορεί να περάσει τα σύνορα μίας κυψέλης και να μετακινηθεί στην γειτονική της, έχοντας ήδη τρέχουσα κλήση στο τερματικό του. Στην περίπτωση αυτή, η κλήση θα πρέπει να περάσει στη γειτονική κυψέλη, παρέχοντας στο χρήστη συνεχώς την επιθυμητή υπηρεσία. Εάν, όμως, οι γειτονικές κυψέλες δεν έχουν αρκετά κανάλια για να υποστηρίξουν το handoff, η κλήση αναγκαστικά θα γίνει blocked. Ειδικά σε συστήματα των οποίων τα μεγέθη των κυψελών είναι σχετικά μικρά (Microcellular systems), η διαδικασία του Handoff συνεπάγεται αρκετές επιπτώσεις στη συνολική απόδοση του συστήματος. Εδώ, ένα σημαντικό ζήτημα που προκύπτει, είναι η μείωση της πιθανότητας τερματισμού μιας τρέχουσας κλήσης, αφού από την πλευρά του χρήστη, είναι προτιμότερο να γίνει blocked μια νέα κλήση, παρά να τερματιστεί η τρέχουσα του. Συνεπώς, το σύστημα πρέπει να μειώσει τις πιθανότητες των ανεπιτυχών handoffs, μέσω της διατήρησης κάποιων καναλιών ειδικά για handoff κλήσεις, κατανέμοντας κανάλια για αιτήσεις μεταπομπής, για παράδειγμα, πιο γρήγορα από ότι για νέες κλήσεις.

2.6.2 Αρχιτεκτονική Δικτύου για Υποστήριξη Διαχείρισης Κινητικότητας

Η διαχείριση της κινητικότητας, λοιπόν, έχει να κάνει με την αρραγή υποστήριξη real-time καθώς και non-real-time υπηρεσιών εν κινήσει, με τέτοιο τρόπο ώστε να ελαχιστοποιείται η απώλεια πακέτων και η καθυστέρηση. Από τις πολλές έρευνες που συναντώνται στη βιβλιογραφία έχουν εντοπιστεί αρκετοί περιορισμοί όσον αφορά στις προτεινόμενες μεθόδους της mobility management (Kibria et al). Κάποιοι από τους περιορισμούς αυτούς έχουν ως αποτέλεσμα τη μείωση του επιπέδου ισχύος σήματος, τη μεγάλη απαίτηση σε ισχύ μπαταρίας τερματικού, την αύξηση του overhead σηματοδοσίας κατά το fast handover και άλλα. Έτσι από τους Kibria et al έχει προταθεί ένας αναβαθμισμένος μηχανισμός διαχείρισης κινητικότητας, ο οποίος βασίζεται στα μελετημένα και αποδεκτά πρότυπα όπως αυτά αναφέρονται στις έρευνες και τη βιβλιογραφία, τον οποίο θα δούμε αναλυτικά παρακάτω.

Για την διευκόλυνση της αρραγούς κινητικότητας διαμέσου ετερογενών δικτύων, προτείνεται μία ιεραρχική δομή όπως παρουσιάστηκε στην Εικόνα 5 της υποενότητας 3.4.2. Στο χαμηλότερο στρώμα της δομής, όπως ήδη είδαμε, βρίσκονται οι Base Stations (BSs) οι οποίοι ελέγχονται από τους Access Routers (ARs). Η δομή διαχείρισης κινητικότητας αυτή, χρησιμοποιεί ένα αποκλειστικό σύστημα ελέγχου/ σηματοδοσίας Βασικού Δικτύου Πρόσβασης (Basic Access Network, BAN), ανεξάρτητο από τα υπάρχοντα ετερογενή δίκτυα. Όλα τα επικαλυπτόμενα ετερογενή δίκτυα μέσα στο domain, προσπελαύνουν το διαδίκτυο μέσω του Mobility Anchor Point (MAP). Τέλος, η πύλη (Gateway, GW) συνδέει τα πολλαπλά MAPs ώστε να επιτευχθεί ταχύτερη συνδεσιμότητα και μεταφορά δεδομένων κατά τη διάρκεια των inter – domain handovers.

2.6.3 Μεταπομπή Περιηγόμενου Τερματικού

Ένα περιηγόμενο τερματικό αναγνωρίζεται από τρεις IP διευθύνσεις:

- Home address του τερματικού
- IP address του αντίστοιχου MAP
- Local address, η οποία ανατίθεται από τον αντίστοιχο AR (κάθε AR διατηρεί μια βάση δεδομένων με IPv6 διευθύνσεις)

Κατά την registration διαδικασία, κάθε κόμβος στη διαδρομή προς το MAP αποθηκεύει (1) την πληροφορία για την Local Address του τερματικού και (2) το ID της διεπαφής μέσω της οποίας λαμβάνονται τα πακέτα αίτησης καταχώρησης (registration request packet). Μετά από την αρχική καταχώρηση, κάθε ακόλουθο πακέτο δεδομένων από και προς το τερματικό ανανεώνει τη διαδρομή δρομολόγησης προς το MAP και μειώνει την ανάγκη για αποστολή paging μηνυμάτων στο τερματικό με το ιστορικό πρόσφατων δεδομένων, μέσα σε ορισμένο χρονικό διάστημα.

Το τερματικό, στο οποίο έχει αποδοθεί μία Local Address, «ακούει» σε περιοδικές ευρυεκπομπές του AR αναγνωριστικού μετά από τη διαδικασία καταχώρησης, ακόμα και εάν βρίσκεται στην κατάσταση Idle, πράγμα που του επιτρέπει να περιάγεται σε μεγαλύτερες περιοχές πρόσβασης χωρίς να απαιτείται η εκ νέου registration του. Επειδή η Local Address, εξ' ορισμού, είναι έγκυρη μόνο σε συγκεκριμένη περιοχή κάλυψης του serving AR, αν το τερματικό μετακινηθεί στην περιοχή ενός άλλου Access Router, θα πρέπει να του αποδοθεί νέα, αντίστοιχη τοπική διεύθυνση.

Κατά το inter – access router handover, την ώρα που ο παλιός AR στέλνει τις διάφορες παραμέτρους (παραμέτροι ποιότητας τρέχουσας υπηρεσίας, ρυθμός κωδικοποίησης βίντεο και άλλα) στο νέο, το MAP εκπέμπει πακέτα δεδομένων και στα δύο αυτά ARs, κάνοντας έτσι την αλλαγή της τοπικής διεύθυνσης και την διαδικασία της μεταπομπής διαφανή για το δίκτυο κορμού και τον απομακρυσμένο Home Agent. Κατά τη διάρκεια του inter – domain handover από την άλλη, ο GW στέλνει πακέτα δεδομένων ταυτόχρονα στο παλιό και στο νέο MAP. Έτσι, επιτρέπει την ταχεία σύνδεση ανάμεσα στα domains και αποκρύπτει τα συχνά handovers από τους χρήστες που κινούνται σε γρήγορα οχήματα.

Στην προταθείσα αρχιτεκτονική διαχείρισης της κινητικότητας, τρεις τύποι handovers είναι πιθανοί: Inter – BS, inter – AR και inter – MAP, με τα δύο τελευταία να είναι τα πιο δύσκολα. Κατά τη διαδικασία του inter – AR handover, πάνω από το ίδιο domain ακολουθούνται τα παρακάτω βήματα:

- Στέλνεται μέσω πολυεκπομπής η λίστα των διαθέσιμων RATs
- Το τερματικό αποστέλλει αίτηση για μεταπομπή
- Παράγονται οι τοπικές διευθύνσεις από τον υπηρετούντα AR
- Γίνονται διαπραγματεύσεις πόρων με το νέο AR μέσω του MAP
- Το MAP στέλνει, με bi – casting, πακέτα δεδομένων στον παλιό και το νέο AR και τελικά
- Το τερματικό καταχωρείται στο νέο AR ώστε να επαναεδραιωθεί η τρέχουσα σύνοδος επικοινωνίας και
- Το MAP στέλνει μία αίτηση στον παλιό AR ώστε ο τελευταίος να αποδεσμεύσει τη σύνδεση.

Το inter-MAP handover ακολουθεί την ίδια διαδικασία, με τη διαφορά ότι ο GW είναι αυτός που εκτελεί (1) τη διαπραγμάτευση πόρων, (2) το bi-casting πακέτων δεδομένων και στα δύο domains και (3) την αποστολή αίτησης για αποδέσμευση σύνδεσης στον παλιό MAP.

2.6.4 Μεταπομπή Ενεργοποιημένη από το QoS

Οι QoS management μέθοδοι που προτάθηκαν από αρκετούς ερευνητές, είτε οδηγούν σε μεγάλη σπατάλη πόρων λόγω της παροχής μεγάλου bandwidth, είτε διαφοροποιούν άνισα τις real-time και non-real-time υπηρεσίες. Για παράδειγμα, σε ένα από τα προταθέντα QoS management schemes, το τερματικό θα πρέπει να δεσμεύει το απαιτούμενο εύρος ζώνης πριν τη διαδικασία μεταπομπής, δίνοντας μεγαλύτερη προτεραιότητα στις υπηρεσίες πραγματικού χρόνου έναντι αυτών του μη-πραγματικού. Εάν δεν υπάρχουν αρκετοί διαθέσιμοι πόροι, τότε το μη-πραγματικού χρόνου φορτίο θα επιβραδυνθεί αρκετά σε σχέση με το real-time. Σε άλλες έρευνες, τα Πρωτόκολλα Δέσμευσης Πόρων που προτείνονται, δεν είναι κλιμακούμενα, και σπαταλούν τους διαθέσιμους πόρους με το να δεσμεύουν bandwidth σε αρκετούς, διαφορετικούς σταθμούς βάσης παρ' όλο που το τερματικό μπορεί να χρησιμοποιεί μόνο έναν από αυτούς.

Για την αποφυγή των παραπάνω περιορισμών και προβλημάτων, μπορούν να χρησιμοποιηθούν Umbrella Cells, για μεταπομπές ενεργοποιημένες από το QoS. Αντί, λοιπόν, το τερματικό να πρέπει πρώτα να δεσμεύσει πόρους, θα μπορεί να εγκαταστήσει μία σύνδεση με μία Umbrella Cell υψηλού όγκου πόρων και να τη χρησιμοποιεί είτε μέχρι να βρει ένα σταθμό βάσης με επαρκείς πόρους είτε μέχρι να λήξει η προθεσμία προκαθορισμένης τιμής ενός χρονομετρητή. Όταν λήξει η τιμή του handoff timer και αν ο νέος BS δεν έχει επαρκείς πόρους, ακολουθούν δύο περιπτώσεις: (1) ο νέος BS μπορεί να αρνηθεί την υπηρεσία που ζητάει ο τερματικό ή (2) ο νέος BS μπορεί να μειώσει τους πόρους για την παροχή υπηρεσιών σε ένα άλλο τερματικό, ώστε να τους διαθέσει στο νέο, το οποίο αιτείται μεταπομπής.

Εναλλακτικά, κατά τη διάρκεια της μεταπομπής, μπορεί να χρησιμοποιηθεί ένας μηχανισμός γνωστός ως Alternative Best – Effort (ABE), ο οποίος διαχωρίζει τις best effort κλάσεις υπηρεσιών σε πολλαπλές «διαφορετικές αλλά ίσες» («different but equal») best effort κλάσεις, οι οποίες διαφοροποιούνται ανάλογα με την καθυστέρηση ή την απώλεια πακέτων που συνεπάγεται η κάθε μία. Σε περιπτώσεις που ζητείται από ένα σχετικά

φορτωμένο δίκτυο να υποστηρίξει εφαρμογές με καθαρά πραγματικού χρόνου περιορισμούς (π.χ. διαδραστικό audio), η ABE υπηρεσία θα επιτρέψει στις εφαρμογές να επιλέξουν ανάμεσα στην από άκρο σε άκρο καθυστέρηση και στην αύξηση του συνολικού throughput. Το τερματικό, λοιπόν, μπορεί εναλλακτικά να χρησιμοποιήσει την ABE υπηρεσία κατά την περίοδο μεταπομπής, μέχρι να βρεθούν και να δεσμευτούν πόροι στη νέα κυψέλη. Αυτό σημαίνει ότι όταν το τερματικό εκτελεί τον handover αλγόριθμο, θα θεωρήσει ότι όλες τις υπηρεσίες ως best effort, και κατόπιν θα χρησιμοποιήσει τον ABE αλγόριθμο για να επιλέξει ανάμεσά τους ανάλογα με το αν επιθυμεί μικρή καθυστέρηση ή χαμηλό throughput, βασισμένο στις απαιτήσεις υπηρεσίας και το προφίλ του.

ΔΙΑΧΕΙΡΙΣΗ ΤΕΡΜΑΤΙΚΩΝ ΣΕ ΠΕΡΙΒΑΛΛΟΝΤΑ ΔΙΚΤΥΩΝ ΥΨΗΛΩΝ ΤΑΧΥΤΗΤΩΝ B3G

3.1 RECONFIGURABILITY

Η εξέλιξη των ασύρματων επικοινωνιών τα τελευταία χρόνια, οδήγησε σε μία σαφή τάση προς τα συστήματα Πέραν της Τρίτης Γενιάς (B3G), τα οποία υποστηρίζουν την ολοκλήρωση και την συνύπαρξη πολλαπλών, διαφορετικών Τεχνολογιών Ραδιοπρόσβασης σε ένα κοινό σύνθετο ραδιο – περιβάλλον, όπως είδαμε στο προηγούμενο κεφάλαιο. Στόχος των συστημάτων αυτών είναι η πλήρης εκμετάλλευση των δυνατοτήτων που εισάγουν οι ποικίλλες, διαθέσιμες τεχνολογίες πρόσβασης, οι οποίες μπορούν να συγκλίνουν σε ένα σύνθετο ασύρματο περιβάλλον όπου ο χρήστης είναι διαρκώς βέλτιστα συνδεδεμένος («ABC») και μπορεί να μετακινείται αρραγώς στις διάφορες τεχνολογίες.

Μέσω της συνεργασίας μεταξύ των RATs/ δικτύων, ο χρήστης θα μπορεί να κατευθύνεται στο πιο κατάλληλο από αυτά, ανάλογα με τις περιοχές υπηρεσίας, τις χρονικές ζώνες, τις απαιτήσεις των προφίλ χρήστη και την απόδοση των δικτύων. Συνεπώς, τα διάφορα συστήματα θα πρέπει να διαχειρίζονται καταλλήλως τους πόρους τους, ώστε να μπορούν να συνεργαστούν με τα άλλα δίκτυα της σύνθετης ραδιο – υποδομής, παρέχοντάς τους τη διαθέσιμη χωρητικότητά τους για την απορρόφηση του φορτίου τους, ώστε να γίνει αποτελεσματικά η αντιμετώπιση νέων συνθηκών (π.χ. αλλαγή στις απαιτήσεις φορτίου, hot – spot καταστάσεις και άλλα) και η διαχείριση υπηρεσιών.

Η εκμετάλλευση των B3G συστημάτων, λοιπόν, μπορεί να υποστηριχθεί από την έννοια της Επαναδιαμορφωτικότητας (Reconfigurability). Η εφαρμογή του Reconfigurability concept σε ένα περιβάλλον διαφόρων ασύρματων τεχνολογιών πρόσβασης, παρέχει βασικές τεχνολογίες που επιτρέπουν στα τερματικά και σε στοιχεία δικτύου να επιλέγουν δυναμικά (διαφανώς και ασφαλώς) και να προσαρμόζονται στο σύνολο των RATs που είναι πιο κατάλληλα για τις συνθήκες που συναντώνται στην εκάστοτε περιοχή ή χρονική ζώνη της ημέρας.

Αξίζει να σημειωθεί ότι σύμφωνα με το reconfigurability concept, η επιλογή των RATs δε θα εμποδίζεται από τις προεγκατεστημένες στα

τερματικά τεχνολογίες. Αντίθετα, τα διάφορα απαιτούμενα προγράμματα θα μπορούν να μεταφορτώνονται, να εγκαθίστανται και να ελέγχονται δυναμικά.

Για να γίνει δυνατή η εφαρμογή της Reconfigurability στα B3G συστήματα, απαιτούνται εξελιγμένες λειτουργίες διαχείρισης τερματικών, οι οποίες θα τα κάνουν ικανά να προσαρμόζονται στα πιο κατάλληλα δίκτυα. Απαιτείται, δηλαδή, ένα σύστημα διαχείρισης τερματικών το οποίο να παρέχει τις απαιτούμενες λειτουργίες για την εκμετάλλευση του σύνθετου ραδιο – περιβάλλοντος. Το σύστημα αυτό, από την πλευρά του χρήστη/ τερματικού, θα πρέπει να εστιάζει στον προσδιορισμό των δικτύων που παρέχουν οικονομικά τα καλύτερα επίπεδα QoS για τις εκάστοτε εφαρμογές. Από τη σκοπιά της B3G τεχνολογίας δε, το σύστημα θα πρέπει να εκμεταλλεύεται τις δυνατότητες της ραδιο – υποδομής, πράγμα που μπορεί να γίνει ενεργητικά ή παθητικά. Ενεργητικά, το σύστημα μπορεί να αντιδρά σε νέες συνθήκες της περιοχής υπηρεσίας, όπως σε ξαφνική εμφάνιση νέων hot spots. Παθητικά, το σύστημα διαχείρισης μπορεί απλά να αναμένει για αλλαγές του προτύπου απαιτήσεων. Μία άλλη απαίτηση που τίθεται στο διαχειριστικό σύστημα, είναι το να παίζει το ρόλο του διαμεσολαβητή κατά την κατανομή πόρων, ώστε να επιτρέπεται η συνεργασία των δικτύων στο πολυσύνθετο περιβάλλον. Τέλος, τα στοιχεία του συστήματος θα πρέπει να είναι ικανά στο να κατευθύνουν τους χρήστες στα πιο κατάλληλα δίκτυα.

Το κεφάλαιο αυτό πραγματεύεται τα μέρη και τις λειτουργίες που πρέπει να έχει ένα σύστημα διαχείρισης τερματικών, ώστε να μπορεί να γίνει αποτελεσματικά η εκμετάλλευση των διαφορετικών τεχνολογιών ραδιοπρόσβασης του σύνθετου περιβάλλοντος των B3G δικτύων. Θα δοθούν τα χαρακτηριστικά που πρέπει να έχουν τα ικανά για την υποστήριξη του Reconfiguration concept ευφυή τερματικά, και θα αναλυθούν οι συνιστώσες Παραμέτρων και Πολιτικών, Παρακολούθησης και Ανίχνευσης και Διαπραγμάτευσης και Επιλογής από τεχνικής και γνωσιακής άποψης.

3.2 ΕΥΦΥΗ ΑΣΥΡΜΑΤΑ ΤΕΡΜΑΤΙΚΑ

Προκειμένου να μπορούν τα ασύρματα τερματικά να λειτουργήσουν σε ένα ευφυές επαναδιαμορφώσιμο περιβάλλον πολλαπλών τεχνολογιών ραδιοπρόσβασης και να εκμεταλλευτούν τις ποικίλλες δυνατότητες που αυτό προσφέρει, απαιτείται η επέκτασή τους με αρκετά θεμελιώδη χαρακτηριστικά.

Κατ' αρχάς, θα πρέπει να είναι δυνατή η διαχείριση των προτιμήσεων του χρήστη καθώς και των δυνατοτήτων συσκευής, όπως επίσης και των πολιτικών των δικτύων ραδιοπρόσβασης. Απαιτείται, δηλαδή, η ακριβής περιγραφή και παρουσίαση της πληροφορίας, καθώς και η δυνατότητα επαναδιαμόρφωσης των αντίστοιχων πολιτικών και profiles.

Παράλληλα, είναι ουσιώδης η δυνατότητα απόκτησης γενικών πληροφοριών, όπως η ανίχνευση διαθέσιμων δικτύων ραδιοπρόσβασης και η παρακολούθηση του εκάστοτε περιβάλλοντος. Η ανίχνευση έχει να κάνει κυρίως με την περιοδική εκτέλεση ελέγχων, ούτως ώστε να μπορεί να προσδιοριστεί το αν μία ή περισσότερες νέες εναλλακτικές έγιναν διαθέσιμες. Πιο συγκεκριμένα, θα πρέπει να είναι εφικτή η αναγνώριση πιθανών εισόδων σε ένα νέο RAT στην περιοχή υπηρεσίας το οποίο θα μπορούσε να προσφέρει καλύτερες επιλογές (όπως υψηλότερα επίπεδα QoS, μικρότερο κόστος ανά QoS level/ υπηρεσία). Η παρακολούθηση, από την άλλη, αναφέρεται στη συγκέντρωση στατιστικών στοιχείων από διαφορετικά RATs, με σκοπό την καταγραφή της κατάστασής τους και την λήψη απόφασης για το αν απαιτείται επαναδιαμόρφωση ή όχι.

Πολύ σημαντικό χαρακτηριστικό που πρέπει να έχουν τα B3G τερματικά είναι και η δυνατότητα διαπραγμάτευσης των προσφορών με τα ποικίλα διαθέσιμα δίκτυα, και η επιλογή της καταλληλότερης επαναδιαμόρφωσης για τη συσκευή. Η διαδικασία επιλογής μπορεί να καταλήξει στην αλλαγή από ένα δίκτυο/ RAT σε ένα άλλο, συνεπώς η εκλογή του δικτύου πρέπει να είναι πάντα σύμφωνη με τις προτιμήσεις του χρήστη και τις δυνατότητες της συσκευής, λαμβάνοντας φυσικά υπ' όψιν τις διάφορες καταστάσεις της περιοχής υπηρεσίας και τις χρονικές ζώνες της ημέρας. Παράλληλα, η αλλαγή από ένα RAT σε άλλο πρέπει να γίνεται αρραγώς, χωρίς να διακόπτεται η επικοινωνία. Πρέπει να τονιστεί ότι η επιλογή των δικτύων δε θα πρέπει να περιορίζεται από τις προεγκατεστημένες στο τερματικό τεχνολογίες. Για το λόγο αυτό, πρέπει να παρέχεται η δυναμική μεταφόρτωση, εγκατάσταση και επικύρωση των λογισμικών μονάδων που απαιτείται για τη διαδικασία επαναδιαμόρφωσης και την υποστήριξη ενός ενδεχόμενου, επιπρόσθετου RAT.

Όσον αφορά την ασφάλεια, ένα σημαντικό σημείο στο οποίο πρέπει να δοθεί μεγαλύτερη βάση είναι η επαλήθευση και η εξακρίβωση της προέλευσης του προγραμματιστικού κώδικα που μεταφορτώθηκε, πριν την εγκατάστασή του στο τερματικό. Παράλληλα, θα πρέπει να παρέχονται μηχανισμοί ώστε να επιβλέπεται η λειτουργικότητα των μονάδων λογισμικού κατά τη διάρκεια

εκτέλεσής τους, ώστε να αποφευχθούν τυχόν μη εξουσιοδοτημένες ενέργειες από αυτά. Από την πλευρά της αξιοπιστίας, απαιτούνται κατάλληλες τεχνικές, μηχανισμοί και διαδικασίες, ώστε να διασφαλιστεί το ότι η εκτέλεση της επαναδιαμόρφωσης δε θα προκαλέσει κανένα πρόβλημα στην απόδοση του εκάστοτε συστήματος.

3.3 ΠΡΟΚΛΗΣΕΙΣ ΔΙΑΧΕΙΡΙΣΗΣ ΤΕΡΜΑΤΙΚΩΝ

Παρ' όλο που οι αρχιτεκτονικές διαχείρισης έχουν εξελιχθεί σημαντικά όσον αφορά τη διαχείριση των υποδομών των σταθερών δικτύων, η εμφάνιση της προσωπικής ασύρματης τεχνολογίας απαιτεί την επέκταση της διαχείρισής της προς νέες διαστάσεις, όπως είδαμε και στην ενότητα 3.1. Οι διαστάσεις αυτές δίνονται από διαφορετικούς παράγοντες, και εισάγουν πολυάριθμες προκλήσεις που πρέπει να ληφθούν υπ' όψιν κατά το σχεδιασμό και την υλοποίηση ενός συστήματος διαχείρισης B3G ασύρματων τερματικών.

Κατ' αρχάς, η διαχείριση στον τομέα που εξετάζουμε, στοχεύει σε εξοπλισμούς χαμηλών πόρων που απαιτούν τη χρήση ελάχιστου λογισμικού. Η σύνδεση με αυτές τις συσκευές, προφανώς, θα γίνεται ασύρματα, μέσω αναξιόπιστων ζεύξεων, διαμέσου πολλών γεωγραφικών περιοχών. Παράλληλα, η περιαγωγή ανάμεσα σε διαφορετικά δίκτυα πρόσβασης θα απαιτεί την σύνδεση του UE με management agents μίας ή περισσότερων περιοχών. Τέτοιες περιπτώσεις συναντώνται σπάνια στην τρέχουσα τεχνολογία, όπου η σύνδεση του χρήστη με τον πάροχο/ δίκτυο γίνεται στο ίδιο domain.

Άλλος περιορισμός που πρέπει να ληφθεί υπ' όψιν είναι το γεγονός ότι οι διαχειριζόμενες οντότητες δε θα είναι πάντα συνδεδεμένες. Λόγω παραγόντων όπως η κάλυψη υπηρεσίας, η ισχύς της μπαταρίας του εξοπλισμού και η συμπεριφορά των χρηστών, οι ενέργειες της διαχείρισης θα πρέπει να εκτελούνται ασύγχρονα.

Η σύνδεση των δικτύων προς το τερματικό, όπως είδαμε, θα γίνεται πάνω από ετερογενείς ζεύξεις. Ενώ οι υπάρχοντες μηχανισμοί διαχείρισης προϋποθέτουν ένα SNMP πάνω από την IP επικοινωνία, το εύρος των ασύρματων πρωτοκόλλων (WAP, GPRS, HTML και SMS) απαιτεί ένα ευπροσάρμοστο πρότυπο που θα λειτουργεί αποτελεσματικά και διαφανώς πάνω από αυτές τις τεχνολογίες.

Τέλος, επειδή προβλέπεται ότι σε μια πρώτη φάση, οι περισσότερες νέες λειτουργίες θα είναι σχετικές με τις εφαρμογές, θα πρέπει να δοθεί περισσότερη σημασία στη διαχείριση των επιπέδων υπηρεσίας και εφαρμογής. Στην επόμενη ενότητα θα δούμε τις συστάσεις των συστημάτων διαχείρισης τερματικών, από γνωσιακής και τεχνικής πλευράς, καθώς και τον τρόπο με τον οποίο αυτές δίνουν τις λύσεις στους αναφερθέντες περιορισμούς.

3.4 ΠΑΡΑΜΕΤΡΟΙ ΚΑΙ ΠΟΛΙΤΙΚΕΣ

Όπως αναφέρθηκε και σε προηγούμενη ενότητα, ένα Ευφυές Επαναδιαμορφώσιμο Σύστημα Διαχείρισης Συσκευών (Cognitive Reconfigurable Equipments Management System, CREMS), αποτελείται από τρεις συνιστώσες: «Παράμετροι και Πολιτικές», «Παρακολούθηση και Ανίχνευση», «Διαπραγμάτευση και Επιλογή». Σε αυτήν, και στις επόμενες ενότητες που ακολουθούν, θα δούμε τις λειτουργίες και τις επακόλουθες τεχνικές προσεγγίσεις των συνιστωσών αυτών.

Η πρώτη από τις αναφερθείσες συνιστώσες ενός CREMS, συλλέγει, διατηρεί και παρέχει πληροφορίες για τις δυνατότητες της συσκευής, τη συμπεριφορά του χρήστη, τις προτιμήσεις, τις απαιτήσεις, τους περιορισμούς και τις πολιτικές των δικτύων.

3.4.1 Παράμετροι Συσκευής

Οι παράμετροι της συσκευής είναι η πληροφορία που σχετίζεται με τις δυνατότητές της και προσδιορίζει ένα σύνολο των ενδεχόμενων, επιτρεπτών επαναδιαμορφώσεων της. Πιο συγκεκριμένα, οι επαναδιαμορφώσεις αυτές εμπεριέχουν τα RATs στα οποία η συσκευή μπορεί να λειτουργήσει, καθώς και το ανάλογο φάσμα και την ισχύ εκπομπής που μπορούν να χρησιμοποιηθούν.

3.4.2 Προφίλ Χρήστη

Η πληροφορία που σχετίζεται με το προφίλ χρήστη προσδιορίζει το σύνολο των εφαρμογών που μπορούν να χρησιμοποιηθούν από το συγκεκριμένο χρήστη και το σύνολο των επιπέδων QoS που σχετίζονται με τη χρήση μίας εφαρμογής. Η σχετική με το χρήστη πληροφορία, προσδιορίζει παράλληλα το μέγεθος του φορτίου που παράγεται από τη χρήση της εκάστοτε εφαρμογής, σε συγκεκριμένα επίπεδα ποιότητας. Το φορτίο αυτό είναι ένας τρόπος έκφρασης της προτίμησης σε συγκεκριμένα επίπεδα QoS. Με άλλα λόγια, υψηλά φορτία για μια εφαρμογή υπονοούν αυξημένο ενδιαφέρον του χρήστη για το συγκεκριμένο επίπεδο ποιότητας όταν αυτός χρησιμοποιεί την αντίστοιχη εφαρμογή. Τέλος, το user profile μπορεί να περιλαμβάνει την πληροφορία για τη μέγιστη τιμή που είναι διατεθειμένος να πληρώσει ο χρήστης, προκειμένου να χρησιμοποιήσει τη συγκεκριμένη εφαρμογή στα συγκεκριμένα επίπεδα ποιότητας.

Η εισαγωγή στοιχείων στο προφίλ χρήστη γίνεται μέσω μίας Γραφικής Διεπιφάνειας (GUI), μέσω της οποίας ο χρήστης μπορεί να δώσει

διαφορετικές προτεραιότητες σε παραμέτρους, οι οποίες τελικά επηρεάζουν τη διαδικασία επιλογής δικτύου πρόσβασης ή επαναδιαμορφώσεων. Ο καθορισμός των προτεραιοτήτων ισοδυναμεί με την απόδοση τιμών στους συντελεστές w_q , w_o , w_t και w_c , οι οποίοι αντιστοιχούν στις παραμέτρους «ποιότητα (Quality)», «προτιμητέος πάροχος δικτύου (preferred network Operator)» και «προτιμητέος τύπος τεχνολογίας (preferred Technology type)» και «κόστος (Cost)».

Οι τιμές των παραμέτρων αυτών αναπαριστούν τα βάρη των μεταβλητών εισόδου που θα χρησιμοποιηθούν στους αλγόριθμους επιλογής επαναδιαμορφώσεων και δικτύων. Εάν ο χρήστης μια συγκεκριμένη στιγμή, για παράδειγμα, ορίσει ότι για την συγκεκριμένη εφαρμογή το πιο σημαντικό κριτήριό του είναι η ποιότητα, ακολουθούμενη από τον τύπο της τεχνολογίας, το κόστος και τέλος τον πάροχο του δικτύου, οι παράμετροι θα λάβουν τέτοιες τιμές ώστε να ικανοποιείται η σχέση: $w_q > w_t > w_c > w_o$.

3.4.3 Πολιτικές Δικτύων

Οι πολιτικές δικτύων καθορίζουν τους περιορισμούς του παρόχου, που πρέπει να ληφθούν υπ' όψιν κατά το χειρισμό των διαφόρων καταστάσεων. Αυτοί οι περιορισμοί προστίθενται σε εκείνους που προσδιορίζονται από τα profiles του χρήστη και της συσκευής. Όσον αφορά το χρήστη, οι πολιτικές μπορούν να περιορίσουν τα επίπεδα ποιότητας και εφαρμογών που επιτρέπεται αυτός να χρησιμοποιήσει. Από την πλευρά της συσκευής, οι πολιτικές μπορούν να επιτρέψουν μόνο ένα υποσύνολο δυνατοτήτων. Επομένως, καθορίζουν τα επιτρεπτά RATs, τις συχνότητες και τις ισχύεις εκπομπής που μπορούν να χρησιμοποιηθούν, όταν ένα επιτρεπτό RAT επιλέγεται. Οι πολιτικές δικτύων μπορούν επίσης να καθορίσουν τις τιμές αλγορίθμων και παραμέτρων που θα έπρεπε να χρησιμοποιηθούν κατά το χειρισμό. Τέλος, μπορούν, προεραϊκά, να καθορίσουν, για κάθε σχετική περίπτωση, ένα αντίστοιχο σύνολο των «καλύτερων» διαμορφώσεων που μπορούν να χρησιμοποιηθούν ως κανόνες ή απλές εισηγήσεις.

3.4.4 Γνωστικές Πτυχές

Η συνιστώσα Παραμέτρων και Πολιτικών ενός CREMS, λοιπόν, παράγει γνώση πάνω στη συμπεριφορά, τις προτιμήσεις, τις απαιτήσεις και τους περιορισμούς του χρήστη. Σημαντικές είναι οι παρακάτω πτυχές:

- Η συμπεριφορά κινητικότητας και κυκλοφορίας του χρήστη όταν μία εφαρμογή χρησιμοποιείται, σε συγκεκριμένο τόπο και χρόνο. Η πληροφορία αυτή δείχνει και τις εφαρμογές που προτιμώνται.
- Οι προτιμήσεις για συγκεκριμένα επίπεδα ποιότητας υπηρεσίας, όταν μία εφαρμογή χρησιμοποιείται. Οι προτιμήσεις αυτές εκφράζονται με το βαθμό χρησιμοποίησης που σχετίζεται με την κάθε εφαρμογή και το αντίστοιχο QoS level.

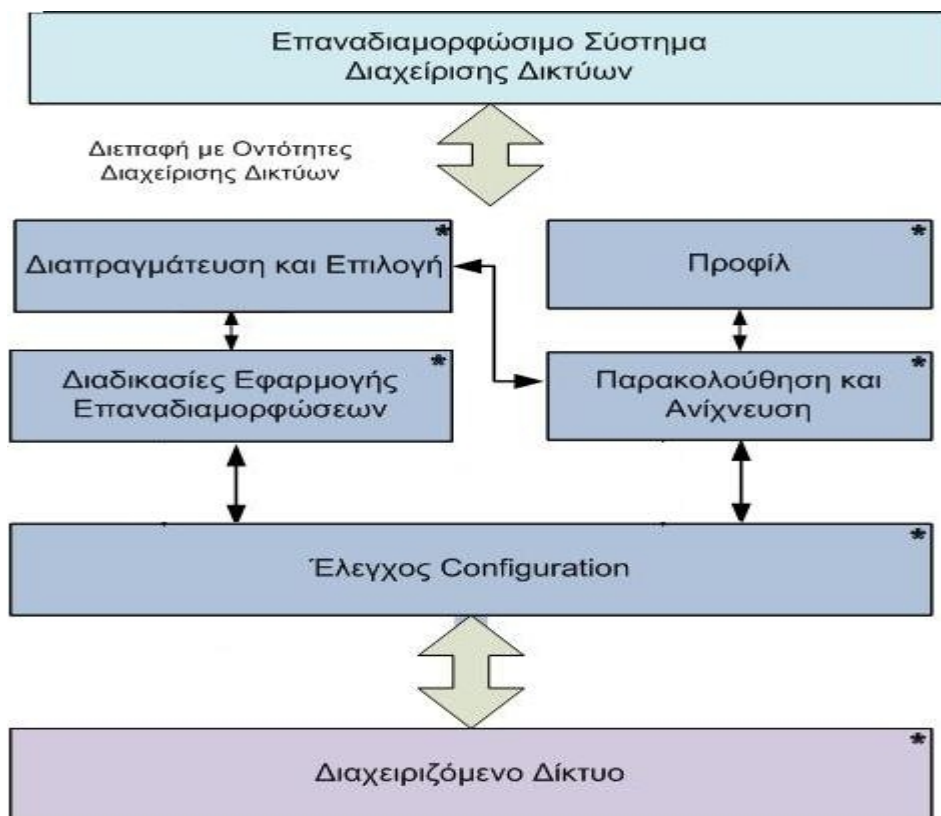
Πρέπει να αναφερθεί ότι η συμπεριφορά κυκλοφορίας ανά εφαρμογή έχει να κάνει με τη συχνότητα και τη διάρκεια της χρήσης της. Η συμπεριφορά κινητικότητας δε, καλύπτει πεδία όπως η ταχύτητα και η κατεύθυνση του τερματικού. Γενικά, η κυκλοφοριακή και κινητική συμπεριφορά έχουν εγγενή στοχαστική φύση. Επομένως, απαιτείται προηγούμενη γνώση για την αύξηση της αξιοπιστίας στην ακρίβεια της διαχείρισης των τερματικών. Επιπλέον, η έκφραση των προτιμήσεων του χρήστη για συγκεκριμένα επίπεδα ποιότητας μέσω των utility values είναι δύσκολη, καθώς η πλειοψηφία των χρηστών δεν είναι τεχνολογικά εμπειρογνώμονες ώστε να μπορούν να εκφράσουν απ' ευθείας τις ανάγκες τους για τα επίπεδα QoS. Σημαντικός περιορισμός, επίσης, είναι και το γεγονός ότι αυτές οι προτιμήσεις μπορούν να αλλάξουν ή να εξελιχθούν με την πάροδο του χρόνου, με αποτέλεσμα ο προσδιορισμός των επιπέδων QoS μέσω utility values να είναι ακόμα πιο δύσκολος.

3.4.5 Τεχνική Προσέγγιση

Το γνωσιακό μέρος της λειτουργίας του τερματικού βασίζεται σε τυχαίες μεταβλητές, υπολογισμένες ή αναφορικές τιμές, στην αλληλεπίδραση με το τμήμα Παρακολούθησης και Ανίχνευσης και στο χρήστη. Οι τυχαίες μεταβλητές συσχετίζονται με κάθε παράμετρο κυκλοφορίας και κινητικότητας (για κάθε εφαρμογή), καθώς και με την συσχετισμένη με κάθε εφαρμογή και επίπεδο QoS utility παράμετρο. Οι μετρήσεις προέρχονται από το τμήμα Παρακολούθησης και Ανίχνευσης, το οποίο θα δούμε παρακάτω, παρέχοντας τις μεταβλητές στις παραμέτρους κινητικότητας και κυκλοφορίας, καθώς και από το QoS της παροχής της εφαρμογής.

Οι μετρήσεις κυκλοφορίας και κινητικότητας μπορούν να χρησιμοποιηθούν για τη διαχείριση πιθανολογικών σχέσεων. Κάθε μία από αυτές τις σχέσεις μπορεί να εκφραστεί ως: «Η πιθανότητα για μια συγκεκριμένη παράμετρο (κυκλοφορίας ή κινητικότητας) να πάρει μια συγκεκριμένη (υπολογισμένη ή αναφορική) τιμή, σε ένα δοθέν πλαίσιο χρήσης (που περιλαμβάνει το χρήστη, την τοποθεσία, το χρόνο και την εφαρμογή που χρησιμοποιείται)».

Μια πιο σύνθετη προσέγγιση απαιτείται για τη διαχείριση των πιθανολογικών σχέσεων που αναφέρονται στις προτιμήσεις του χρήστη (utility values). Το τμήμα Παραμέτρων και Πολιτικών, λοιπόν, λαμβάνει από τη συνιστώσα Παρακολούθησης και Ανίχνευσης τα επίπεδα QoS στα οποία παρέχεται κάθε εφαρμογή, ενώ μπορεί να ζητηθεί και από το χρήστη να εκφράσει την ικανοποίησή του, μέσω μίας user – friendly, υψηλού επιπέδου κλίμακας βαθμολόγησης. Το αποτέλεσμα μπορεί να χρησιμοποιηθεί για να τεθούν σε λειτουργία εκείνες οι τιμές αναφοράς που βρίσκονται πιο κοντά στην υπολογισμένη τιμή. Η πιθανολογική σχέση εδώ είναι: «Η πιθανότητα για την utility, να πάρει μια τιμή αναφοράς, σε ένα δοθέν πλαίσιο του τελικού χρήστη (που καλύπτει το χρήστη, το ρόλο, την τοποθεσία, το χρόνο, την χρησιμοποιούμενη εφαρμογή και τα επίπεδα ποιότητας που προσφέρονται)».



Εικόνα 7. Αρχιτεκτονική Συστήματος Διαχείρισης Τερματικών

3.5 ΠΑΡΑΚΟΛΟΥΘΗΣΗ ΚΑΙ ΑΝΙΧΝΕΥΣΗ

Ο ρόλος της συνιστώσας Παρακολούθησης και Ανίχνευσης έχει να κάνει με την απόκτηση πληροφορίας για το πλαίσιο εφαρμογής, ώστε να μπορεί να γίνει αντιληπτή η τρέχουσα κατάσταση της συσκευής και του περιβάλλοντός της, κυρίως με την έννοια της αναγνώρισης των διαθέσιμων δικτύων σε μία συγκεκριμένη περιοχή. Παράλληλα, η συνιστώσα αυτή εκτελεί λειτουργίες ελέγχου της κατάστασης των δικτύων αυτών. Σε περίπτωση που παρατηρηθούν ασυμφωνίες, μπορεί να κριθεί απαραίτητη η επανεπιλογή των καταλληλότερων διαμορφώσεων.

3.5.1 Διαδικασία Παρακολούθησης

Ο έλεγχος εκτελείται ανά τακτές χρονικές περιόδους. Η διαδικασία ελέγχου αφορά στην συλλογή πληροφοριών της τρέχουσας σύνδεσης και της χρησιμοποιούμενης εφαρμογής. Μόλις η πληροφορία αυτή συλλεχθεί, οι τιμές των ελεγμένων παραμέτρων συγκρίνονται με ένα σύνολο προκαθορισμένων κατωφλίων. Σε περίπτωση που παρατηρηθούν παραβιάσεις και αν η επαναδιαμόρφωση καθιστεί απαραίτητη, ενεργοποιείται η διαδικασία ανίχνευσης, με σκοπό την αναγνώριση ενδεχόμενων εναλλακτικών διαμορφώσεων, δηλαδή των RATs που προφέρουν καλύτερες συνθήκες λειτουργίας.

3.5.2 Διαδικασία Ανίχνευσης

Η διαδικασία της ανίχνευσης μπορεί, επίσης, να ενεργοποιείται σε τακτά χρονικά διαστήματα ούτως ώστε η συσκευή να μπορεί να διατηρεί ενημερωμένες «οπτικές εικόνες» του περιβάλλοντος. Η συνιστώσα Παρακολούθησης και Ανίχνευσης βασίζεται στο σύνολο των εντοπισμένων διαμορφώσεων, στο υποσύνολο των διαμορφώσεων που συμφωνούν με τους ορισμένους από το χρήστη περιορισμούς και στα profiles της συσκευής. Για παράδειγμα, εάν οι προτιμήσεις του χρήστη (στο αντίστοιχο προφίλ) υποδεικνύουν ένα κάτω όριο στα 256 Kbps για μια βιντεοκλήση, θα απορριφθούν όλα τα RATs που προσφέρουν χαμηλότερα επίπεδα ποιότητας. Στην περίπτωση που ένα εναλλακτικό RAT μπορεί να χρησιμοποιηθεί, η

συνιστώσα Διαπραγμάτευσης και Επιλογής ενεργοποιείται για να επανεκτιμήσει την επιλεχθείσα τεχνολογία.

3.5.3 Γνωστικές Πτυχές

Η γνωστική λειτουργία της συνιστώσας Παρακολούθησης και Ανίχνευσης απαιτεί συγκεκριμένες δυνατότητες ώστε να εξάγει πρόσθετη πληροφορία για το τρέχον γενικό πλαίσιο εφαρμογής και προβλέψεις για μελλοντικές περιπτώσεις χρήσης. Η πρόσθετη πληροφορία στο τρέχον πλαίσιο εφαρμογής προσδίδει σταθερότητα και αξιοπιστία στις διαδικασίες παρακολούθησης και ανίχνευσης.

Η Παρακολούθηση, ουσιαστικά, απαντά στο ερώτημα: «Ποιά είναι τα πιο πιθανά επίπεδα QoS που επιτυγχάνονται συνήθως από μία δοθείσα διαμόρφωση, σε ένα δοθέν πλαίσιο εφαρμογής (φορτίο, κυκλοφορία, συνθήκες παρεμβολών)». Παράλληλα, η Ανίχνευση απαντά στο ερώτημα: «Ποιες είναι οι πιο πιθανές συνθήκες (π.χ. bit rate, κάλυψης μίας δοθείσας εναλλακτικής επαναδιαμόρφωσης, υποψήφιας για μελλοντική επιλογή) σε ένα συγκεκριμένο πλαίσιο εφαρμογής (τόπος, χρόνος) ». Όσον αφορά στις συνολικές προβλέψεις που πρέπει να γίνουν, η στοχαστικά πιθανολογική ερώτηση που πρέπει να απαντηθεί είναι: «Δοθείσης μιας συγκεκριμένης κατάστασης στην τρέχουσα χρονική ζώνη (φορτίου, κινητικότητας, παρεμβολών), ποιες είναι οι περιπτώσεις που είναι πιο πιθανό να συναντηθούν σε μεταγενέστερη χρονική ζώνη».

Οι ιδιότητες των επαναδιαμορφώσεων, πολύ πιθανόν να μην είναι συνεχείς σε ένα δυναμικό, reconfigurable περιβάλλον, μιας και βασίζονται στις οντότητες των δικτύων, στις συνθήκες παρεμβολών, στις λειτουργίες που ήδη χρησιμοποιούνται στο περιβάλλον και άλλα. Επομένως, η δυναμική Ανίχνευση θα πρέπει να παρέχει μεθόδους ώστε να μπορεί να διατηρεί τη γνώση πάνω στα συνήθη χαρακτηριστικά των εναλλακτικών υποψηφίων reconfigurations σε συγκεκριμένα πλαίσια εφαρμογής (τοποθεσίες, χρονικές ζώνες). Η δυναμική Παρακολούθηση έχει επίσης παρεμφερές κίνητρο.

Πρέπει να τονιστεί ότι γενικά, η εκτίμηση των επιπέδων QoS που προσφέρονται από την εκάστοτε διαμόρφωση είναι δύσκολη, και απαιτεί πιθανολογικές σχέσεις. Τέλος, απαραίτητες είναι και οι προβλέψεις του γενικού πλαισίου εφαρμογής για το δυναμικό χειρισμό των καταστάσεων, πράγμα που αποτελεί βασικό χαρακτηριστικό των ευφυών, γνωστικών συστημάτων.

3.5.4 Τεχνική Προσέγγιση

Το γνωστικό μέρος το τμήματος Παρακολούθησης και Ανίχνευσης ενός CREMS βασίζεται σε σύνολα τυχαίων μεταβλητών, τιμών αναφοράς και αποτελεσμάτων μετρήσεων.

Όσον αφορά τη δυναμική Ανίχνευση, οι τυχαίες μεταβλητές συσχετίζονται με τις εναλλακτικές διαμορφώσεις και συγκεκριμένα με κάθε ιδιότητα διαμόρφωσης (όπως bit rate, κάλυψη κ.ά.). Κάθε τυχαία μεταβλητή μπορεί να λάβει τιμή από ένα σύνολο διακριτών αναφορικών τιμών. Οι μετρήσεις, που δείχνουν τις πραγματικές τιμές των παραμέτρων μίας εναλλακτικής διαμόρφωσης, διεξάγονται από τη διαδικασία της Ανίχνευσης η οποία χρησιμοποιεί τις εξαγόμενες πληροφορίες για να διαχειριστεί τις πιθανολογικές σχέσεις. Κάθε μία από τις σχέσεις αυτές εκφράζεται ως « Η πιθανότητα μια ιδιότητα επαναδιαμόρφωσης (τυχαία μεταβλητή) να λάβει μια τιμή αναφοράς, για μια δοθείσα εναλλακτική επαναδιαμόρφωση, σε ένα δοθέν πλαίσιο (π.χ. τόπος, χρόνος) ». Εξ' ορισμού, με κάθε μέτρηση, θα αυξάνονται οι πιθανότητες των αναφορικών τιμών που είναι κοντά στην υπολογισθείσα τιμή, ενώ αυτές που απέχουν θα μειωθούν.

Όσον αφορά τη δυναμική Παρακολούθηση, από την άλλη, οι τυχαίες μεταβλητές μπορούν να συσχετιστούν με την επιλεγμένη επαναδιαμόρφωση, τη γενική κατάσταση που συναπαντάται (πληροφορία σχετική με το χρόνο, εφαρμογές που χρησιμοποιούνται, συνθήκες παρεμβολών) και κάθε παράμετρο QoS. Οι λειτουργίες της Παρακολούθησης, εκτός των άλλων, παρέχουν και τις μετρήσεις για τις αναφερθείσες πτυχές.

Προκειμένου να μειωθεί η πολυπλοκότητα και να αποφευχθεί η διατήρηση πληροφορίας για μεγάλο αριθμό περιπτώσεων, η συνιστώσα Παρακολούθησης και Ανίχνευσης μπορεί να λάβει υπ' όψιν αναφορικά πλαίσια εφαρμογής ώστε να αναγνωρίζει τις πιο πιθανές καταστάσεις και να ανανεώνει τις κατάλληλες πιθανοτικές πληροφορίες. Απαντάται, δηλαδή, το ερώτημα: «Ποια η πιθανότητα μια παράμετρος QoS (τυχαία μεταβλητή) να λάβει συγκεκριμένη τιμή (υπολογισμένη ή αναφορική), δοθείσης της διαμόρφωσης και των συνθηκών του γενικού πλαισίου».

Οι προβλέψεις του γενικού πλαισίου λειτουργίας βασίζονται σε λειτουργίες για κάθε χρονική ζώνη. Σε κάθε χρονική ζώνη, οι λειτουργίες Παρακολούθησης και Ανίχνευσης ελέγχουν τις περιπτώσεις που συναντώνται και αναγνωρίζουν και καταγράφουν την εμφάνιση των αντίστοιχων κοντινότερων περιπτώσεων αναφοράς. Με αυτή την έννοια, η συνιστώσα αυτή μπορεί να καταγράψει τις μεταβάσεις που λαμβάνουν χώρα ανάμεσα σε διάφορες περιπτώσεις αναφοράς, σε επιτυχείς χρονικές ζώνες. Η συχνότητα των μεταβάσεων, με αυτόν τον τρόπο, γίνεται ένας βασικός τρόπος υπολογισμού/ εκτίμησης των πιθανοτήτων και πρόβλεψης των μελλοντικών καταστάσεων και συνθηκών.

3.6 ΔΙΑΠΡΑΓΜΑΤΕΥΣΗ ΚΑΙ ΕΠΙΛΟΓΗ

Η συνιστώσα Διαπραγμάτευσης Επαναδιαμόρφωσης και Επιλογής αποφασίζει για την/ τις πιο κατάλληλη/ ες διαμόρφωση/ εις για την υπό διαχείριση συσκευή, από τη σκοπιά των διαθέσιμων QoS επιπέδων, λαμβάνοντας υπ' όψιν το τρέχον γενικό πλαίσιο εφαρμογής, τα προφίλ του χρήστη και της συσκευής, και τις πολιτικές των δικτύων. Πολύ χοντρικά, η λειτουργία της συνιστώσας αυτής περιγράφεται ως μία αντικειμενική συνάρτηση βελτιστοποίησης η οποία ως είσοδο λαμβάνει τους παραπάνω παράγοντες υπό μορφή σχετικών μεταβλητών απόφασης ή κατάλληλων περιορισμών.

Όπως είπαμε και προηγουμένως, οι πολιτικές των δικτύων εκφράζουν τους περιορισμούς που εισάγουν οι ISPs, ορίζοντας έτσι συγκεκριμένες καταστάσεις σε ένα σύνολο επαναδιαμορφώσεων. Με άλλα λόγια, θέτουν παραμέτρους όπως QoS levels και τιμές, δημιουργώντας απεικονίσεις διαφορετικών πλαισίων εφαρμογής. Η συνιστώσα Διαπραγμάτευσης και Επιλογής αντιμετωπίζει τη διαφορετικότητα των πλαισίων αυτών λαμβάνοντας τις κατάλληλες αποφάσεις:

- Εάν υπάρχουν περιορισμοί και κανόνες, τότε επιλέγεται εκείνη η επαναδιαμόρφωση (από το σύνολο των προτεινόμενων) η οποία είναι συμβατή με τις επιλογές του χρήστη και τις δυνατότητες της συσκευής.
- Εάν οι περιορισμοί δεν μπορούν να αντιμετωπιστούν, επιλέγεται μία επαναδιαμόρφωση έτσι ώστε να συμφωνεί με τους αλγόριθμους των Πολιτικών.

Η ύπαρξη πολλαπλών «βέλτιστων» επαναδιαμορφώσεων ανά περίπτωση συνεπάγεται αρκετά πλεονεκτήματα: Πρώτον, διευκολύνει την επίτευξη καλύτερων επιπέδων ποιότητας, αυξάνοντας έτσι την αξιοπιστία. Δεύτερον, καθιστά δυνατή την επίλυση συγκρούσεων, δηλαδή την επιλογή μίας νέας επαναδιαμόρφωσης σε περίπτωση που η χρήση της τρέχουσας εκτίθεται σε περιορισμούς. Τέλος, δίνει τη δυνατότητα στον εξοπλισμό να συμμετέχει ενεργά σε σχέδια διαπραγματεύσεων.

3.6.1 Γνωστικές Πτυχές

Από τη γνωστική σκοπιά, η συνιστώσα Διαπραγμάτευσης Επαναδιαμόρφωσης και Επιλογής παρέχει λειτουργικότητα για την εκτίμηση της πιο πιθανής κατάστασης, υπό συγκεκριμένη επαναδιαμόρφωση σε δοθέν πλαίσιο (φορτίο, κινητικότητα, παρεμβολές). Η λειτουργικότητα αυτή στηρίζεται σε αρκετούς τυχαίους παράγοντες όπως η τοποθεσία, τα επίπεδα εφαρμογής και ποιότητας, συνθήκες παρεμβολών κ.ά. Επομένως, δημιουργούνται πιθανολογικά μοντέλα προκειμένου να αυξηθεί η βεβαιότητα για την πιο πιθανή κατάσταση των επαναδιαμορφώσεων, σε συγκεκριμένα πλαίσια.

3.6.2 Τεχνική Προσέγγιση

Το γνωσιακό τμήμα της συνιστώσας Διαπραγμάτευσης Επαναδιαμόρφωσης και Επιλογής βασίζεται σε τυχαίες μεταβλητές, υπολογισμένες τιμές, τιμές αναφοράς και σε αλληλεπιδράσεις με τη συνιστώσα Παρακολούθησης και Ανίχνευσης. Οι τυχαίες μεταβλητές αντιστοιχίζονται στην επιλεγείσα επαναδιαμόρφωση, στο γενικό πλαίσιο εφαρμογής και στην τιμή της αντικειμενοστρεφούς συνάρτησης βελτιστοποίησης. Η υπό συζήτηση συνιστώσα, λοιπόν, παρέχει μετρήσεις και πληροφορίες για τον υπολογισμό της συνάρτησης αυτής, τις οποίες χρησιμοποιεί για να εξάγει την τιμή της. Με τον τρόπο αυτό, προσδιορίζει το πλησιέστερο πλαίσιο αναφοράς και διατηρεί τις πιθανολογικές σχέσεις. Κάθε μία από τις σχέσεις αυτές μπορεί να περιγραφεί ως «Η πιθανότητα η αντικειμενοστρεφής συνάρτηση να λάβει μια συγκεκριμένη τιμή, σε δεδομένη διαμόρφωση και δοθείσες συνθήκες (φορτίο, κινητικότητα, παρεμβολές).»

3.7 ΑΛΛΗΛΕΠΙΔΡΑΣΗ ΤΩΝ ΣΥΝΙΣΤΩΣΩΝ

Στην ενότητα αυτή θα δούμε συνοπτικά τη λειτουργία ενός ευφυούς συστήματος διαχείρισης τερματικών, και τις αντίστοιχες αλληλεπιδράσεις μεταξύ των συνιστωσών που εξετάσαμε προηγουμένως. Πιο αναλυτικά οι ακριβείς μέθοδοι και οι λειτουργίες των εν λόγω συνιστωσών, θα συζητηθούν στις παρακάτω ενότητες.

Στην πρώτη φάση, το CREMS αντιλαμβάνεται μία διέγερση η οποία μπορεί να προέρχεται από τη συνιστώσα Παρακολούθησης και Ανίχνευσης (δηλώνοντας ένα νέο πλαίσιο εφαρμογής) ή από το τμήμα Παραμέτρων και Πολιτικών (εάν ενεργοποιηθούν νέες πολιτικές από τον ISP).

Στη δεύτερη φάση, η διέγερση προωθείται στη συνιστώσα Διαπραγμάτευσης Επαναδιαμόρφωσης και Επιλογής, η οποία συνεργάζεται με τις άλλες δύο συνιστώσες για να αποκτήσει πληροφορίες σχετικές με τα προφίλ (χρήστη και συσκευής), τις πολιτικές των δικτύων και τις γενικές συνθήκες του περιβάλλοντος.

Κατά την τρίτη φάση λαμβάνονται οι αποφάσεις για τις βέλτιστες επαναδιαμορφώσεις, λαμβάνοντας υπ' όψιν τα αναφερθέντα προφίλ και τις δικτυακές πολιτικές. Στόχος είναι η επιλογή εκείνων των επαναδιαμορφώσεων που θα οδηγήσουν στην καλύτερη αντιμετώπιση της γενικής κατάστασης.

Στην τέταρτη φάση, η συνιστώσα Διαπραγμάτευσης και Επιλογής γνωστοποιεί τις αποφάσεις της, οι οποίες γίνονται δεκτές από τις υπόλοιπες συνιστώσες και εφαρμόζονται.

Στην τελική, Πέμπτη φάση, γίνεται ο χειρισμός του νέου πλαισίου εφαρμογής, δηλαδή της νέας κατάστασης μετά τη διέγερση. Κατά τη διάρκεια αυτής της φάσης όλες οι αναφερθείσες συνιστώσες συνεργάζονται για την βελτίωση της γνώσης και της εμπειρίας τους.

ΕΥΦΥΕΣ ΕΠΑΝΑΔΙΑΜΟΡΦΩΣΙΜΟ ΣΥΣΤΗΜΑ ΔΙΑΧΕΙΡΙΣΗΣ ΤΕΡΜΑΤΙΚΩΝ

4.1 ΕΙΣΑΓΩΓΗ

Στο προηγούμενο κεφάλαιο, εξετάσαμε σε θεωρητικό επίπεδο, τα μέρη ενός ευφυούς επαναδιαμορφώσιμου συστήματος διαχείρισης τερματικών, καθώς και τον τρόπο με τον οποίο αυτά αλληλεπιδρούν. Στο παρόν κεφάλαιο, θα δούμε τις ακριβείς λειτουργίες και τους αλγορίθμους ενός τέτοιου συστήματος, σε κάθε φάση λειτουργίας του. Θα δούμε, δηλαδή, τον τρόπο λειτουργίας του εν λόγω συστήματος στα εκάστοτε πλαίσια εφαρμογής, ξεκινώντας από τον προσδιορισμό της διέγερσης η οποία το θέτει σε ισχύ και καταλήγοντας στις λειτουργίες ελέγχου που εκτελούνται μετά τη μεταφόρτωση του απαιτούμενου, για την υποστήριξη της Reconfigurability, λογισμικού. Στο τέλος του κεφαλαίου, για καλύτερη κατανόηση των φάσεων λειτουργίας του συστήματος διαχείρισης τερματικών, παραθέτουμε και ένα παράδειγμα, στο οποίο φαίνονται με πιο απτό τρόπο, οι καταστάσεις και λειτουργίες του συστήματος διαχείρισης.

4.2 ΔΙΕΓΕΡΣΗ ΤΟΥ ΣΥΣΤΗΜΑΤΟΣ

Όπως είπαμε στο προηγούμενο κεφάλαιο, ο ρόλος της συνιστώσας Παρακολούθησης και Ανίχνευσης είναι ο προσδιορισμός διαθέσιμων δικτύων σε συγκεκριμένη περιοχή και η παρακολούθηση της κατάστασής τους. Η παρακολούθηση εκτελείται σε τακτά χρονικά διαστήματα και συνεπάγεται την «αφύπνιση» του τερματικού σε περίπτωση που αυτό είναι σε Power Save κατάσταση. Κατά τη διαδικασία αυτή, συλλέγεται πληροφορία για την τρέχουσα σύνδεση, δηλαδή μετρώνται παράμετροι όπως η ισχύς του λαμβανόμενου σήματος, το επίπεδο θορύβου, το bit rate, το bit error rate

κ.λπ. Η συλλεχθείσα πληροφορία συγκρίνεται με προσδιορισμένα κατώφλια (thresholds) και αν υφίσταται ασυμφωνία μεταξύ των υπολογισμένων τιμών και των τιμών κατωφλίων, τίθεται σε λειτουργία η Ανίχνευση. Γενικά, η διαδικασία της Ανίχνευσης εκτελείται σε συγκεκριμένα χρονικά διαστήματα, αλλά μπορεί να εκτελεστεί και όποτε εντοπιστεί πιθανή απαίτηση για διέγερση και εκκίνηση του συστήματος διαχείρισης.

Πιθανές αιτίες διέγερσης του συστήματος διαχείρισης είναι: η εμφάνιση μίας νέας αίτησης υπηρεσίας, η ενημέρωση του προφίλ χρήστη, η σοβαρή υποβάθμιση σήματος, ο εντοπισμός ενός νέου διαθέσιμου attachment point κ.λπ.

4.3 ΠΑΡΑΚΟΛΟΥΘΗΣΗ

Η παρακολούθηση εναλλακτικών τεχνολογιών ραδιοπρόσβασης μπορεί να γίνει μέσω πιλοτικού καναλιού. Το κανάλι αυτό μπορεί να λειτουργεί σε συγκεκριμένη συχνότητα, και μέσα από αυτό όλα τα RATs μίας συγκεκριμένης περιοχής μπορούν να γνωστοποιούν την ύπαρξή τους και να διαφημίζουν τις υπηρεσίες τους. Η πληροφορία που θα πρέπει να παρέχεται από τα RAT είναι το RAT ID, η συχνότητα στην οποία λειτουργούν, ο τύπος τους (π.χ. UMTS, WLAN και άλλα), καθώς και οι υπηρεσίες και τα επίπεδα ποιότητας που προσφέρουν.

Για την επιτυχή εκτέλεση της φάσης της Παρακολούθησης, απαιτείται μία Μονάδα Προσαρμογής Δικτυακών Διεπαφών (Network Intefrace Adaptation Module, NIAM), η οποία παρέχει στο σύστημα διαχείρισης τις μετρήσεις από τα διάφορα δίκτυα, δείχνοντας έτσι την ποιότητα του σήματος ή την κατάσταση της σύνδεσης για κάθε μία από τις διεπαφές. Παρέχει, δηλαδή, μία λίστα για κάθε σημείο σύνδεσης στο γειτονικό περιβάλλον, με την κάθε λίστα να περιλαμβάνει πληροφορίες για την ισχύ του σήματος στο Access Point, τη διαθεσιμότητα φάσματος, τον τύπο της χρησιμοποιούμενης τεχνολογίας και τον πάροχο του δικτύου.

4.4 ΑΝΙΧΝΕΥΣΗ

Όταν η συνιστώσα Παρακολούθησης και Ανίχνευσης ανιχνεύσει ένα νέο RAT, πρέπει να αποφασίσει για το αν θα το καταχωρήσει ως πιθανή

επιλογή ή θα το απορρίψει. Για να γίνει αυτό, η συνιστώσα αυτή λαμβάνει πρώτα πληροφορίες από τη συνιστώσα Προφίλ και συγκρίνει τα δεδομένα του υπό εξέταση RAT με αυτά που είναι καταχωρημένα ως «επιτρεπτά» στο τερματικό. Εάν οι υπηρεσίες ή η ποιότητα του RAT δε συμπίπτουν με τις προτιμήσεις του χρήστη, τότε αυτό θα απορριφθεί. Εάν δε, συμφωνεί με τις παραμέτρους, ενεργοποιείται η συνιστώσα Διαπραγμάτευσης και Επιλογής, για την επανεκτίμησή του.

4.5 ΔΙΑΠΡΑΓΜΑΤΕΥΣΗ

Για να καταστεί δυνατή η αυτόματη λειτουργία διαπραγμάτευσης στο σύστημα διαχείρισης και ελέγχου, μέσω της συνιστώσας Διαπραγμάτευσης και Επιλογής, πρέπει να ληφθούν υπ' όψιν δύο κύρια ζητήματα. Το πρώτο έχει να κάνει με το πρωτόκολλο διαπραγμάτευσης, το οποίο προσδιορίζει την πληροφορία που ανταλλάσσεται μεταξύ των αλληλοεπιδρώντων οντοτήτων και θέτει τους κανόνες της διαπραγμάτευσης αυτής. Το δεύτερο θέμα αναφέρεται στην στρατηγική της διαπραγμάτευσης, δηλαδή στο μοντέλο λήψης αποφάσεων των διαπραγματευομένων μερών. Παρακάτω, θα δούμε μια πρώτη προσέγγιση του πρωτοκόλλου διαπραγμάτευσης, μέσω λεπτομερούς περιγραφής της ροής πληροφορίας που ανταλλάσσεται ανάμεσα στο σύστημα διαχείρισης τερματικού και της πλευράς του δικτύου.

4.5.1 Έναρξη Επαναδιαμορφωτικών Διαδικασιών από το Τερματικό

Το τερματικό αποστέλλει μία αίτηση προσφοράς (offer request) στο δίκτυο (ή τα δίκτυα) που υπάρχουν διαθέσιμα στην τρέχουσα περιοχή υπηρεσίας, με σκοπό να του παράσχει πληροφορία η οποία θα επιτρέψει σε κάθε δίκτυο που θα λάβει την αίτηση, να υπολογίσει την αντίστοιχη απαίτηση δικτυακών πόρων. Η αίτηση προσφοράς πρέπει να περιέχει πληροφορία για την επιθυμητή υπηρεσία (ή υπηρεσίες) και για τα στοιχεία του προφίλ χρήστη. Οι πληροφορίες για τις υπηρεσίες και τα προφίλ θα χρησιμοποιηθούν για την εκτίμηση των πόρων που ζητώνται από το δίκτυο που λαμβάνει την αίτηση. Κάθε δίκτυο απαντά με ένα σύνολο προσφορών, προσδιορίζοντας τα επίπεδα QoS που μπορεί να προσφέρει και το αντίστοιχο κόστος. Το τερματικό λαμβάνει τις προσφορές των διαφόρων διαθέσιμων δικτύων και τις εκτιμά μέσω της διαδικασίας της επιλογής. Σε περίπτωση που μία προσφορά γίνει

δεκτή, αποστέλλεται ένα μήνυμα αποδοχής στο αντίστοιχο δίκτυο. Αλλιώς, αποστέλλεται μήνυμα το οποίο ενημερώνει το δίκτυο ότι η προσφορά του έχει απορριφθεί.

4.5.2 Έναρξη Επαναδιαμορφωτικών Διαδικασιών από το Δίκτυο

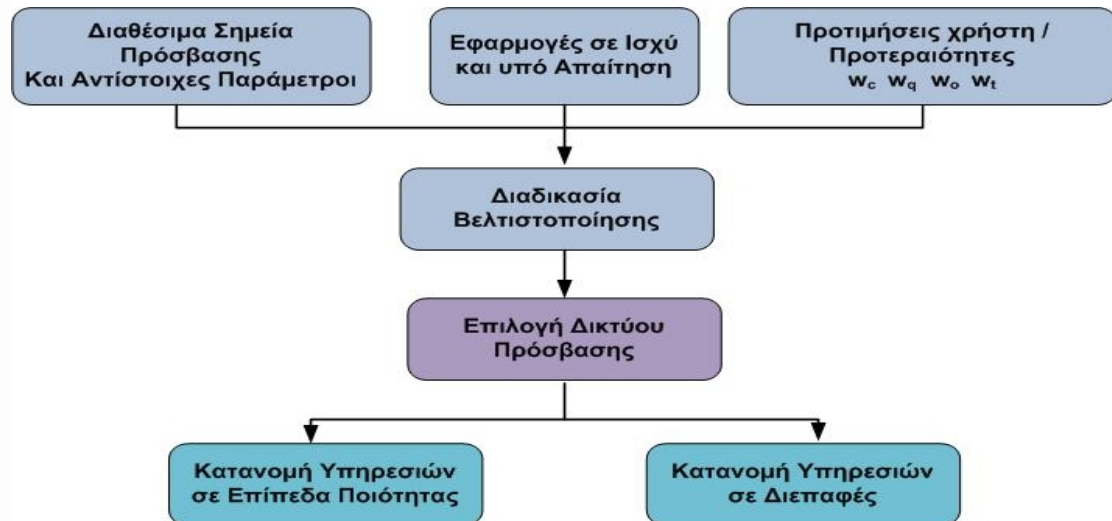
Η reconfiguration διαδικασία μπορεί, εναλλακτικά, να ενεργοποιηθεί και από το δίκτυο. Σε επείγουσες καταστάσεις, όπως για παράδειγμα κατά την συνολική υποβάθμιση δικτύου, το τελευταίο μπορεί να στείλει εντολή στο τερματικό να αλλάξει τις ρυθμίσεις του μέσω κατάλληλης διεπαφής. Σε τέτοιες περιπτώσεις, το τερματικό πληροφορεί το σύστημα διαχείρισης δικτύου για την τρέχουσα configuration του, δηλαδή τις ικανότητες της συσκευής και το σύνολο των διαθέσιμων RATs/ δικτύων που υποστηρίζει. Σε απάντηση, το σύστημα διαχείρισης δικτύου προσδιορίζει στο τερματικό σε ποιό δίκτυο θα πρέπει να συνδεθεί. Εναλλακτικά, το σύστημα διαχείρισης δικτύου μπορεί να παράσχει στο τερματικό μια λίστα από προτεινόμενα δίκτυα. Για κάθε ένα από αυτά, το τερματικό θα πρέπει να στείλει ένα μήνυμα προσφοράς, μετά την οποία ακολουθεί η ίδια διαδικασία όπως περιγράφηκε πιο πάνω.

4.6 ΕΠΙΛΟΓΗ

Η διαδικασία επιλογής έχει να κάνει με ένα βραχυπρόθεσμο πρόβλημα βελτιστοποίησης που στοχεύει στην ανάθεση του τερματικού σε συγκεκριμένο δίκτυο. Η λύση αυτού του προβλήματος βελτιστοποίησης επιτρέπει την επιλογή της κατάλληλης τεχνολογίας ραδιοπρόσβασης, για συγκεκριμένο χρήστη, μέσω της οποίας μπορεί να γίνει αποτελεσματική λήψη υπηρεσιών, όσον αφορά το κόστος και το QoS. Πιο συγκεκριμένα, η λειτουργία της επιλογής είναι υπεύθυνη για τη βέλτιστη επιλογή της τοπικής διεπαφής (τεχνολογίας) του κινητού τερματικού, καθώς και του σημείου σύνδεσης του δικτύου (Access Router, Access Point), βασισμένη στις απαιτήσεις της υπηρεσίας, στις προτιμήσεις του χρήστη και στην τρέχουσα διαθεσιμότητα του δικτύου.

Η διαδικασία της βελτιστοποίησης του προβλήματος, η οποία διενεργείται από την συνιστώσα Διαπραγματεύσεως και Επιλογής, έχει ως αποτέλεσμα την κατανομή των υπηρεσιών που ζητά ο χρήστης σε συγκεκριμένα επίπεδα ποιότητας, καθώς και στην ανάθεσή τους σε συγκεκριμένα δίκτυα. Ο υπολογισμός αυτών των κατανομών βελτιστοποιεί μία

αντικειμενική συνάρτηση, η οποία σχετίζεται με τα επίπεδα ποιότητας στα οποία παρέχεται η κάθε υπηρεσία, και το φορτίο που παράγεται από την ανάθεση των απαιτήσεων χρήστη σε υψηλά επίπεδα ποιότητας. Κατά την βελτιστοποίηση της συνάρτησης, λαμβάνονται υπ' όψιν οι περιορισμοί που έχουν να κάνουν με τις δυνατότητες του τερματικού ή το όριο του συνολικού χρηματικού ποσού που επιθυμεί να διαθέσει ο χρήστης.



Εικόνα 8. Διαδικασία Επιλογής Reconfigurations

4.6.1 Βελτιστοποίηση Αντικειμενικής Συνάρτησης Επιλογής

Ο αλγόριθμος επιλογής επαναδιαμόρφωσης βασίζεται στα παρακάτω δεδομένα εισόδου (E. Adamorouli et al):

- Ένα σύνολο μετρήσεων που αναπαριστούν τη διαθεσιμότητα, την ποιότητα του σήματος και άλλες παραμέτρους που παρατηρούνται σε κάθε διαθέσιμο attachment point, παρεχόμενες από τη NIAM.
- Το σύνολο των εφαρμογών που ήδη τρέχουν στο κινητό τερματικό, τα αντίστοιχα επίπεδα ποιότητας στα οποία οι εφαρμογές αυτές παρέχονται, καθώς και το σύνολο των εφαρμογών που ο χρήστης αιτεί να χρησιμοποιήσει
- Το σύνολο των προτιμήσεων χρήστη, σύμφωνα με τις οποίες καθορίζεται η προτεραιότητα των παραμέτρων «ποιότητας», «παρόχου δικτύου», «τύπου τεχνολογίας» και «κόστους».

Η έξοδος του αλγορίθμου δίνει τον τρόπο με τον οποίο πρέπει να κατανεμηθούν οι τρέχουσες και απαιτούμενες υπηρεσίες, στα κατάλληλα επίπεδα ποιότητας και στις αντίστοιχες δικτυακές διεπαφές. Πιο συγκεκριμένα,

οι εν λόγω κατανομές των υπηρεσιών θα πρέπει να υπολογίζονται ξεχωριστά για κάθε εκτελούμενη εφαρμογή, βελτιστοποιώντας την αντικειμενική συνάρτηση που σχετίζεται με τα βάρη που αποδίδονται στα διάφορα κριτήρια επιλογής.

4.6.1.1 Αντικειμενική Συνάρτηση Επιλογής

Έτσι, αν P είναι το σύνολο των attachment point που «βλέπει» το τερματικό, με

$$P = \{p_1, p_2, \dots, p_n\}, \quad n \in \mathbb{N}$$

και $Q(p)$ το σύνολο των επιπέδων ποιότητας στα οποία το attachment point p μπορεί να προσφέρει την υπό εξέταση υπηρεσία, με

$$Q(p) = \{q_1, q_2, \dots, q_m\}, \quad m \in \mathbb{N}$$

τότε ο στόχος είναι ο υπολογισμός της σχέσης (1):

$$\begin{aligned} OF(p,q) &= w_q \cdot \text{Quality}(p,q) &+ \\ &w_o \cdot \text{Operator}(p) &+ \\ &w_t \cdot \text{Technology}(p) &- \\ &w_c \cdot \text{Cost}(p,q) \end{aligned}$$

για όλα τα $p \in P$ και $q \in Q(p)$, καθώς και ο προσδιορισμός του:

$$\max \{ \max \{ OF(p, q) \} \} \quad (2),$$

ως το βέλτιστο attachment point και επίπεδο ποιότητας για κάθε μια από τις υπηρεσίες.

4.6.1.2 Παράγοντας Ποιότητας

Ο παράγοντας $\text{Quality}(p, q)$ της σχέσης (1) δεν είναι έκφραση της ονομαστικής ποιότητας που προσφέρεται από κάθε attachment point, αλλά έκφραση της συνδυασμένης επίδρασης του ονομαστικού επιπέδου ποιότητας και της παρατηρούμενης ισχύος σήματος από κάθε attachment point. Επομένως:

$$\text{Quality}(p, q) = q_s(p) \cdot q$$

όπου το $q_s(p)$ εκφράζει την ισχύ του λαμβανόμενου σήματος από το σημείο πρόσβασης p , το q εκφράζει το επίπεδο ποιότητας στο οποίο το attachment point p μπορεί να προσφέρει την υπό μελέτη υπηρεσία και ο συντελεστής $q_s(p)$ μπορεί να λάβει τιμές στο πεδίο $[0, 1]$.

4.6.1.3 Παράγοντας Κόστους

Ο συντελεστής $Cost(p, q)$ της σχέσης (1) αντιπροσωπεύει το κόστος μιας συγκεκριμένης απόφασης κατανομής, δηλαδή το κόστος στο οποίο το *attachement point* p μπορεί να προσφέρει την υπηρεσία σε επίπεδο ποιότητας q . Η πληροφορία για τα κόστη και για το ποιες υπηρεσίες προσφέρονται, λαμβάνεται από το δίκτυο είτε κάθε φορά που το κινητό τερματικό ενεργοποιείται, είτε ανά τακτά χρονικά διαστήματα (π.χ. μια φορά την ημέρα), και αποθηκεύεται στο τερματικό υπό μορφή XML εγγράφου. Το έγγραφο αυτό δε χρειάζεται να ανανεώνεται πολύ συχνά (π.χ. πάνω από μια φορά την ημέρα), αφού είναι απίθανο οι πάροχοι δικτύων να αλλάζουν τόσο συχνά τις τιμές τους. Τα δεδομένα αυτού του εγγράφου αντιστοιχούν στο κόστος παροχής μιας υπηρεσίας σε συγκεκριμένο επίπεδο ποιότητας, από συγκεκριμένο πάροχο και μέσω συγκεκριμένης τεχνολογίας, ανά μονάδα πληροφορίας (π.χ. Kb) ή χρόνου (π.χ. sec).

Οι τιμές κόστους που λαμβάνονται από το XML document κανονικοποιούνται από τον αλγόριθμο βελτιστοποίησης πριν χρησιμοποιηθούν στον υπολογισμό της αντικειμενικής συνάρτησης. Η διαδικασία κανονικοποίησης είναι απαραίτητη για τη μετατροπή των τιμών του συντελεστή $Cost(p, q)$ σε μια κλίμακα που τις καθιστά συγκρίσιμες με τις τιμές του $Quality(p, q)$. Αν $actualCost(p, q)$ είναι το κόστος ανά μονάδα υπηρεσίας, όπως λαμβάνεται από το XML document, $maxCost$ το μέγιστο κόστος ανά μονάδα για τη συγκεκριμένη αυτή υπηρεσία και $NoQoSLevels$ είναι ο αριθμός των διαφορετικών επιπέδων QoS στις οποίες μπορεί να τρέξει η υπηρεσία, τότε:

$$Cost(p, q) = (noQoSLevels) / (maxCost) \cdot (actualCost(p, q))$$

Τέλος, όσον αφορά τους συντελεστές $Operator(p)$ και $Technology(p)$ στην (1), ο χρήστης έχει τη δυνατότητα να καθορίσει τον προτιμητέο πάροχο δικτύου και τύπο τεχνολογίας, μέσω μιας γραφικής διεπαφής χρήστη. Σε περίπτωση που ένα υποψήφιο *attachement point* ανήκει στους προτιμητέους παρόχους, και υποστηρίζει την προτιμητέα τεχνολογία, του προστίθεται ένα «bonus». Η τιμή αυτού του bonus είναι ένα ποσοστό της απόλυτης τιμής του :

$$w_q \cdot Quality(p, q) - w_c \cdot Cost(p, q)$$

και σε περίπτωση που ο πάροχος προσφέρει την επιθυμητή τεχνολογία ισούται με:

$$Provider(p) = 0.5 \cdot Abs [w_q \cdot Quality(p, q) - w_c \cdot Cost(p, q)]$$

$$Technology(p) = 0.25 \cdot Abs [w_q \cdot Quality(p, q) - w_c \cdot Cost(p, q)]$$

(όπου $Abs = \text{«απόλυτη τιμή»}$).

Εάν η συνθήκη δεν ισχύει, δηλαδή αν δεν υπάρχει $Operator$ που να προσφέρει την τεχνολογία που ζητά το τερματικό, οι $Provider(p)$ και $Technology(p)$ λαμβάνουν την τιμή 0.

4.6.1.4 Έξοδος

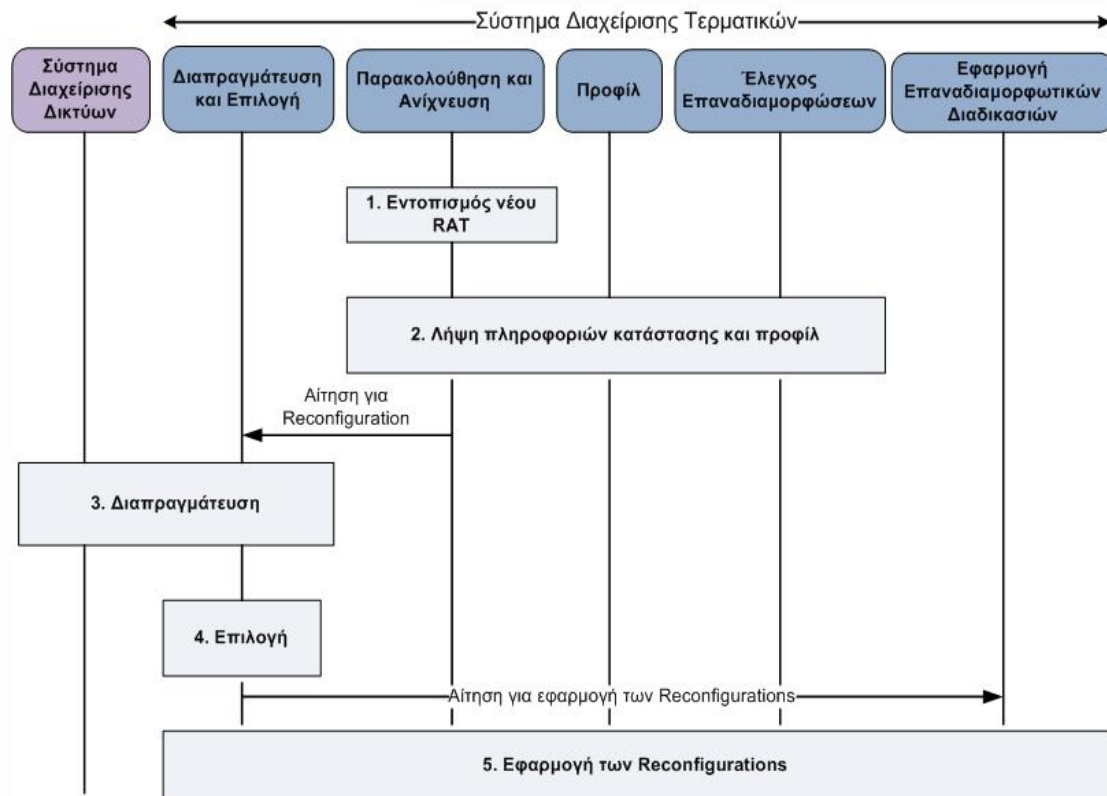
Οι παραπάνω συναρτήσεις υπολογίζονται έτσι ώστε να ικανοποιούνται οι σχέσεις (1) και (2). Σε περίπτωση που αυτό γίνει εφικτό, οι υπηρεσίες και εφαρμογές κατανέμονται και αποδίδονται στα κατάλληλα επίπεδα ποιότητας, στις αντίστοιχες τιμές και στα διαθέσιμα RATs.

4.7 ΕΦΑΡΜΟΓΗ ΕΠΑΝΑΔΙΑΜΟΡΦΩΤΙΚΩΝ ΔΙΑΔΙΚΑΣΙΩΝ

Κατα τη διαδικασία εφαρμογής των επαναδιαμορφωτικών διαδικασιών, ελέγχονται ως προς την αξιοπιστία και την ασφάλεια, οι διάφορες μονάδες λογισμικού που απαιτούνται για την πραγματοποίηση των Reconfigurations, πριν αυτές μεταφορτωθούν στο τερματικό. Στη συνέχεια, εκτελείται η διαδικασία επαναδιαμόρφωσης όπως αυτή αποφασίστηκε από τη συνιστώσα Διαπραγμάτευσης και Επιλογής, μέσω της μεταφόρτωσης, εγκατάστασης και επικύρωσης των software modules. Τέλος, εκτελούνται λειτουργίες εντοπισμού ελαττωμάτων και αποκατάστασης λαθών.

4.8 ΕΛΕΓΧΟΣ ΕΠΑΝΑΔΙΑΜΟΡΦΩΣΗΣ

Με την επιτυχή ολοκλήρωση των παραπάνω λειτουργιών, εκτελείται ο έλεγχος και η εποπτεία της εκτέλεσης της reconfiguration, μέσω διεπαφών στις οποίες το σύστημα διαχείρισης τερματικών επικοινωνεί με το επίπεδο εφαρμογής ελέγχοντας τις αντίστοιχες λειτουργίες του, και με το φυσικό επίπεδο ελέγχοντας τους διαθέσιμους πόρους. Επιπρόσθετα, εκτελούνται διαδικασίες επικοινωνίας και με τις άλλες στοίβες πρωτοκόλλου, όταν αυτό είναι απαραίτητο.



Εικόνα 9. Λειτουργία Ευφυούς Επαναδιαμορφώσιμου Συστήματος Διαχείρισης Τερματικών

4.9 ΔΙΑΧΕΙΡΙΣΗ ΤΕΡΜΑΤΙΚΩΝ (ΣΤΗΝ ΠΡΑΞΗ)

Στην ενότητα αυτή, θα δούμε πώς λειτουργεί ένα ευφύες σύστημα διαχείρισης τερματικών στην πράξη, εάν η απαίτηση για επιλογή reconfigurations προέλθει από την κινητικότητα του χρήστη και την αλλαγή των χρησιμοποιούμενων εφαρμογών. Η περίπτωση χρήσης και λειτουργίας που θα αναλυθεί, προέρχεται από τα αποτελέσματα προσωμοίωσης ενός συστήματος διαχείρισης, όπως αυτό προτάθηκε από τους E. Adamopoulou et al στην εργασία «Intelligent Access Network Selection in Heterogeneous Networks: Simulation Results».

4.9.1 Απόκριση του Συστήματος Διαχείρισης Τερματικών σε Αλλαγή Χρησιμοποιούμενων Εφαρμογών και Κινητικότητα

4.9.1.1 Πλαίσιο Εφαρμογής

Στην υποενότητα αυτή, θα δούμε πώς λειτουργεί ο αλγόριθμος επιλογής επαναδιαμορφώσεων, στην πράξη. Θα εξεταστούν τα βήματα που ακολουθούνται, σε ένα πλαίσιο εφαρμογής που προσομοιώνει μια συνηθισμένη μέρα της ζωής του επιχειρηματία κ. Green.

Ο κ. Green πηγαινοέρχεται από το σπίτι στο γραφείο του, χρησιμοποιώντας και το αμάξι του και το μετρό. Εκμεταλλεύεται τον διαθέσιμο χρόνο που έχει για να φτάσει στο γραφείο του, τρέχοντας αρκετές από τις υπηρεσίες του 4G τερματικού του. Οι υπηρεσίες προσφέρονται στα καταλληλότερα QoS επίπεδα μέσω συνδέσεων στα καλύτερα διαθέσιμα δίκτυα πρόσβασης (GSM, UMTS, WLAN, DVB).

Ο εν λόγω επιχειρηματίας, έχει επεξεργαστεί το user profile του βάζοντας τις ρυθμίσεις με την παρακάτω σειρά: Πρώτα, επέλεξε το «κόστος». Σαν δεύτερη σημαντικότερη παράμετρο επέλεξε την «ποιότητα». Τρίτο στη σειρά προτιμήσεων έβαλε τον «πάροχο δικτύου» και τελευταίο ως λιγότερη σημαντική από τις τέσσερις επιλογές, έβαλε τον «τύπο τεχνολογίας».

Επομένως, οι μεταβλητές της αντικειμενικής συνάρτησης επιλογής, λαμβάνουν τις τιμές: $w_c = 0.8$, $w_q = 0.6$, $w_o = 0.4$ και $w_t = 0.2$. Τα διαθέσιμα και τα μέγιστα επιτρεπτά επίπεδα ποιότητας για κάθε υπηρεσία του σεναρίου που εξετάζουμε, όπως τα όρισε ο κος Green, παρουσιάζονται στον παρακάτω πίνακα:

	SMS	Video - Κλήση	Video - Streaming	Πλοήγηση στο Διαδίκτυο
Συνολικός Αριθμός QoS Επιπέδων	1	5	5	5
Μέγιστο Επιτρεπτό Επίπεδο Ποιότητας	1	3	4	4

Πίνακας 2. Διαθέσιμα και Μέγιστα Επίπεδα QoS (του σεναρίου)

Ο πίνακας 3 παρουσιάζει μία λίστα με όλα τα διαθέσιμα attachment points του σεναρίου, σε κάθε περιοχή εφαρμογής που θα εξεταστεί. Με πράσινο, συμβολίζονται τα attachment points που έχουν ισχυρό σήμα, με κίτρινο συμβολίζεται η μεσαία ισχύς σήματος ενώ με κόκκινο τα χαμηλά επίπεδα ισχύος σήματος. Επίσης, να σημειωθεί ότι το UMTS#2 ανήκει στον πάροχο Operator#3.

Σπίτι	Πάρκινγκ	Αμάξι	Μετρό	Διαδρομή προς το Κτήριο	Γραφείο
GSM#1	GSM#1	GSM#2	GSM#3	GSM#4	GSM#4
UMTS#1	GSM#2	UMTS#2	WLAN#1	UMTS#3	UMTS#3
DBV#1	UMTS#1				WLAN#2
	UMTS#2				DVB#2
	DVB#1				

Πίνακας 3. Διαθέσιμα Attachment Points

4.9.1.2 Αποτελέσματα

Στις παρακάτω παραγράφους δίνεται μία σύντομη περιγραφή κάθε φάσης του σεναρίου που εξετάζουμε, καθώς και λεπτομερής εξήγηση των αντίστοιχων αποφάσεων κατανομής.

■ Μέσα Στο Σπίτι

Ο κος Green ξυπνάει το πρωί, ανοίγει το 4G τερματικό του και τρέχει μία video streaming εφαρμογή για να ενημερωθεί ζωντανά για τις τελευταίες εξελίξεις του χρηματιστηρίου. Επίσης, αποστέλλει ένα SMS στο συνεργάτη του, για να επιβεβαιώσει την ώρα του σημερινού meeting. Λίγο αργότερα, αποφασίζει ότι θα ήθελε να δει τις ειδήσεις για τις μετοχές σε καλύτερο επίπεδο ποιότητας, οπότε πάει στο προφίλ και προσδιορίζει την «ποιότητα» ως το πιο σημαντικό συντελεστή, και το «κόστος» ως το δεύτερο παράγοντα.

Για χάριν του παραδείγματος, υπολογίστηκε ο βαθμός της πρώτης κατανομής υπηρεσιών, χρησιμοποιώντας τη σχέση (1). Όπως είπαμε, πριν επεξεργαστεί ο κος Green το προφίλ του, οι σχέσεις των μεταβλητών ήταν: $w_c = 0.8$, $w_q = 0.6$, $w_o = 0.4$ και $w_t = 0.2$. Έστω, λοιπόν, ότι το attachment point p_1

= DVB#1 και $q_1 = 1$ (επίπεδο ποιότητας 1). Με τις τιμές αυτές, η σχέση (1) γίνεται:

$$OF(p_1, q_1) =$$

$$0.6 \cdot \text{Quality}(p_1, q_1) + 0.4 \cdot \text{Operator}(p_1) + 0.2 \cdot \text{Technology}(p_1) \\ - 0.8 \cdot \text{Cost}(p_1, q_1)$$

Όπου:

$$\text{Quality}(p_1, q_1) = q_s(p_1) \cdot q_1 = 1.0 \cdot 1 = 1$$

$$\text{Operator}(p_1) = 0 \text{ (δεν τέθηκαν τιμές παραμέτρων)}$$

$$\text{Technology}(p_1) = 0 \text{ (δεν τέθηκαν τιμές παραμέτρων)}$$

$$\text{Cost}(p_1, q_1) = (\text{noQoSLevels}) / (\text{maxCost}) \cdot (\text{actualCost}(p_1, q_1)) =$$

$$5 / 0.95 \cdot 0.0085 = 0.044737$$

Συνεπώς: $OF(p_1, q_1) = 0.56421$.

Με τον ίδιο τρόπο υπολογίζονται οι τιμές της αντικειμενικής συνάρτησης για τα attachment points $p_2 = \text{UMTS\#1}$ και $p_3 = \text{GSM\#1}$. Μετά τον υπολογισμό προκύπτει ότι $OF(p_1, q_1) > OF(p_i, q_i)$, $\forall i, j$ με $i \cdot j \neq 1$ και $i \leq 3$, $j \leq 4$. Έτσι, ότι ο αλγόριθμος επιλογής reconfiguration κατανέμει την video streaming υπηρεσία στο attachment point $p_1 = \text{DVB\#1}$ και στο επίπεδο ποιότητας $q_1 = 1$.

Οι παραπάνω υπολογισμοί, όπως είπαμε, προκύπτουν από τις επιλογές του user profile του κ. Green, πριν αυτός τις αλλάξει. Είδαμε, όμως, ότι μόλις ο επιχειρηματίας ξύπνησε, έτρεξε μία video streaming εφαρμογή. Το δίκτυο DVB προσφέρει χαμηλότερες τιμές από τα άλλα διαθέσιμα, οπότε προτιμάται από το GSM και το UMTS. Παράλληλα, ως επίπεδο ποιότητας επιλέγεται το «1», καθώς είναι το πιο φθινό (ο χρήστης επέλεξε το «κόστος» ως πιο σημαντικό παράγοντα). Όσον αφορά την Short Message Service, αυτό παρέχεται μέσω του GSM attachment point, καθώς αυτή είναι η πιο αποτελεσματική οικονομικά λύση.

Στο σημείο όπου ο χρήστης αποφάσισε να τρέξει την εφαρμογή σε καλύτερη ποιότητα και άλλαξε το προφίλ του θέτοντας την «ποιότητα» ως τον πιο σημαντικό παράγοντα, το τερματικό επιλέγει το DVB δίκτυο επειδή προσφέρει υψηλότερα Bitrates σε χαμηλή τιμή για video – related υπηρεσίες.

Το μέγιστο επίπεδο της ποιότητας, όπως φαίνεται από τον πίνακα 2, για αυτού του είδους τις υπηρεσίες είναι το «4».

■ Στο Χώρο Στάθμευσης

Ο κος Green αφήνει το σπίτι του και κατευθύνεται προς το χώρο στάθμευσης. Συνεχίζει να χρησιμοποιεί την υπηρεσία video streaming όσο προχωράει. Όσο πλησιάζει προς το parking, γίνονται διαθέσιμα νέα attachment points, ενώ το σήμα που λαμβάνεται από τα σημεία σύνδεσης που χρησιμοποιούνταν πριν, γίνεται ασθενέστερο.

Η εμφάνιση νέων διαθέσιμων attachment points οδηγεί στην επανεξέταση της τρέχουσας κατανομής της video streaming service. Όπως βλέπουμε και από τον πίνακα 3, το UMTS#2 παρέχει πολύ καλύτερη ισχύ σήματος από τα attachment points DVB#1, GSM#1 και UMTS#1. Παράλληλα, το GSM#2 έχει το μειονέκτημα ότι δεν μπορεί να προσφέρει QoS επίπεδα άνω του 3 για streaming υπηρεσίες. Έτσι, επιλέγεται τελικά το UMTS#2, σε επίπεδο ποιότητας 3 (μιας και δεν μπορεί να επιτευχθεί επίπεδο 4 μέσω ενός UMTS attachment point).

■ Στο Αμάξι

Ο κος Green φτάνει στο αμάξι του, τερματίζει την video streaming υπηρεσία και κατευθύνεται οδικά προς τον κοντινότερο σταθμό μετρό. Συναντά κυκλοφοριακή συμφόρηση και αποφασίζει να εκμεταλλευτεί το χρόνο του αλλάζοντας το προφίλ του, ορίζοντας τον Operator#3 ως τον πλέον προτιμητέο, και θέτει σε εφαρμογή μία υπηρεσία web browsing για να κατεβάσει ένα μεγάλο αρχείο που περιέχει μια αναφορά για την οικονομική κατάσταση της εταιρίας του.

Στο σημείο αυτό, λοιπόν, το τερματικό επιλέγει το UMTS#2, επειδή ανήκει στον πάροχο Operator #3 και σαν αποτέλεσμα ο αλγόριθμος επιλογής αποδίδει ένα «bonus» στον πάροχο αυτόν. Παράλληλα, επιλέγεται το 4^ο επίπεδο ποιότητας καθώς είναι το μέγιστο επιτρεπτό για πλοήγηση στο Διαδίκτυο, και ο κος Green έχει ορίσει (από πριν) την ποιότητα ως τον πιο σημαντικό παράγοντα.

■ Στο Μετρό

Ο κος Green φτάνει στο σταθμό του μετρό. Το download της οικονομικής αναφοράς δεν έχει ολοκληρωθεί. Επειδή το μετρό καθυστερεί, ο κος Green συνειδητοποιεί ότι δε θα καταφέρει να φτάσει στο γραφείο στην ώρα του για το πρωινό meeting. Έτσι, ξεκινάει μια βιντεοκλήση, ώστε να συμμετάσχει στο meeting επί τόπου.

Το τερματικό, λοιπόν, εξετάζει τις προσφερόμενες τεχνολογίες. Η WLAN, προσφέρει το μέγιστο επιτρεπτό επίπεδο ποιότητας, σε υψηλά bitrates και χαμηλές τιμές, οπότε και επιλέγεται.

■ Διαδρομή προς το Κτήριο της Εταιρίας

Κατεβαίνοντας από το μετρό στον πιο κοντινό στο γραφείο σταθμό, ο επιχειρηματίας του σεναρίου βαδίζει προς το κτήριο συνεχίζοντας να συμμετέχει στο meeting. Στη φάση αυτή, το web browsing κατανέμεται στο GSM#4 σε επίπεδο QoS «3», αφού ο συνδυασμός αυτός προσφέρει, για τη συγκεκριμένη υπηρεσία, την καλύτερη ποιότητα στο χαμηλότερο κόστος, έναντι του UMTS#3 που ναι μεν προσφέρει QoS level 4, αλλά είναι πιο ακριβό. Παράλληλα, η βιντεοκλήση ανατίθεται στο UMTS#3, επειδή προσφέρει μεγαλύτερη ισχύ σήματος (ενώ το κόστος είναι περίπου το ίδιο με το GSM#4), και στο επίπεδο «3», που είναι το μέγιστο επιτρεπτό.

Συνεχίζοντας τη διαδρομή, ο κ. Green αποφασίζει ότι δεν του επαρκεί η τρέχουσα ποιότητα της βιντεοκλήσης, οπότε τροποποιεί το προφίλ του και ορίζει το μέγιστο επιτρεπτό επίπεδο για αυτή την υπηρεσία στο «5», από το «3» που ήταν. Μετά την τροποποίηση του προφίλ, λοιπόν, η βιντεοκλήση κατανέμεται στο υψηλότερο επίπεδο ποιότητας (επίπεδο 5), ούτως ώστε να ικανοποιήσει τις νέες προτιμήσεις του χρήστη. Αυτό είναι το μέγιστο επίπεδο στο οποίο μπορεί να τρέξει μια βιντεοκλήση, και είναι η προτιμότερη επιλογή, αφού ο χρήστης όρισε την «ποιότητα» ως τον πιο σημαντικό παράγοντα.

■ Στο Γραφείο

Ο κος Green φτάνει τελικά στο γραφείο. Κατευθύνεται ταχέως προς την αίθουσα συνεδριάσεων και μπαίνοντας τερματίζει τις υπηρεσίες web browsing και βιντεοκλήσης.

Η εμφάνιση νέων attachment points οδηγεί το τερματικό στο να επανεξετάσει τις τρέχουσες κατανομές των υπηρεσιών σε δίκτυα και επίπεδα ποιότητας. Στο σημείο αυτό, πριν τερματιστούν οι εφαρμογές, επιλέγεται η τεχνολογία WLAN, καθώς προσφέρει υψηλότερα bit-rates σε χαμηλή τιμή και στα μέγιστα επιτρεπτά επίπεδα ποιότητας.

ΛΟΓΙΣΜΙΚΟ ΛΗΨΗΣ ΠΑΡΑΜΕΤΡΩΝ ΑΣΥΡΜΑΤΗΣ ΚΑΡΤΑΣ ΔΙΚΤΥΟΥ

5.1 ΕΙΣΑΓΩΓΗ

Όπως είδαμε και στα προηγούμενα κεφάλαια, η ανάπτυξη μιας σύνθετης B3G ραδιοϋποδομής που επιτρέπει από άκρο σε άκρο network και terminal reconfigurability, εξελίσσει τα σχετικά αυτοματοποιημένα τερματικά σε ευφυή, προσαρμόσιμα, επαναδιαμορφώσιμα συστήματα, ικανά να υποστηρίξουν και να εκμεταλλευτούν τις δυνατότητες και τα πλεονεκτήματα που προκύπτουν από την ένωση των διαφορετικών τεχνολογιών ραδιοπρόσβασης.

Προκειμένου, όμως, να μπορούν τα τερματικά να προσαρμόζονται δυναμικά στις συνεχώς μεταβαλλόμενες συνθήκες των δικτύων, απαραίτητη είναι η αντίστοιχη διαχείρισή τους, η οποία όπως ήδη είδαμε, ενεργοποιείται μετά από την εκάστοτε διέγερση. Είδαμε, λοιπόν, ότι το τερματικό συλλέγει πληροφορία για την τρέχουσα σύνδεση και τη συγκρίνει με τα προσδιορισμένα κατώφλια, ώστε αν οι τιμές δε συμφωνούν μεταξύ τους, να ξεκινήσει η φάση της ανίχνευσης των κατάλληλων επαναδιαμορφώσεων και δικτύων.

Η πληροφορία που συλλέγεται έχει να κάνει με παραμέτρους όπως η ισχύς του λαμβανόμενου σήματος, το επίπεδο θορύβου, ο ρυθμός μετάδοσης δεδομένων, η τρέχουσα συχνότητα λειτουργίας και άλλες. Μετά τη λήψη τους, οι παράμετροι συμμετέχουν σε αλγορίθμους επιλογής, διαπραγμάτευσης και ελέγχου, με στόχο την βέλτιστη κατανομή των εφαρμογών και υπηρεσιών στα κατάλληλα δίκτυα και τεχνολογίες.

Για να ολοκληρωθεί η φάση της συλλογής πληροφοριών και παραμέτρων της τρέχουσας σύνδεσης, απαιτείται το κατάλληλο λογισμικό το οποίο να τις λαμβάνει από την ασύρματη κάρτα δικτύου και να τις αποστέλλει στους κατάλληλους αλγορίθμους και μηχανισμούς που θα τους ελέγξουν και θα τους συγκρίνουν με τα αναφερθέντα κατώφλια και με τις προτιμήσεις του χρήστη όπως αυτός τις έθεσε στο αντίστοιχο Προφίλ.

Στην κατεύθυνση αυτή, λοιπόν, εστιάζει η προκείμενη πτυχιακή εργασία, η οποία ως στόχο έχει την ανάπτυξη και υλοποίηση λογισμικού λήψης παραμέτρων από ασύρματη κάρτα και εμφάνιση αυτών σε UI (User Interface) τερματικού, τύπου Internet Tablet.

Πιο συγκεκριμένα, κατά την εκπόνηση της εργασίας και την υλοποίηση του λογισμικού, χρησιμοποιήθηκε και τροποποιήθηκε καταλλήλως η τεχνολογία των IWTools (και συγκεκριμένα το Iwconfig) της Hewlett Packard, έτσι ώστε να λαμβάνει και να επιστρέφει τις μετρήσεις των ασύρματων καρτών δικτύου σε ένα γραφικό περιβάλλον χρήστη υλοποιημένο στην πλατφόρμα του Maemo, η οποία προσομοιώνει το Nokia 770 Internet Tablet.

Πριν, όμως, προχωρήσουμε στην ανάλυση και περιγραφή των λειτουργιών του προκείμενου προγράμματος, σκόπιμο είναι να δοθεί το θεωρητικό και λειτουργικό υπόβαθρο των χρησιμοποιηθέντων τεχνολογιών μέσω των οποίων πραγματοποιήθηκε η υλοποίηση λογισμικού λήψης και εμφάνισης παραμέτρων ασύρματης πρόσβασης. Η περιγραφή των τεχνολογιών θα γίνει κατά σειρά «από κάτω προς τα πάνω», ξεκινώντας από το πρόγραμμα πυρήνα και καταλήγοντας στη γραφική διεπαφή του χρήστη.

5.2 IWTOOLS

Το Linux Wireless Extension και τα Wireless Tools είναι Έργα Ανοιχτού Κώδικα (Open Source Projects) που στηρίζονται από την Hewlett Packard από το 1996 και δημιουργούνται με τη συμβολή πολλών χρηστών Linux ανά τον κόσμο.

Το Wireless Extension είναι ένα γενικό Application Programming Interface το οποίο επιτρέπει στους drivers δικτύων να παρουσιάζουν στο χρήστη στατιστικά στοιχεία, καθώς και configurations ενός κοινού ασύρματου LAN, δίνοντάς του παράλληλα τη δυνατότητα να τροποποιήσει και να ρυθμίσει τις παραμέτρους αυτές. Είναι, με άλλα λόγια, ένα απλό σετ από εργαλεία που υποστηρίζει όλες τις εκδόσεις και παραλλαγές των Wireless LANs, ανεξάρτητα από τον τύπο τους, το οποίο επιτρέπει στους χρήστες να αλλάζουν on the fly τις παραμέτρους της ασύρματης σύνδεσής τους, χωρίς να απαιτείται η επανεκκίνηση του αντίστοιχου driver ή του λειτουργικού συστήματος.

Η τροποποίηση και ο χειρισμός των Wireless Extensions γίνεται μέσω των Wireless Tools, ενός συνόλου, δηλαδή, σχετικά ανεπεξέργαστων εργαλείων που στοχεύουν στην πλήρη υποστήριξη των Extensions. Κάποια παραδείγματα Wireless Tools, καθώς και μία λακωνική περιγραφή των λειτουργιών τους, συνοψίζονται παρακάτω:

- **iwconfig**: Χειρίζεται τις βασικές παραμέτρους ασύρματης πρόσβασης. Είναι το κυριότερο «Wireless» εργαλείο και χρησιμοποιείται για διαχείριση των συσκευών και την παρουσίαση των πιο χρήσιμων παραμέτρων. Πιο αναλυτικά, θα το δούμε παρακάτω.
- **iwlist**: Επιτρέπει την ανίχνευση ασύρματων σημείων πρόσβασης, και τη δημιουργία καταλόγων με όλες τις υποστηριζόμενες συχνότητες, τους ρυθμούς μετάδοσης, τα κλειδιά κρυπτογράφησης κ.ά., για το κάθε ένα από αυτά.
- **iwspy**: Ανιχνεύει και επιστρέφει την συνολική ποιότητα ζεύξης ανά κόμβο. Παράλληλα, ελέγχει την υποστήριξη του Mobile IP και επιτρέπει την απόκτηση στατιστικών ανά MAC address. Για κάποιους drivers και συσκευές, δε, αποτελεί το μοναδικό τρόπο λήψης στατιστικών και παραμέτρων σε Ad-Hoc mode.

- iwpriv: Επιτρέπει το χειρισμό των ιδιωτικών ioctls (input/ output controls) ενός driver, δηλαδή όλων των ιδιωτικών παραμέτρων οι οποίες δεν αποτελούν μέρος του iwconfig.
- IfRename: Επιτρέπει την, βασισμένη σε διάφορα στατικά κριτήρια, μετονομασία των διεπαφών.
- iwgetid: Εμφανίζει την ESSID ή NWID μίας συγκεκριμένης συσκευής.
- iwlib: Είναι η βιβλιοθήκη των Wireless Tools και χρησιμοποιείται στην περίπτωση που ο χρήστης επιθυμήσει να δημιουργήσει δικές του εφαρμογές πάνω στα Wireless Extensions.

Οι περισσότερες διανομές Linux προσφέρουν έτοιμα, προ – μεταγλωττισμένα πακέτα των παραπάνω εργαλείων, με τις περισσότερες από αυτές να τις έχουν by default προεγκατεστημένες. Αξίζει να αναφερθεί ότι η κάθε διανομή έχει τα δικά της configuration scripts, τα οποία είναι σχετικά διαφορετικά από τα αντίστοιχα άλλων distributions. Η εκτέλεση των εργαλείων αυτών, μπορεί να γίνει απλά, εισάγοντας την αντίστοιχη εντολή στην κονσόλα των Linux (για παράδειγμα «iwlist scanning»).

5.2.1 IWConfig

Το iwconfig, όπως αναφέρθηκε και πιο πάνω, χρησιμοποιείται για τη ρύθμιση των ασύρματων διεπαφών και τον ορισμό των παραμέτρων της ασύρματης δικτυακής διεπαφής, όπως για παράδειγμα της συχνότητας. Εάν δεν προσδιοριστούν οι παράμετροι, μπορεί επίσης να χρησιμοποιηθεί απλά για την εμφάνιση των στατιστικών στοιχείων και των μετρήσεων της διεπαφής, όπως αυτές εξάγονται από το directory: «proc/net/wireless» των Linux.

Οι τιμές των παραμέτρων και των στατιστικών εξαρτώνται από τη συσκευή. Παράλληλα, κάθε driver παρέχει ένα υποσύνολο αυτών των παραμέτρων, ανάλογα με την υποστήριξη του υλικού, με το εύρος των τιμών να ποικίλλει από συσκευή σε συσκευή. Παρακάτω, θα δούμε ποιες ακριβώς είναι οι υπό συζήτηση παράμετροι, αναφέροντας, για κάποιες από αυτές, ενδεικτικά τις τιμές και τα χαρακτηριστικά που μπορούν να λάβουν και να επιστρέψουν.

■ MAC Protocol

Η παράμετρος αυτή εμφανίζει το όνομα του χρησιμοποιούμενου MAC πρωτοκόλλου, π.χ. «IEEE 802.11g». Εάν το πρωτόκολλο είναι ιδιόκτητο, εμφανίζει μόνο το όνομα της ασύρματης συσκευής.

■ ESSID

Η ESSID παράμετρος, εμφανίζει το ESSID (Extended Service Set Identifier) ή το όνομα του δικτύου (Network Name). Χρησιμοποιείται για την αναγνώριση των κυψελών που βρίσκονται στο ίδιο εικονικό δίκτυο. Σε αντίθεση με τη διεύθυνση του AP ή το NWID, τα οποία προσδιορίζουν μία συγκεκριμένη κυψέλη, η ESSID προσδιορίζει μία ομάδα κυψελών που είναι συνδεδεμένες μέσω επαναληπτών ή της υποδομής, στις οποίες ο χρήστης μπορεί να περιιάγεται διαφανώς.

■ Mode

Η Mode παράμετρος δείχνει τον εξαρτώμενο από την τοπολογία του δικτύου, τύπο λειτουργίας της συσκευής. Ο τύπος της λειτουργίας μπορεί να είναι: Ad – Hoc (δίκτυο αποτελούμενο μόνο από μία κυψέλη χωρίς σημείο πρόσβασης), Managed (που σημαίνει ότι ο κόμβος είναι συνδεδεμένος σε ένα δίκτυο το οποίο αποτελείται από πολλά Access Points), Master (που συνεπάγεται ότι ο κόμβος που εξετάζεται είναι είτε ο Synchronisation Master είτε λειτουργεί ως Access Point), Repeater (δηλαδή ο κόμβος προωθεί πακέτα μεταξύ άλλων ασύρματων κόμβων), Secondary (που σημαίνει ότι ο κόμβος λειτουργεί είτε ως backup master είτε ως repeater), Monitor (που δείχνει ότι ο κόμβος δεν είναι συσχετισμένος με καμία κυψέλη και ελέγχει παθητικά όλα τα πακέτα στη συχνότητα λειτουργίας του) ή Auto (δηλαδή ο κόμβος επιλέγει αυτόματα τον τύπο λειτουργίας του, βασισμένος στα στοιχεία και τα χαρακτηριστικά του AP).

■ Frequency

Εμφανίζει τη συχνότητα λειτουργίας ή το κανάλι της συσκευής. Εάν η εμφανιζόμενη τιμή είναι μικρότερη από 1000, τότε εμφανίζεται το κανάλι, ενώ αν είναι μεγαλύτερη, εμφανίζεται η συχνότητα σε Hz.

■ Access Point

Λαμβάνει και εμφανίζει τη διεύθυνση του σημείου πρόσβασης. Εάν η διεύθυνση είναι “00:00:00:00:00:00”, συνεπάγεται ότι η κάρτα απέτυχε να

συσχετιστεί με ένα Access Point. Εναλλακτικά, σε περίπτωση αποτυχίας σύνδεσης της συσκευής με ένα Access Point, μπορεί να επιστραφεί η τιμή «Not – Associated».

■ Bit Rate

Όπως είναι προφανές, η παράμετρος αυτή υπολογίζει και εμφανίζει το BitRate, σε bits το δευτερόλεπτο, δηλαδή την ταχύτητα με την οποία εκπέμπονται τα bits στο μέσο.

■ Tx – Power

Η παράμετρος αυτή λαμβάνει και παρουσιάζει την εκπεμπόμενη από τη συσκευή ισχύ, σε dBm.

■ Sensitivity

Η παράμετρος Sensitivity, εμφανίζει το ρυθμισμένο κατώφλι ευαισθησίας, προσδιορίζοντας το πόσο ευαίσθητη είναι η κάρτα σε δυσμενείς συνθήκες λειτουργίας (όπως χαμηλό επίπεδο σήματος, παρεμβολή και άλλα). Η θετική επιστρεφόμενη τιμή του κατωφλίου ευαισθησίας, δηλώνει την ανεπεξέργαστη τιμή που χρησιμοποιείται από το υλικό. Εάν, δε, η τιμή είναι αρνητική, τότε εμφανίζεται η Sensitivity σε μονάδες dBm.

Ανάλογα με την υλοποίηση του hardware, αυτή η παράμετρος μπορεί να ελέγξει ποικίλλες λειτουργίες. Σε σύγχρονες κάρτες, η παράμετρος αυτή συνήθως ελέγχει το handover / roaming threshold, δηλαδή το χαμηλότερο επίπεδο σήματος για το οποίο το υλικό παραμένει συσχετισμένο με το τρέχον Access Point. Όταν το επίπεδο σήματος πέσει κάτω από το κατώφλι, η κάρτα αρχίζει να αναζητά νέο/ καλύτερο Access Point. Μία υψηλή τιμή του κατωφλίου διαβεβαιώνει ότι η κάρτα θα είναι πάντα συσχετισμένη με το καλύτερο AP, ενώ μια χαμηλή μειώνει τον αριθμό των αποτυχημένων handoffs.

Για παλαιότερες κάρτες δικτύου, η παράμετρος αυτή συνήθως ελέγχει το defer threshold, δηλαδή το χαμηλότερο επίπεδο σήματος για το οποίο το υλικό κρίνει πως το κανάλι είναι κατειλημένο. Επίπεδα σήματος υψηλότερα από το κατώφλι αναγκάζουν το υλικό να εμποδίζει τις μεταδόσεις του, ενώ σε χαμηλότερα από το κατώφλι επίπεδα τα σήματα αγνοούνται και το υλικό είναι ελεύθερο να εκπέμψει.

■ RTS Threshold

Το RTS/ CTS (Request to Send / Clear to Send) προσθέτει μία χειραψία πριν από κάθε μετάδοση πακέτου, ώστε να επιβεβαιωθεί ότι το κανάλι είναι ελεύθερο. Η διαδικασία αυτή προσθέτει επιπλέον φορτίο, αλλά αυξάνει την απόδοση στην περίπτωση κρυμμένων κόμβων ή μεγάλου αριθμού ενεργών κόμβων. Η παράμετρος αυτή, λοιπόν, θέτει το μέγεθος του μικρότερου πακέτου για το οποίο ο κόμβος θα στείλει RTS. Εάν η τιμή ρυθμιστεί να είναι ίση με το μέγιστο μέγεθος πακέτου, ο μηχανισμός ελέγχου απενεργοποιείται.

■ Fragmentation Threshold

Κατά τον κατακερματισμό, ένα IP πακέτο χωρίζεται σε μικρότερα τμήματα, τα οποία εκπέμπονται στο μέσο. Στις περισσότερες περιπτώσεις, η διαδικασία αυτή προσθέτει επιπλέον φορτίο (overhead), αλλά σε πολύ θορυβώδη περιβάλλοντα μειώνει το επίπεδο λαθών και επιτρέπει στα πακέτα να περάσουν μέσα από ριπές παρεμβολών. Η παράμετρος αυτή, λοιπόν, θέτει και εμφανίζει το μέγιστο μέγεθος κατάτμησης των IP πακέτων.

■ Power Management

Η διαχείριση ισχύος περιλαμβάνει λειτουργίες που χειρίζονται και ελέγχουν την ένταση/ ισχύ της κάρτας. Θέτει παραμέτρους όπως η περίοδος ανάμεσα σε ενεργοποιήσεις της κάρτας, power saving καταστάσεις και άλλα. Η διαχείριση μπορεί να περιλαμβάνει όλα τα ληφθέντα πακέτα ή συγκεκριμένα από αυτά (π.χ. μόνο unicast, μόνο multicast, απόρριψη συγκεκριμένων πακέτων και άλλα). Εάν, για παράδειγμα, η παράμετρος Power Management δείχνει ότι υφίσταται διαχείριση ισχύος η οποία στοχεύει μόνο σε unicast πακέτα, σημαίνει ότι το τερματικό αγνοεί όλα τα υπόλοιπα (non – unicast) πακέτα, εξοικονομώντας, έτσι, ισχύ η οποία θα καταναλωνόταν αν τα λάμβανε.

■ Link Quality

Η παράμετρος Link Quality, υπολογίζει και εμφανίζει τη συνολική ποιότητα της σύνδεσης. Η τιμή μπορεί να βασιστεί στα επίπεδα παρεμβολής, στο ρυθμό δεδομένων, στο FER, στην ισχύ του λαμβανόμενου σήματος και σε άλλες μετρήσεις της ασύρματης κάρτας.

■ Signal Level

Όπως δηλώνει και το όνομά της, η παράμετρος αυτή εμφανίζει την ένταση του λαμβανόμενου σήματος (το πόσο ισχυρό είναι το σήμα που λαμβάνει η κάρτα), χρησιμοποιώντας τη meta – πληροφορία του driver που δίνεται από το /proc/net/wireless.

■ Noise Level

Εμφανίζει το επίπεδο του παρασιτικού θορύβου (που υφίσταται όταν κανένα πακέτο δεν εκπέμπεται). Όπως και με το Signal Level, η πληροφορία λαμβάνεται από τη meta – information του driver (/proc/net/wireless).

■ Rx invalid nwid

Εμφανίζει τον αριθμό των πακέτων που λήφθηκαν με διαφορετική NWID ή ESSID. Χρησιμοποιείται για τον εντοπισμό προβλημάτων ρύθμισης (configuration problems) ή την ύπαρξη γειτονικού δικτύου (στην ίδια συχνότητα).

■ Rx invalid crypt

Εμφανίζει τον αριθμό των πακέτων, τα οποία δεν μπόρεσαν να αποκρυπτογραφηθούν από το υλικό. Μπορεί να χρησιμοποιηθεί για τον εντοπισμό μη έγκυρων ρυθμίσεων κρυπτογράφησης.

■ Rx invalid fragmentation

Εμφανίζει τον αριθμό των πακέτων, τα οποία δεν επανασυνδέθηκαν ορθά από το υλικό.

■ Tx excessive retries

Εμφανίζει τον αριθμό των πακέτων, τα οποία δεν κατάφερε να παραδώσει το hardware. Τα περισσότερα MAC protocols, θα επαναπροωθήσουν το πακέτο (για συγκεκριμένο πλήθος φορών) πριν εγκαταλείψουν την προσπάθεια.

- Invalid misc

Εμφανίζει άλλα πακέτα που χάθηκαν για λόγους που σχετίζονται με συγκεκριμένες λειτουργίες ασύρματης πρόσβασης.

- Missed beacon

Εμφανίζει το πλήθος των περιοδικών beacons που χάθηκαν από την κυψέλη ή το AP. Τα beacons στέλνονται σε τακτές χρονικές περιόδους για λόγους διατήρησης του συντονισμού της κυψέλης και η αποτυχία της λήψης τους, συνήθως υποδηλώνει ότι η κάρτα είναι εκτός εμβέλειας.

5.3 D – BUS

Το D - Bus είναι ένας διαδιεργασιακός μηχανισμός επικοινωνίας - ένα μέσο για την τοπική επικοινωνία ανάμεσα σε διεργασίες που τρέχουν στον ίδιο host. Είναι connection - based, πράγμα που τον κάνει πιο «ευφυή» από τα χαμηλότερου επιπέδου πρωτόκολλα όπως το UDP, και μεταφέρει μηνύματα ως ξεχωριστά στοιχεία, αντί για συνεχείς ροές δεδομένων που μεταφέρει το TCP. Παράλληλα, το D - Bus περιέχει μία δομημένη θεώρηση των δεδομένων που μεταφέρει και τα αντιμετωπίζει ως δεδομένα binary μορφής: ακεραίους αριθμούς, πεδία, αριθμούς κινητής υποδιαστολής και τα λοιπά. Γίνεται προφανές, λοιπόν, ότι για το D - Bus τα δεδομένα δεν είναι απλά σύνολα από bytes, αλλά είναι στοιχεία προκαθορισμένου είδους και τύπου.

Η υλοποίηση αυτού του μηχανισμού επικοινωνίας, γίνεται μέσω ποικίλων γλωσσών προγραμματισμού. Για την αποφυγή όμως, του low – level programming, οι γλώσσες αυτές μπορούν να προσφέρουν ειδικές διεπαφές (bindings), οι οποίες αποκρύπτουν τις λεπτομέρειες του DBus προγραμματισμού και διευκολύνουν τη συγγραφή και υλοποίηση μονάδων λογισμικού. Χάρη στα bindings, ο προγραμματισμός χρήσης DBus λαμβάνει μία πιο αντικειμενοστρεφή μορφή, με αποτέλεσμα ο προγραμματιστής να μη βλέπει τις εμπλεκόμενες οντότητες ως τμήματα προγράμματος αλλά ως οντότητες. Ας δούμε όμως ποιες είναι αυτές οι οντότητες, και ποιες οι λειτουργίες και στόχοι που υλοποιούν για την επιτυχή διαδιεργασιακή επικοινωνία μεταξύ των επικοινωνούντων εφαρμογών.

- DBus Daemon

Για να μπορέσουν δύο ή περισσότερες εφαρμογές να ανταλλάξουν μηνύματα, θα πρέπει πρωτίστως να συνδεθούν στον dbus daemon, ο οποίος λειτουργεί σαν ένας «δρόμος» μέσω του οποίου δρομολογούνται τα μηνύματα. Κάθε διεργασία που επιθυμεί να «επικοινωνήσει» με κάποια άλλη, συνδέεται στο bus, αποστέλλει τα μηνύματά της, και το bus (σαν ένας router) τα προωθεί στη διεργασία προορισμού.

Οι DBus Daemons χωρίζονται σε δύο κατηγορίες: System Bus και Session Bus. Το System Bus χρησιμοποιείται για τη λήψη μηνυμάτων που αφορούν σε γεγονότα του συστήματος ή του πυρήνα (όπως π.χ. στην προσθήκη νέας συσκευής). Το Session Bus, από την άλλη, χρησιμοποιείται για την inter – process communication ανάμεσα σε διεργασίες που βρίσκονται είτε στο ίδιο είτε σε καταμεμημένο σύστημα.

Σε κάθε bus αποδίδεται ένα μοναδικό όνομα, μια διεύθυνση δηλαδή, η οποία περιγράφει τον τρόπο με τον οποίο μία διεργασία μπορεί να συνδεθεί σε αυτό. Οι διευθύνσεις αυτές έχουν τη μορφή είτε ενός filename από ένα Unix – domain socket είτε ενός TCP port μέσω του οποίου ο daemon λαμβάνει τα αφιχθέντα μηνύματα. Αξίζει, βέβαια, να τονιστεί ότι χάρη στα bindings, η πληροφορία και η διαδικασία σύνδεσης και διευθυνσιοδότησης του bus αποκρύπτεται, με αποτέλεσμα ο προγραμματιστής να μη «βλέπει» το dbus με το όνομά του αλλά με τη μορφή μίας σύνδεσης με αυτό.

- Αντικείμενα

Τα αντικείμενα είναι, ουσιαστικά, οι απεικονίσεις των απομακρυσμένων διεργασιών. Συμπεριφέρονται σαν οντότητες οι οποίες έχουν συγκεκριμένα χαρακτηριστικά και υλοποιούν μεθόδους (προγραμματιστικές συναρτήσεις). Με πολύ απλά λόγια, όταν μία διεργασία A ζητάει μία μέθοδο από την απομακρυσμένη διεργασία B, αποστέλλει τα κατάλληλα μηνύματα προς την τελευταία, ενεργοποιώντας έτσι τη λειτουργία του αντικειμένου, το οποίο λαμβάνει το request message, εξάγει τις αποσταλμένες παραμέτρους (εάν υπάρχουν), τις χρησιμοποιεί για να τρέξει την αντίστοιχη μέθοδο/ συνάρτηση που υλοποιεί και επιστρέφει τα αποτελέσματα.

- Proxies

Όταν δημιουργείται ένα αντικείμενο, «ζει» έξω από το πρόγραμμα και προκειμένου μία διεργασία να έχει πρόσβαση σε αυτό, χρησιμοποιεί αναφορές οι οποίες είναι γνωστές ως «Proxies» και έχουν τη μορφή προγραμματιστικών αντικειμένων. Οι proxies, ουσιαστικά, μετατρέπουν τις κλήσεις μεθόδων σε method calls προς την απομακρυσμένη διεργασία, τις αποστέλλουν, αναμένουν για και λαμβάνουν την αντίστοιχη απάντηση, εξάγουν την τιμή της και την περνάνε στο τοπικό πρόγραμμα.

- Μέθοδοι

Οι μέθοδοι είναι οι υπηρεσίες που προσφέρουν τα αντικείμενα και έχουν συνήθως τη μορφή προγραμματιστικών συναρτήσεων. Ο client δηλαδή, όταν αποστέλλει ένα μήνυμα αίτησης υπηρεσίας προς ένα αντικείμενο, ουσιαστικά ζητάει από το τελευταίο να εκτελέσει την αντίστοιχη μέθοδο και να του επιστρέψει την παραγόμενη τιμή μέσω του reply message. Η διαδικασία της κλήσης μίας μεθόδου αποτελείται, λοιπόν, από δύο μηνύματα: ένα request μήνυμα από την client διεργασία προς το απομακρυσμένο αντικείμενο, και ένα reply μήνυμα από το αντικείμενο προς τον client.

Όπως ήδη αναφέρθηκε πιο πάνω, η κλήση της μεθόδου γίνεται μέσω του proxy ο οποίος δημιουργεί το request message, το οποίο περιέχει το όνομα της μεθόδου, τις αντίστοιχες παραμέτρους εφ' όσον αυτό απαιτείται, το όνομα του απομακρυσμένου αντικειμένου και το όνομα του interface που ορίζει τη μέθοδο. Το request μήνυμα αποστέλλεται στον bus daemon, ο οποίος εξετάζει τη διεύθυνση προορισμού (δηλαδή το όνομα της διεργασίας στην οποία απευθύνεται το μήνυμα) και το προωθεί στη διεργασία. Η τελευταία, το παραλαμβάνει και το προωθεί στο αντικείμενο, το οποίο υλοποιεί την αιτούμενη μέθοδο. Το αντικείμενο με τη σειρά του, εκτελεί τη μέθοδο και επιστρέφει τα αντίστοιχα αποτελέσματα τα οποία τελικά μετατρέπονται σε reply message το οποίο προωθείται στο bus. Ο daemon λαμβάνει το reply μήνυμα, εξετάζει πάλι τη διεύθυνση προορισμού και το δρομολογεί τελικά στη διαδικασία η οποία αιτήθηκε της μεθόδου.

- Διεπαφές

Το όνομα, τα χαρακτηριστικά και οι μέθοδοι των αντικειμένων προσδιορίζονται μέσω των interfaces, τα οποία έχουν την ίδια λειτουργία με τα αντίστοιχα της Java. Οι διεπαφές έχουν τη μορφή XML αρχείων στα οποία προσδιορίζεται το όνομα του αντικειμένου, η διεύθυνσή του, οι μέθοδοι που αυτό υλοποιεί καθώς και οι παράμετροι τις οποίες αυτό λαμβάνει ή/ και επιστρέφει ανά μέθοδο.

- Μηνύματα

Οι τύποι των μηνυμάτων που μπορούν να χρησιμοποιηθούν κατά την διαδιεργασιακή επικοινωνία μέσω DBus είναι τέσσερις: κλήσεις μεθόδων, απαντήσεις μεθόδων, μηνύματα λάθους και σήματα. Κάθε μήνυμα αποτελείται από ένα header και το body. Η κεφαλίδα προσδιορίζει το είδος του μηνύματος και περιέχει τις απαραίτητες διευθύνσεις πηγής και προορισμού, το όνομα της μεθόδου που καλείται και άλλα. Το σώμα του μηνύματος περιέχει το «ωφέλιμο φορτίο», όπως για παράδειγμα τις παραμέτρους που αποστέλλει ο client ή τις τιμές που επιστρέφει ο server.

5.4 MAEMO

Η πλατφόρμα του Maemo είναι το λειτουργικό σύστημα που τρέχει σε Internet Tablets (υπολογιστές χειρός) όπως το Nokia 770. Είναι δημιουργημένη από πολλά μέρη ανοιχτού κώδικα και αποτελείται από το software stack του πυρήνα των Linux και το Hildon UI framework. Λόγω του ότι βασίζεται στο λειτουργικό σύστημα των Linux, δανείζεται αρκετά στοιχεία από την αρχιτεκτονική του τελευταίου.

Το περιβάλλον χρήστη του Maemo βασίζεται στο GNOME framework, και πιο συγκεκριμένα στο GTK+ toolkit, το οποίο προσφέρει ένα μεγάλο πλήθος widgets για τη δημιουργία διεπαφών. Σαν «παιδί» του GNOME, το Maemo έχει κληρονομήσει πολλές βασικές συνιστώσες του, όπως το GStreamer framework πολυμέσων, την GConf διαχείριση ρυθμίσεων και την XML βιβλιοθήκη.

Το βασικό κανάλι επικοινωνίας των Maemo applications είναι το DBUS, το οποίο προσφέρει και ένα κανάλι για την αλληλεπίδραση των εφαρμογών με το σύστημα. Έτσι, εκτός από τη διαδικαριακή επικοινωνία, το Maemo προσφέρει και επικοινωνία με τα χαμηλότερα στρώματα του λειτουργικού, για να «ακούει» σε γεγονότα όπως η «χαμηλή ισχύς μπαταρίας».

Η διεπαφή χρήστη βασίζεται στο X Window System, έχοντας το Matchbox window για manager. Το API πάνω από το X Window αποτελείται από το GTK+ widget toolkit το οποίο επεκτείνεται με τα Hildon extensions. Το Hildon framework, ουσιαστικά, προσθέτει επιπλέον οντότητες στο GNOME ώστε να μπορεί να υποστηρίξει στοιχεία όπως ο πίνακας ελέγχου, το status bar, ο task navigator και άλλα.

Τα απαραίτητα εργαλεία για τη δημιουργία εφαρμογών, παρέχονται από το Maemo SDK, το οποίο δημιουργήθηκε για να προσομοιώνει το Nokia Internet Tablet σε έναν απλό υπολογιστή. Χάρη σε αυτό, είναι δυνατή η δημιουργία και η δοκιμή εφαρμογών σε περιβάλλον ενός κοινού desktop, ο οποίος θα παράγει τα ίδια αποτελέσματα και την ίδια συμπεριφορά που θα είχε η εφαρμογή σε μία συσκευή που τρέχει το Maemo. Τα εργαλεία ανάπτυξης αυτά, βρίσκονται στο Scratchbox (cross-compiler εξ' ολοκλήρου βασισμένος στο Linux). Έτσι, γίνεται πλέον εμφανές ότι ό,τι μπορεί να μεταγλωττιστεί και να τρέξει στο Scratchbox, θα μπορεί να τρέξει και στο Internet Tablet (τουλάχιστον κατά 99%).

5.5 ΜΕΘΟΔΟΛΟΓΙΑ

5.5.1 Στόχος

Ανάπτυξη λογισμικού το οποίο θα λαμβάνει μετρήσεις ασύρματης κάρτας δικτύου και θα τις εμφανίζει μέσω γραφικού περιβάλλοντος χρήστη στην πλατφόρμα του Maemo.

5.5.2 Εργαλεία, τεχνολογίες

- Τελευταία έκδοση του προγράμματος Iwconfig (από το πακέτο «Wireless Tools 29»)
- Πλατφόρμα του Maemo, έκδοση Gregale 2.2
- D – BUS
- Γλώσσα προγραμματισμού C
- GTK+ toolkit
- Λειτουργικό σύστημα: Linux Ubuntu 7.10, πάνω από διπύρινο AMD επεξεργαστή 64 bits
- Linksys WUSB54GC (Wireless – G USB adapter)

5.5.3 Υλοποίηση

5.5.3.1 Λήψη Μετρήσεων Ασύρματης Κάρτας

Όπως ήδη αναφέρθηκε σε παραπάνω ενότητα, το `lwconfig` είναι ένα πρόγραμμα πυρήνα, το οποίο χρησιμοποιείται για τη λήψη, εμφάνιση και ρύθμιση των παραμέτρων ενός ασύρματου `driver`. Ο κώδικάς του αποτελείται από τέσσερα κυρίως μέρη: μία συνάρτηση λήψης παραμέτρων από την ασύρματη κάρτα δικτύου (`get_info()`), δύο συναρτήσεις εμφάνισης των παραμέτρων, αρκετές συναρτήσεις ρύθμισης των παραμέτρων και τη `main`.

Οι συναρτήσεις εμφάνισης των παραμέτρων είναι οι: `display_info()` και `print_info`. Η `print_info()` ελέγχει το αν οι διεπαφές έχουν `wireless extensions` (δηλαδή, το αν είναι ασύρματες ή όχι). Εάν έχουν, καλεί την `display_info()` ανά διεπαφή. Αλλιώς, εμφανίζει το μήνυμα «no wireless extensions». Η `display_info()`, λαμβάνει τις μετρήσεις και τις εμφανίζει μέσα από κατάλληλες συνθήκες «if» οι οποίες ελέγχουν το αν υπάρχουν οι εκάστοτε τιμές παραμέτρων και το είδος τους. Εάν κάποια από τις παραμέτρους δεν υπάρχει, δεν εκτελούνται οι εντολές μέσα στην «if» και δεν εμφανίζεται το αντίστοιχο πεδίο κατά την εκτέλεση του προγράμματος.

Τέλος, η `main()`, δημιουργεί ένα κανάλι στο NET πυρήνα ανοίγοντας `sockets`, καλεί την `print_info()` έτσι ώστε το όνομα της συσκευής να είναι το πρώτο που θα αναγράφεται κατά την εκτέλεση, ορίζει συνθήκες οι οποίες καλούν και εκτελούν συναρτήσεις για το «help» και το «version» και τελικά τερματίζει τη σύνδεση κλείνοντας τα `sockets`.

Προκειμένου, λοιπόν, το `lwconfig` να λάβει τις ασύρματες μετρήσεις και να τις εμφανίσει στο `Maemo`, απαιτήθηκαν αρκετές αλλαγές και τροποποιήσεις, οι οποίες ακολουθούν παρακάτω:

Επειδή θέλουμε να αφαιρέσουμε τη δυνατότητα από το `χρήση` να ρυθμίζει και να τροποποιεί τις μετρήσεις της κάρτας, από τον κώδικα του `lwconfig.c` διαγράφηκαν όλες οι συναρτήσεις που ρυθμίζουν τις παραμέτρους.

Η `display_info()`, όπως είπαμε και προηγουμένως, εμφανίζει μόνο όσες παραμέτρους έχουν τιμή. Παράλληλα, χρησιμοποιεί συναρτήσεις και δομές οι οποίες επιστρέφουν πολλές από τις τιμές με προκαθορισμένο τρόπο (για παράδειγμα, η συχνότητα λειτουργίας επιστρέφεται ως «Frequency: 2 GHz», με το «Frequency:» να είναι αναπόσπαστο μέρος του επιστρεφόμενου `string`). Έτσι, διαγράφηκε όλη η `display_info()` και ξαναγράφηκε από την αρχή, έτσι ώστε:

- Οι παράμετροι οι οποίες δεν έχουν τιμή να παίρνουν τιμή «0», «Null», «None» ή «Off» (ανάλογα με την περίπτωση), έτσι ώστε κανένα πεδίο στη γραφική διεπαφή χρήστη να μη μένει κενό.
- Κάθε παράμετρος να αρχικοποιείται, ελέγχεται και εμφανίζεται μέσα από ξεχωριστή συνάρτηση. Με τον τρόπο αυτό, η τροποποίηση του προγράμματος έτσι ώστε να μπορεί να γίνει server μέσω Dbus ή να χρησιμοποιηθεί στα callbacks του GUI να μπορεί να γίνει πολύ εύκολα και με ελάχιστες εντολές. Στην περίπτωση του Dbus, κάθε μία από τις void συναρτήσεις που δημιουργήθηκαν, με τις ελάχιστες κατάλληλες τροποποιήσεις μπορεί να μετατραπεί σε μέθοδο που επιστρέφει στον client (εδώ, στο πρόγραμμα του Maemo) την εκάστοτε παράμετρο. Στην περίπτωση των callbacks, στις συναρτήσεις αυτές μπορούν να προστεθούν οι κατάλληλες εντολές οι οποίες θα λαμβάνουν ως όρισμα την κάθε παράμετρο, και μέσω των GtkWidgets θα τις εμφανίζουν στα αντίστοιχα πεδία του GUI, με μία απλή ονομαστική κλήση τους.
- Κάθε παράμετρος να έχει έναν τύπο τιμής. Επειδή το αρχικό πρόγραμμα του iwconfig απλά εμφανίζει τις παραμέτρους και τα κατάλληλα μηνύματα μέσω της συνάρτησης printf(), σε πολλές περιπτώσεις η εμφανιζόμενη τιμή διαφέρει από την επιστρεφόμενη. Για παράδειγμα, στην περίπτωση του RTS Threshold, εμφανίζεται είτε το string «Off» είτε το int info->rts.value (ανάλογα με την περίπτωση). Για να λειτουργήσουν, όμως, σωστά οι συναρτήσεις που δημιουργήθηκαν, και για να μπορέσουν να μετατραπούν σε κλήσεις των callbacks ή σε μεθόδους απομακρυσμένων διεργασιών, απαραίτητη ήταν η δημιουργία μεταβλητών στις οποίες θα αποθηκεύεται η τιμή της κάθε παραμέτρου. Έτσι, σε αρκετά σημεία, έγινε μετατροπή των αριθμητικών τύπων δεδομένων σε strings ή αντίστροφα.
- Κάθε παράμετρος να εμφανίζεται με την καθαρή τιμή της. Όπως είπαμε και προηγουμένως, σε κάποιες περιπτώσεις το iwconfig.c χρησιμοποιεί συναρτήσεις οι οποίες μετατρέπουν τις τιμές των παραμέτρων σε strings και τις επιστρέφουν μαζί με το κατάλληλο μήνυμα. Για παράδειγμα, η τιμή της συχνότητας λειτουργίας, μετατρέπεται από double σε string και επιστρέφεται αναπόσπαστα με τη λέξη «Frequency: ». Για το λόγο αυτό, απαραίτητη ήταν η χρήση προσωρινών buffers ανά παράμετρο, για την εξαγωγή και την εμφάνιση της καθαρής τιμής της.
- Κάθε παράμετρος να παραμένει στο πρόγραμμα όσο αυτό εκτελείται. Στο αρχικό πρόγραμμα του iwconfig οι περισσότερες παράμετροι αποθηκεύονταν σε προσωρινούς buffers οι οποίοι διέγραφαν τα περιεχόμενά τους μετά από κάθε printf της παραμέτρου. Έτσι, για να μπορέσουν να χρησιμοποιηθούν οι παράμετροι σε άλλες συναρτήσεις ή εφαρμογές, απαραίτητη ήταν η δημιουργία μεταβλητών και buffers στις οποίες αυτές θα αποθηκεύονται μέχρι και το τέλος του προγράμματος.

Η `print_info()` ελέγχει το αν υπάρχουν `wireless extensions`, και στην περίπτωση που δεν υπάρχουν εμφανίζει είτε «no wireless extensions» είτε το κατάλληλο μήνυμα λάθους. Αν υπάρχουν, καλεί την `display_info()`. Επειδή, όμως, στη γραφική διεπαφή του χρήστη (στο Maemo) δε θα εμφανίζονται μηνύματα λάθους, και ακόμα και αν δεν υπάρχει ασύρματη σύνδεση θα εμφανίζονται τιμές όπως «0» ή «Null» με αποτέλεσμα κανένα πεδίο να μη μένει κενό, αφαιρέθηκαν όλοι οι έλεγχοι και οι συνθήκες που αναφέρθηκαν. Έτσι, η `print_info()` πλέον έχει μόνο δύο εντολές: μία με την οποία καλεί την `get_info()` για τη λήψη των παραμέτρων, και μία με την οποία καλεί την `display_info()` για την εμφάνισή τους.

Παράλληλα, από την `main()` αφαιρέθηκαν οι εντολές που εμφανίζουν την «version» και το «help», καθώς δε μας είναι απαραίτητες, και κρατήθηκαν μόνο οι εντολές σύνδεσης στα `sockets`, κλήσης της `print_info` και αποσύνδεσης. Επίσης, επειδή όλο το πρόγραμμα του `iwconfig` θα μπει σαν συνάρτηση στο `callbacks.c`, η `main()` μετονομάστηκε σε `open_connection()`, ώστε να αντιμετωπίζεται και να καλείται σαν απλή συνάρτηση μέσα από τα κατάλληλα `callbacks`.

5.5.3.2 Γραφική Διεπαφή Χρήστη

Για τη δημιουργία της γραφικής διεπαφής χρήστη στην πλατφόρμα του Maemo, πάνω στην οποία θα εμφανίζονται οι μετρήσεις της ασύρματης κάρτας, χρησιμοποιήθηκε εξ' ολοκλήρου το GTK+ toolkit και η γλώσσα προγραμματισμού C. Το πρόγραμμα που υλοποιεί το GUI ονομάστηκε `project1.exe` και αποτελείται από τα αρχεία: `main.c`, `interface.c` και `callbacks.c`.

Η `main.c` αρχικοποιεί, κατασκευάζει και εμφανίζει το πρώτο παράθυρο του συστήματος, από το οποίο παράγονται όλα τα υπόλοιπα. Παράλληλα, στο `interface.c` αρχικοποιήθηκαν και δημιουργήθηκαν όλα τα `widgets` που εμφανίζονται στο πρόγραμμα. Στο αρχείο αυτό, αποθηκεύονται όλες οι συναρτήσεις που καθορίζουν τη θέση των `widgets` στο καρτεσιανό επίπεδο, το μέγεθός τους, το όνομά τους, τα `signals` με τα οποία συνδέονται με τα `callbacks` και οι δείκτες οι οποίοι δείχνουν σε κάθε ένα `widget` ξεχωριστά ώστε να μπορεί να χρησιμοποιηθεί επιτυχώς η συνάρτηση `lookup_widget()` την οποία θα δούμε παρακάτω. Τέλος, στο `callbacks.c` δημιουργήθηκαν όλες οι συναρτήσεις που ενεργοποιούνται με το κατάλληλο `event`. Οι συναρτήσεις αυτές έχουν να κάνουν κυρίως με την εμφάνιση ή την απόκρυψη παραθύρων και `widgets`, με την επιλογή και την ενεργοποίηση `menu widgets` καθώς και με την εμφάνιση των παραμέτρων ή των περιγραφών τους στα κατάλληλα `GtkList widgets`. Θα αναφέρουμε επιγραμματικά κάποιες από τις συναρτήσεις που δημιουργήθηκαν, για να γίνει κατανοητή και η περιγραφή του συστήματος

η οποία ακολουθεί παρακάτω. Περισσότερες πληροφορίες δίνονται υπό μορφή σχολίων στον αντίστοιχο κώδικα του παραρτήματος 1.

Gtk_widget_destroy(): Συνάρτηση του GTK+ toolkit. Καλείται όταν θέλουμε να κλείσουμε ή να εξαφανίσουμε ένα widget (π.χ. το παράθυρο window5)

Gtk_main_quit(): Συνάρτηση του GTK+ toolkit. Τερματίζει όλο το πρόγραμμα.

Lookup_widget(): Συνάρτηση του GTK+ toolkit. Λαμβάνει σαν όρισμα ένα widget και ένα όνομα και χρησιμοποιείται για την εύρεση του widget που θέλουμε. Για παράδειγμα, αν το widget είναι το window0 και το όνομα είναι «button1», βρίσκει το επιθυμητό παράθυρο και στη θέση του button1 τοποθετεί έναν pointer.

Gkt_widget_hide(): Συνάρτηση του GTK+ toolkit. Παίρνει σαν όρισμα το widget που επιθυμούμε να αποκρύψουμε. Στο πρόγραμμα, χρησιμοποιείται για την πλοήγηση. Όταν ανοίγει ένα νέο παράθυρο, αποκρύπτεται το προηγούμενο. Έτσι, είναι ενεργό πάντα ένα παράθυρο με αποτέλεσμα να μειώνεται η απαίτηση σε μνήμη κατά την εκτέλεση.

gtk_list_item_new_with_label(): Συνάρτηση του GTK+ toolkit. Λαμβάνει σαν όρισμα μία μεταβλητή τύπου «const char» και δημιουργεί ένα list item το οποίο εμφανίζει το κείμενο της μεταβλητής. Χρησιμοποιήθηκε κυρίως στο παράθυρο «Info» για την εμφάνιση της περιγραφής της κάθε ασύρματης παραμέτρου.

Overview(): Η συνάρτηση αυτή δημιουργήθηκε για την εμφάνιση των παραμέτρων στα αντίστοιχα πεδία του παραθύρου «Overview». Χρησιμοποιεί την συνάρτηση lookup_widget() για να εντοπίσει το σημείο στο οποίο πρέπει να εμφανιστεί η εκτάστοτε παράμετρος, και τοποθετεί έναν δείκτη εκεί. Δημιουργεί ένα προσωρινό αντικείμενο τύπου GList μέσω της gtk_list_item_new_with_label, στην οποία περνάει η μεταβλητή της ασύρματης παραμέτρου. Το νέο list item τοποθετείται στη θέση που δείχνει ο Pointer, και μέσω της gtk_widget_show εμφανίζεται.

Με παρόμοιο σχεδόν τρόπο λειτουργούν και οι συναρτήσεις Measurements(), Thresholds(), Losses(), show_description() και show_cur_value().

5.5.4 Αποτελέσματα

Στην υποενότητα αυτή θα δούμε τα αποτελέσματα της εκτέλεσης του λογισμικού που αναπτύχθηκε, ακολουθούμενα από τις κατάλληλες επεξηγήσεις και σχόλια για τις λειτουργίες που αυτό επιτελεί.

5.5.4.1 Λήψη Παραμέτρων Ασύρματης Κάρτας

Η εικόνα 10 δείχνει τα αποτελέσματα από την εκτέλεση του προγράμματος iwconfig, όπως αυτό τροποποιήθηκε και προσαρμόστηκε για τις ανάγκες του λογισμικού που εξετάζουμε.

```
wlan0 MAC PROTOCOL: IEEE 802.11g
ESSID: AP-307
MODE: Managed
FREQUENCY :2432000000.000000
ACCESS POINT :00:16:B6:D9:27:54
BITRATE :54 Mb/s
TX POWER :20 dBm
SENSITIVITY :-121
RTS THRESHOLD :2347
FRAGMENTATION THRESHOLD :2346
POWER MANAGEMENT :off
QUALITY: Quality:56/100 Signal level:-60 dBm Noise level:-96 dBm
RX INVALID NWID :0
RX INVALID CRYPT :0
RX INVALID FRAG :0
TX EXCESSIVE RETRIES :0
INVALID MISC :0
MISSED BEACON :0
```

Εικόνα 10. Λήψη Παραμέτρων Ασύρματης Κάρτας Δικτύου

Όπως βλέπουμε, εμφανίζονται στοιχεία για το πρωτόκολλο, την ESSID, τον τύπο λειτουργίας, τη συχνότητα, τη διεύθυνση του Access Point και ούτω καθ' εξής. Επειδή, όπως είπαμε, θέλουμε στο GUI να εμφανίζονται όλα τα πεδία, ακόμα και όταν αυτά δεν υπάρχουν, το πρόγραμμα εξετάζει συγκεκριμένες συνθήκες και αποδίδει τις κατάλληλες τιμές στις παραμέτρους. Ως αποτέλεσμα, στην περίπτωση που μία οι περισσότερες παράμετροι δεν έχουν τιμή, η έξοδος του προγράμματος είναι αυτή που φαίνεται στην εικόνα 11, η οποία εμφανίζει το αποτέλεσμα από τη δοκιμαστική εφαρμογή του iwconfig στη διεπαφή του Ethernet η οποία εξ' ορισμού δεν έχει wireless extensions και συνεπώς όλες οι εμφανιζόμενες παράμετροι είναι μηδενικές:

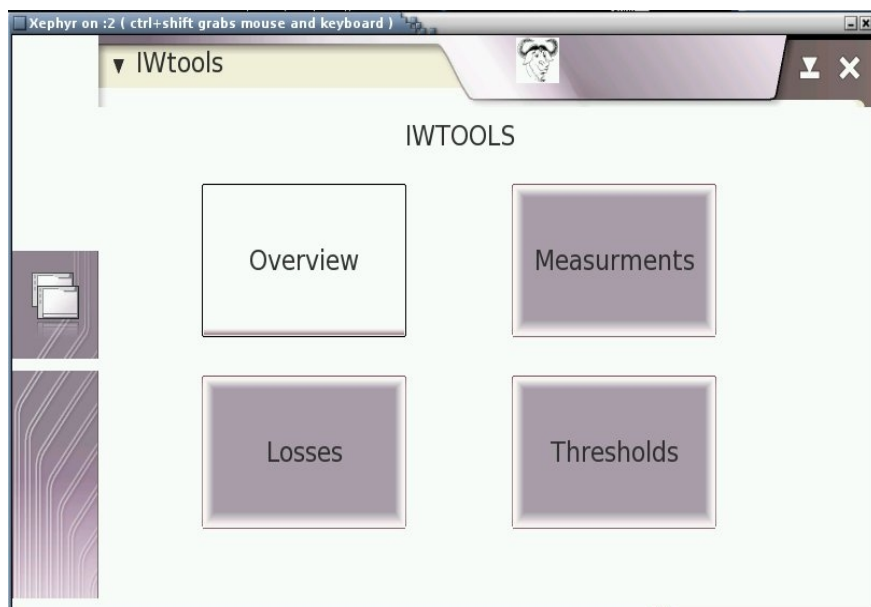
```
eth0 MAC PROTOCOL: None
ESSID: None
MODE: None
FREQUENCY :0.000000
ACCESS POINT :None
BITRATE :Null
TX POWER :Null
SENSITIVITY :0
RTS THRESHOLD :None
FRAGMENTATION THRESHOLD :None
POWER MANAGEMENT :On
QUALITY: Null
RX INVALID NWID :0
RX INVALID CRYPT :0
RX INVALID FRAG :0
TX EXCESSIVE RETRIES :0
INVALID MISC :0
MISSED BEACON :0
```

Εικόνα 11. Έξοδος του iwconfig όταν λείπουν τιμές από τις παραμέτρους

5.5.4.2 Εμφάνιση των Παραμέτρων στη Γραφική Διεπαφή Χρήστη

Το πρόγραμμα που δημιουργήθηκε για την εμφάνιση των παραμέτρων της ασύρματης πρόσβασης, έχει ως αρχικό το παράθυρο που βλέπουμε στην εικόνα 12. Το αρχικό παράθυρο σχεδιάστηκε έτσι ώστε να έχει μόνο τις απολύτως βασικές λειτουργίες και να είναι όσο το δυνατόν πιο φιλικό στο χρήστη.

Οι περισσότεροι χρήστες εξετάζουν ένα πρόγραμμα ξεκινώντας από πάνω αριστερά και καταλήγοντας κάτω δεξιά (όπως κατά την ανάγνωση). Για το λόγο αυτό, το κουμπί το οποίο εμφανίζει τις μετρήσεις και παραμέτρους που είναι πιο χρήσιμες και κατανοητές για έναν απλό, μη εξειδικευμένο χρήστη, τοποθετήθηκε πάνω δεξιά και ονομάστηκε «OVERVIEW». Με το ίδιο σκεπτικό, τοποθετήθηκε το κουμπί «MEASUREMENTS» πάνω δεξιά, το οποίο εμφανίζει τις σχετικές με την ισχύ σήματος, θορύβου και άλλων στατιστικών, παραμέτρους.



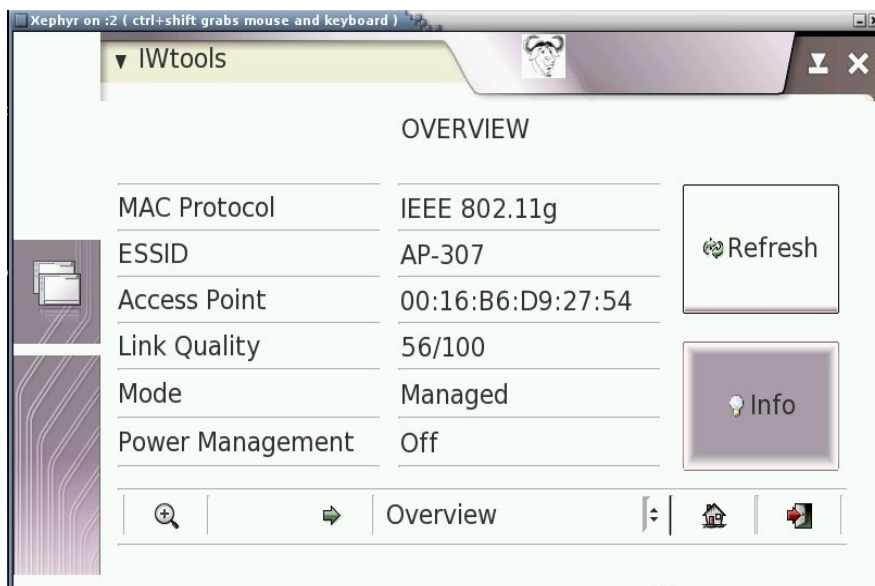
Εικόνα 12. Αρχικό Παράθυρο του GUI

Το κουμπί που δημιουργεί το παράθυρο «LOSSES», το οποίο παρουσιάζει τον αριθμό των πακέτων που χάθηκαν για τον εκάστοτε λόγο τοποθετήθηκε κάτω αριστερά, τρίτο σε σειρά προτίμησης για έναν ανειδίκευτο χρήστη. Και τελευταίο, με τις ίσως λιγότερο κατανοητές και χρήσιμες κατά την απλή, καθημερινή χρήση ενός Internet Tablet κουμπί, τοποθετήθηκε εκείνο

που εμφανίζει τα προσδιορισμένα κατώφλια της συσκευής («THRESHOLDS»).

Η επιλογή ενός από τα κουμπιά που βλέπουμε, συνεπάγεται την κλήση των συναρτήσεων `on_button_clicked()` που δημιουργήθηκαν στο `callbacks.c`, μέσω των οποίων δημιουργείται το παράθυρο που επέλεξε ο χρήστης να ανοίξει. Πιο συγκεκριμένα, η συνάρτηση `on_button_clicked()` κατασκευάζει το απαιτούμενο παράθυρο και το εμφανίζει. Παράλληλα, η επιλογή τερματισμού του προγράμματος, μέσω του κουμπιού «Close» (πάνω δεξιά), καλεί τη συνάρτηση `on_window0_destroy()`, η οποία καταστρέφει το παράθυρο και τερματίζει το πρόγραμμα.

Στην εικόνα 13 βλέπουμε το αποτέλεσμα από την ενεργοποίηση του πρώτου κουμπιού του αρχικού παραθύρου, το οποίο δημιουργεί και εμφανίζει το `window1` «OVERVIEW». Όπως βλέπουμε, στο παράθυρο αυτό, εμφανίζονται εκείνα τα στοιχεία και οι παράμετροι οι οποίες είναι οι περισσότερο χρησιμοποιούμενες από έναν απλό, μη εξειδικευμένο χρήστη, όπως το MAC πρωτόκολλο, η διεύθυνση του Access Point, η ESSID κ.λπ.



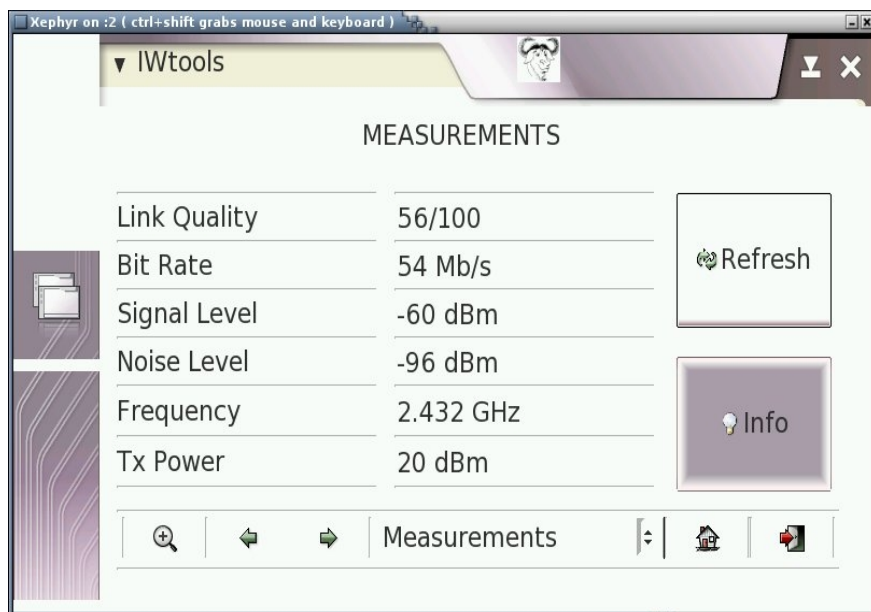
Εικόνα 13. Παράθυρο "OVERVIEW"

Κάθε ένα από τα αντικείμενα κειμενικής μορφής, όπως ο τίτλος του παραθύρου και οι ονομασίες των παραμέτρων, δημιουργήθηκε μέσω των `Gtk Labels` στα οποία ως παράμετρος πέρασε σε `string` μορφή το αντίστοιχο κείμενο. Στο χαμηλότερο μέρος του παραθύρου τοποθετήθηκε ένα `toolbar` το οποίο περιέχει κουμπιά για πλοήγηση στο και έξοδο από το σύστημα, καθώς και ένα `Menu` μέσω του οποίου ο χρήστης μπορεί να επιλέξει το επόμενο

παράθυρο που επιθυμεί να δει χωρίς να χρειάζεται να επιστρέφει στο Home ή να φορτώνει σειριακά τα παράθυρα μέσω των βελών forward και back.

Παράλληλα, οι μετρήσεις της ασύρματης κάρτας εμφανίζονται ως List Items στα αντίστοιχα List Widgets. Όταν φορτώνεται το παράθυρο ή πατιέται το κουμπί «Refresh», καλείται το κατάλληλο callback (π.χ. `on_button_1_clicked()` για την πρώτη περίπτωση και `on_button_refresh1_clicked` για τη δεύτερη) το οποίο καλεί τη συνάρτηση `overview()` την οποία εξηγήσαμε πιο πάνω.

Σαν παράμετρο ονόματος («label») του νέου list item, όπως είπαμε, μπορούμε να περάσουμε όποιο string ή char επιθυμούμε. Στην προκειμένη περίπτωση, για το πρώτο listitem (MAC Protocol), περάσαμε την μεταβλητή `MAC_PROTOCOL` η οποία μέσω της συνάρτησης `IWCONFIG_MAC()` που δημιουργήθηκε στο `iwconfig.c` λαμβάνει την αντίστοιχη τιμή και την εμφανίζει στο list item. Κατά αντιστοιχία, καλούνται και δημιουργούνται τα υπόλοιπα list items του παραθύρου.



Εικόνα 14. Παράθυρο "MEASUREMENTS"

Στην εικόνα 14 βλέπουμε το παράθυρο «MEASUREMENTS» το οποίο εμφανίζει τις παραμέτρους που έχουν να κάνουν με την ποιότητα της σύνδεσης. Όπως μπορούμε να παρατηρήσουμε, στο παράθυρο αυτό, το toolbar περιέχει και τα δύο βέλη πλοήγησης, αντίθετα με το προηγούμενό του («OVERVIEW»). Γενικά, τα «ακριανά» παράθυρα, όπως το πρώτο («OVERVIEW») και το τελευταίο «THRESHOLDS» δεν περιλαμβάνουν τα κουμπιά «forward» και «back» για προφανείς λόγους.

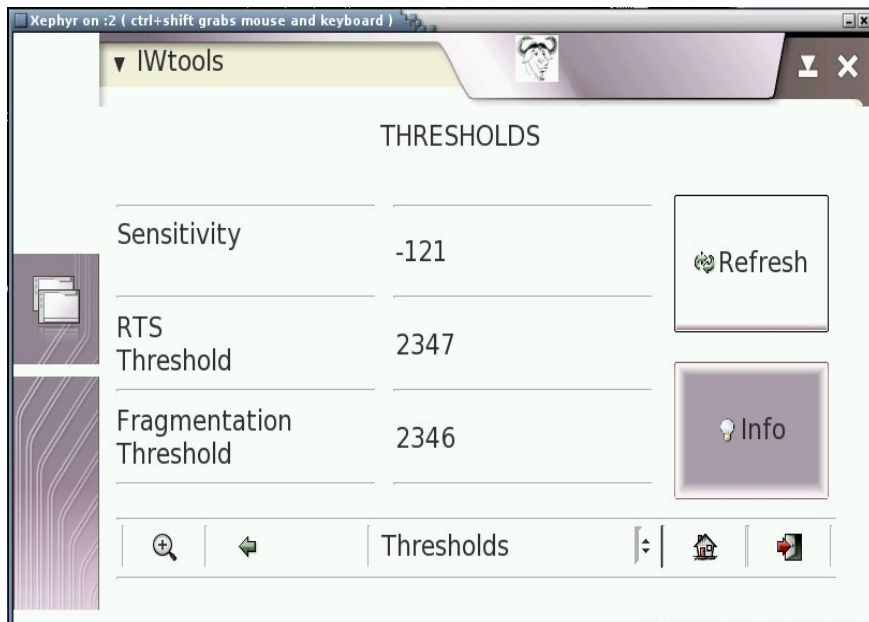
Παρατηρούμε επίσης, ότι έχει αλλάξει και το αρχικό menu item του Menu του toolbar και πλέον δείχνει την επιλογή «Measurements». Παρόλο που τα στοιχεία του μενού δεν αλλάζουν, σε κάθε παράθυρο αλλάζει η θέση του αρχικού item για λόγους αισθητικής.



Εικόνα 15. Παράθυρο "LOSSES"

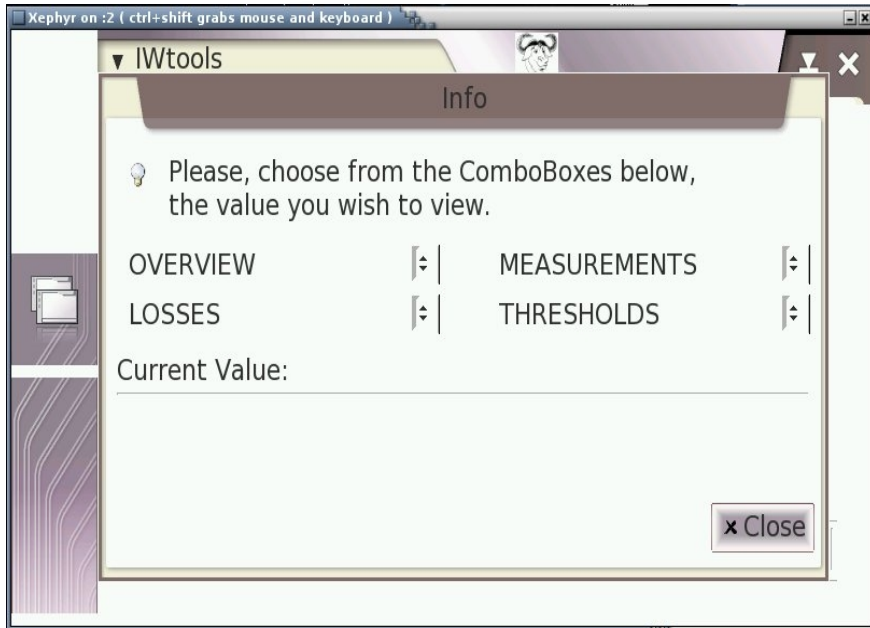
Συνεχίζοντας την περιγραφή των κουμπιών και των λειτουργιών τους, περνάμε στο toolbar button «Home». Με το πάτημα του κουμπιού αυτού, καλείται το callback `on_toolbarbutton_home_clicked()` η οποία: (1) Δημιουργεί ένα προσωρινό widget τύπου pointer («hidewin»), (2) μέσω της `lookup_widget()` το τοποθετεί στο τρέχον παράθυρο και (3) καλεί την συνάρτηση `gtk_widget_hide()` περνώντας της ως παράμετρο το pointer. Με τον τρόπο αυτό, «κρύβεται» το τρέχον παράθυρο και στην οθόνη παραμένει μόνο το αρχικό.

Με το τελευταίο κουμπί του toolbar τα πράγματα είναι πιο απλά. Το κουμπί αυτό τερματίζει το πρόγραμμα, οπότε με την ενεργοποίησή του καλείται απλά η συνάρτηση `gtk_main_quit()`.



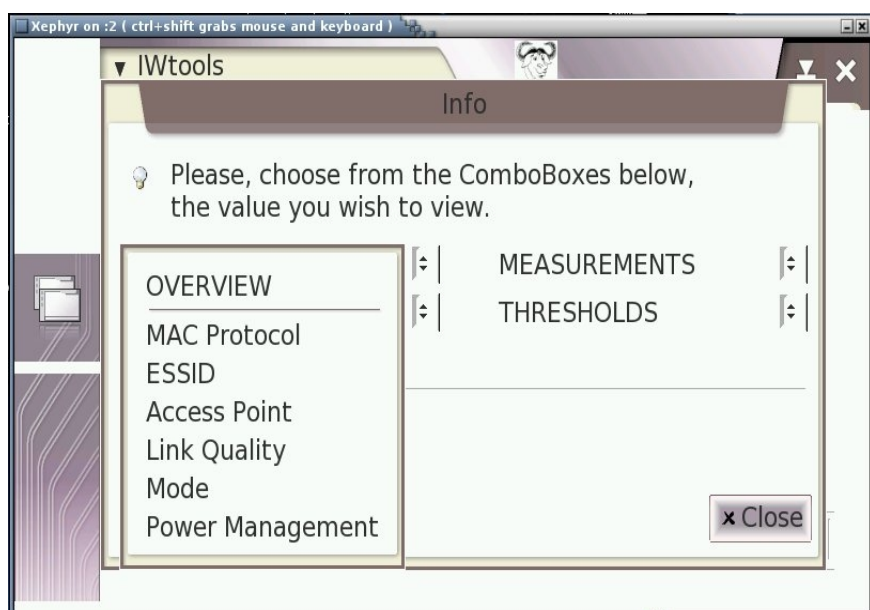
Εικόνα 16. Παράθυρο "THRESHOLDS"

Το κουμπί «Refresh», όπως δηλώνει και το όνομά του, χρησιμοποιείται για την ανανέωση/ ενημέρωση των στοιχείων. Η ιδέα ήταν ότι με την ενεργοποίηση του κουμπιού αυτού, και μέσω των συναρτήσεων `lookup_widget` και `gtk_widget_destroy`, όλα τα `listitems` που περιέχονται στα `GtkLists` θα καταστρέφονταν και στη συνέχεια μέσω της `gtk_list_item_new_with_label()` θα δημιουργούνταν νέα `listitems` με τις νέες πλέον τιμές που ελήφθησαν από τις συναρτήσεις της `display_info()`. Επειδή όμως όλες οι συναρτήσεις που παρέχονται από το `Gtk+ Toolkit` για να κρύψουν ή να καταστρέψουν ένα `Gtk Item` αρνήθηκαν να λειτουργήσουν, το κουμπί «Refresh» πλέον απλά κρύβει το τρέχον παράθυρο, το ξαναδημιουργεί με τα νέα στοιχεία και το εμφανίζει.



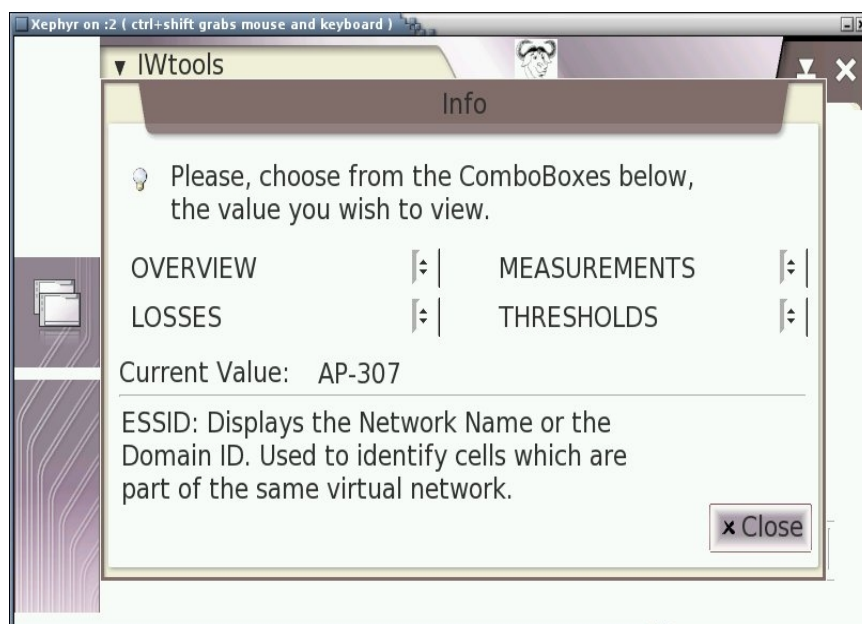
Εικόνα 17. Παράθυρο "Info"

Με την επιλογή του κουμπιού «Info» σε ένα από τα προηγούμενα παράθυρα, δημιουργείται και εμφανίζεται το παράθυρο «Info», σκοπός του οποίου είναι η εμφάνιση της περιγραφής κάθε παραμέτρου καθώς και της τρέχουσας τιμής της. Το παράθυρο αυτό σχεδιάστηκε να είναι μικρότερο από τα άλλα, και να λειτουργεί σαν PopUp. Περιλαμβάνει 4 Menus μέσω των οποίων ο χρήστης μπορεί να επιλέξει την παράμετρο της οποίας την περιγραφή επιθυμεί να διαβάσει, όπως δείχνει και η εικόνα 18:



Εικόνα 18. Επιλογή Παραμέτρου για Θέαση Σχετικής Πληροφορίας

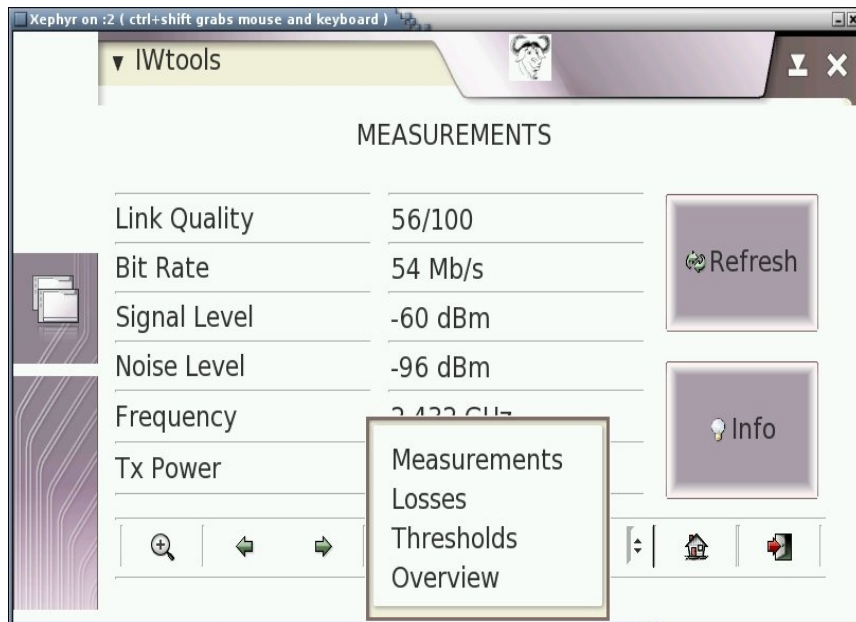
Όταν ενεργοποιείται κάποια από τις επιλογές του menu, καλείται το callback `on_menuitem_activate` η οποία καλεί τις συναρτήσεις `show_description()` και `show_cur_value()`. Όπως βλέπουμε και στην εικόνα 19, η επιλογή του «ESSID» από το menu «OVERVIEW» συνεπάγεται την εμφάνιση της τρέχουσας τιμής της παραμέτρου «ESSID» καθώς και μία σύντομη περιγραφή της.



Εικόνα 19. Εμφάνιση Τρέχουσας Τιμής Παραμέτρου και Σχετικής Περιγραφής της

Προκειμένου με κάθε επιλογή να εμφανίζονται οι νέες τιμές παραμέτρων και τα ενημερωμένα κείμενα περιγραφής, πριν από την κλήση των συναρτήσεων δημιουργίας των List Widgets καλούνται οι συναρτήσεις απόκρυψης και επαναδημιουργίας του παραθύρου, όπως περιγράφηκαν πιο πάνω. Έτσι, πριν φορτωθούν τα νέα στοιχεία, το παράθυρο «καθαρίζει» ώστε να μπορεί να δεχτεί τα επόμενα items.

Τέλος, η επιλογή του κουμπιού «Close» του υπό εξέταση παραθύρου, καλεί τη συνάρτηση `gtk_widget_destroy()` η οποία το καταστρέφει. Ως αποτέλεσμα, κλείνει το «Info» window, και παραμένει το παράθυρο από το οποίο αυτό άνοιξε εξ' αρχής (εικόνα 20).



Εικόνα 20. Menu Πλοήγησης Στο Σύστημα

Τέλος, η τελευταία λειτουργία που υλοποιεί το σύστημα είναι το Menu πλοήγησης. Κάθε παράθυρο, όπως ήδη είδαμε, περιέχει ένα menu μέσω του οποίου ο χρήστης μπορεί να ανοίξει με μη σειριακό τρόπο, ένα από τα υπόλοιπα παράθυρα. Η λειτουργία των menu items είναι απλή: όταν ένα από αυτά επιλέγεται, φορτώνεται το αντίστοιχο παράθυρο και μετά αποκρύπτεται το τρέχον. Με τον τρόπο αυτόν, αποφεύγουμε να έχουμε ανοιχτά πάνω από ένα παράθυρα, πράγμα που μπορεί να επιβραδύνει ή και να τερματίσει το σύστημα.

Όλοι οι κώδικες για την υλοποίηση του GUI και του iwconfig, ευρίσκονται στο Παράρτημα 1, μαζί με τα κατάλληλα σχόλια.

ΣΥΝΤΟΜΟΓΡΑΦΙΕΣ

A

0G	Zero Generation
1G	First Generation
2G	Second Generation
3G	Third Generation
3GPP	3 rd Generation Partnership Project
4G	Fourth Generation
AAA	Authentication, Authorization and Accounting
ABC	Always Best Connected
ABE	Alternative Best – Effort
ACK	Acknowledgment Code
ACS	Admission Control Service
ACTS	Advanced Communication Technologies and Services
AM	Amplitude Modulation
AMPS	Advanced Mobile Phone Service
API	<i>Application Programming Interface</i>
AR	Access Router
ARIB	Association of Radio Industries and Businesses
ARP	Autoradiopuhelin (Car Radio Phone)
AS	Access Slot
AuC	Authentication Centre

B

B3G	Beyond the Third Generation
BAN	Basic Area Network
BB	Bandwidth Broker
BER	Bit Error Rate
BRAN	Broadband Radio Access Network
BS	Base Station
BSC	Base Station Controller
BSS	Base Station Subsystem
BTS	Base Transceiver Station

C

CAMEL	Customized Applications for Mobile network Enhanced Logic
CBR	Constant Bit Rate
CCH	Common Channel

CDMA	Code Division Multiple Access
CEPT	Conférence Européenne des Postes et des Télécommunications
CG	Charging Gateway
CI	Cell Identifier
CN	Core Network
CRE	Composite Radio Environment
CREMS	Cognitive Reconfigurable Equipments Management System
CRNC	Controlling RNC
CS	Circuit Switched
CSCN	Circuit Switched Core Network
CSD	Circuit-switched Data
CSR	Central Software Repository

D

D-AMPS	Digital AMPS
DNS	Domain Name Server
DRNS	Drift RNS
DSA	Dynamic Spectrum Allocation
DVB	Digital Video Broadcasting

E

EDGE	Enhanced Data Rate for Global Evolution
EIR	Equipment Identity Register
ETSI	European Telecommunications Standards Institute

F

FCC	Federal Communications Commission
FDD	Frequency Division Duplex
FDMA	Frequency Division Multiple Access
FER	Frame Error Rate
FES	Fixed Earth Station
FM	Frequency Modulation

G

GERAN	GSM EDGE Radio Access Network
GGSN	Gateway GPRS Support Node
GMSK	Gaussian Minimum Shift Keying
GPRS	General Packet Radio Service
GPS	Global Positioning System
GRM	Gateway Reconfiguration Manager
GSM	Global System for Mobile communications
GTP	GPRS Tunnelling Protocol

GUI	<i>Graphical User Interface</i>
GUP	Generic User Profile
GW	Gateway

H

HAN	Home Area Network
HLR	Home Location Register
HO	Handover
HRM	Home Reconfiguration Manager
HSCSD	High Speed Circuit Switched Data
HSPA	<i>High Speed Packet Access</i>
HSS	Home Subscriber Server
HTML	HyperText Markup Language

I

IMS	IP Multimedia Subsystem
IMSI	International Mobile Subscriber Identity
IMT – 2000	International Mobile Telecommunications System – 2000
IN	Intelligent Network
IP	Internet Protocol
IPv4	Internet Protocol Version 4
IPv6	Internet Protocol Version 6
ISDN	Integrated Services Digital Network
ISP	Internet Service Provider

L

L2TP	Layer Two Tunneling Protocol
LA	Location Area
LBS	Location Based Services
LTE	Long Term Evolution
LU	Location Update

M

MAC	Medium Access Control
MAP	Mobile Application Part
MAP	<i>Mobility Anchor Point</i>
MBMS	Multimedia Broadcast Multicast Service
MIMO	Multiple Input Multiple Output
MMI	Man Machine Interface
MMS	Multimedia Messaging Service

MRC	Maximum Ratio Combining
MS	Mobile Station
MSC	Mobile Switching Centre
MT	Mobile Terminal
MTA	Mobile Telephone system A
MTB	Mobile Telephone system B

N

NIAM	Network Interface Adaptation Module
NMT	Nordic Mobile Telephone System
NP	Network Provider
NRT	Non-Real Time
NSS	Network Switching Subsystem
NTT	Nippon Telephone and Telegraph

O

OFDMA	Orthogonal Frequency Division Multiple Access
OSI	<i>Open Systems Interconnection</i> (Basic Reference Model)
OSS	Operation Subsystem
OTDOA	Observed Time Difference of Arrival

P

P – TMSI	Packet – Temporary Mobile Subscriber Identity
PAN	Personal Area Network
PDA	Personal Digital Assistant
PDC	Personal Digital Cellular
PDN	Packet Data Network
PDP	Packet Data Protocol
PDP	Policy Decision Point
PHY	Physical Layer
PPP	Point – to – Point Protocol
PRM	Proxy Reconfiguration Manager
PS	Packet Switched
PSCN	Packet Switched Core Network
PSTN	Public Switched Telephone Network

Q

QoS	Quality Of Service
QPSK	<i>Quadrature Phase – Shift Keying</i>

R

R3	UMTS Release 3
----	----------------

R4	UMTS Release 4
R5	UMTS Release 5
R6	UMTS Release 6
R7	UMTS Release 7
R99	UMTS Release 99
RA	Routing Area
RAI	Routing Area Identifier
RAT	Radio Access Technology
RNC	Radio Network Controller
RNS	Radio Network Subsystems
RRC	Radio Resource Control
RSCP	Received Signal Code Power
RSSI	Received Signal Strength Indicator
RSVP	Resource Reservation Protocol
RT	Real Time

S

SAE	System Architecture Evolution
SBM	Subnetwork Bandwidth Manager
SDB	Software Download Box
SDMA	Space Division Multiple Access
SGSN	Serving GPRS Support Node
SHO	Soft Handover
SIM	Subscriber Identity Module
SIP	Session Initiated Protocol
SIR	Signal to Interference Ratio
SLA	Service Level Agreement
SLS	Service Level Specifications
SMS	Short Message Service
SNMP	Simple Network Management Protocol
SNR	Signal to Noise Ratio
SRM	Serving Reconfiguration Manager
SRNS	Serving RNS
SS	Supplementary Services

T

TA	Terminal Adaptor
TACS	Total Access Communications System
TDD	Time Division Duplex
TDMA	Time Division Multiple Access
TE	Terminal Equipment
TIA	Telecommunications Industry Association
TMN	Telecommunications Management Network

U

UE	User Equipment
UMTS	Universal Mobile Telecommunications System
UTRA	Universal Terrestrial Radio Access
UTRAN	UMTS Terrestrial Radio Access Network

V

VBR	Variable Bit Rate
VLR	Visitor Location Register
VPN	Virtual Private Network

W

W – CDMA	Wideband CDMA
WAP	Wireless Application Protocol
WLAN	Wireless Local Area Network

X

XML	Extensible Markup Language
-----	----------------------------

ΒΙΒΛΙΟΓΡΑΦΙΑ

A. Κανάτας, Φ. Κωνσταντίνου, Γ. Πάντος, «Συστήματα Κινητών Επικοινωνιών», *A. Παπασωτηρίου*, 2008

B. Nair, «3G and beyond – The future of wireless technologies», 2001

B. Xing and N. Venkatasubramanian, «Multi – Constraint Dynamic Access Selection in Always Best Connected Networks»

E. Adamopoulou, K. Demestichas, A. Koutsorodi, M. Theologou, « Intelligent Access Network Selection in Heterogeneous Networks : Simulation Results», *IEEE Wireless Communications*, 2005

G. Fodor, A. Furuskar, J. Lundsjo, « On Access Selection Techniques in Always Best Connected Networks»

J. De Vriendt, P. Laine, C. Lerouge, X. Xu, «Mobile Network Evolution: A Revolution on the Move», *IEEE Communications Magazine*, 2002

J. Korhonen, «Introduction to 3G Mobile Communications», *2nd edition*, *Artech House mobile communications series*, 2003

M. Dillinger, N. Alonistioti, N . Olaziregi and T. Wiebke, «Network Functions Supporting Reconfiguration in B3G environments»

M. Rubaiyat Kibria, Vinod Mirchandani, and Abbas Jamalipour, « A Consolidated Architecture for 4G/B3G Networks», *The University of Sydney*, 2006

O. Ormond, Dr. Gabriel – Miro Muntean , Dr. John Murphy, « Network Selection Strategy in Heterogeneous Wireless Networks»

P. Demestichas, N. Koutsouris, G. Koundourakis, K. Tsagkaris, A. Oikonomou, V. Stavroulaki, L. Papadopoulou, M.E. Theologou, «Management of Networks and Services in a Composite Radio Context», *IEEE Wireless Communications*, 2004

R. Lloyd – Evans, «QoS in Integrated 3G Networks», *Artech House mobile communications series*, 2002

R. State, «Wireless Terminal Management Architectures», *The MADYNES Research Team*

S. G. Glisic, «Advanced Wireless Networks : 4G Technologies», *John Wiley & Sons Ltd*, 2006

S. N. P. Van Cauwenberge, « Study of soft handover in UMTS», *Master Thesis submitted at Danmarks Tekniske Universitet – COM department*, 2003

T. B. Zahariadis, K. G. Vaxevanakis, C. P. Tsantilas, N. A. Zervos and N. A. Nikolaou, « Global Roaming in Next-Generation Networks», *IEEE Communications Magazine*, 2002

V. Stavroulaki, A. Katidiotis, G. Dimitrakopoulos, P. Demestichas, S. Buljore, “Negotiation and Selection of Equipment Reconfigurations in Beyond 3G Systems”, *Proc. IEEE PIMRC*, 2005

V. Stavroulaki, P. Demestichas, A. Katidiotis, D. Petromanolakis, « Evolution in equipment management concepts: from reconfigurable to cognitive wireless terminals»

<http://maemo.org/>

[http://en.wikipedia.org/wiki/Maemo_\(operating_system\)](http://en.wikipedia.org/wiki/Maemo_(operating_system))

http://www.hpl.hp.com/personal/Jean_Tourrilhes/Linux/Tools.html

<http://library.gnome.org/devel/gtk/stable/index.html>

<http://dbus.freedesktop.org/doc/dbus-tutorial.html>

<http://www.freedesktop.org/wiki/IntroductionToDBus>

<http://dbus.freedesktop.org/doc/dbus-glib/index.html>

<http://www-128.ibm.com/developerworks/linux/library/l-dbus.html?ca=dgr-Inxw95D-BUS>

ΠΑΡΑΡΤΗΜΑ Ι

lwconfig.c

```
#include "iwlib.h"          /* Header */

/***** STATHERES *****/

//Kwdikes lathwn gia na paizei to "args"

#define IWERR_ARG_NUM      -2
#define IWERR_ARG_TYPE    -3
#define IWERR_ARG_SIZE    -4
#define IWERR_ARG_CONFLICT -5
#define IWERR_SET_EXT     -6
#define IWERR_GET_EXT     -7

// -----Lhpsh parametrwn-----

static int
get_info(int          skfd,
         char *       ifname,
         struct wireless_info * info)
{
    struct iwreq      wrq;

    memset((char *) info, 0, sizeof(struct wireless_info));

    /* Lhpsh vasikwn plhroforiwn */
```

```

if(iw_get_basic_config(skfd, ifname, &(info->b)) < 0)
{
    /* If no wireless name : no wireless extensions */
    /* But let's check if the interface exists at all */
    struct ifreq ifr;

    strncpy(ifr.ifr_name, ifname, IFNAMSIZ);
    if(ioctl(skfd, SIOCGIFFLAGS, &ifr) < 0)
        return(-ENODEV);
    else
        return(-ENOTSUP);
}

/* Lhpsh range */
if(iw_get_range_info(skfd, ifname, &(info->range)) >= 0)
    info->has_range = 1;

/* Lhpsh AP address */
if(iw_get_ext(skfd, ifname, SIOCGIWAP, &wrq) >= 0)
{
    info->has_ap_addr = 1;
    memcpy(&(info->ap_addr), &(wrq.u.ap_addr), sizeof (sockaddr));
}

/* Lhpsh bit rate */
if(iw_get_ext(skfd, ifname, SIOCGIWRATE, &wrq) >= 0)
{
    info->has_bitrate = 1;
    memcpy(&(info->bitrate), &(wrq.u.bitrate), sizeof(iwparam));
}

```

```

/* Lhpsh Power Management ruthmisewn */

wrq.u.power.flags = 0;

if(iw_get_ext(skfd, ifname, SIOCGIWPOWER, &wrq) >= 0)
{
    info->has_power = 1;
    memcpy(&(info->power), &(wrq.u.power), sizeof(iwparam));
}

/* Lhpsh statistikwn */

if(iw_get_stats(skfd, ifname, &(info->stats),
               &info->range, info->has_range) >= 0)
{
    info->has_stats = 1;
}

#ifdef WE_ESSENTIAL

/* Lhpsh NickName */

wrq.u.essid.pointer = (caddr_t) info->nickname;
wrq.u.essid.length = IW_ESSID_MAX_SIZE + 1;
wrq.u.essid.flags = 0;

if(iw_get_ext(skfd, ifname, SIOCGIWICKN, &wrq) >= 0)
    if(wrq.u.data.length > 1)
        info->has_nickname = 1;

if((info->has_range) && (info->range.we_version_compiled > 9))
{
    /* Lhpsh Transmit Power */

    if(iw_get_ext(skfd, ifname, SIOCGIWTXPOW, &wrq) >= 0)
        {

```

```

        info->has_txpower = 1;
        memcpy(&(info->txpower), &(wrq.u.txpower), sizeof(iwparam));
    }
}

/* Lhpsh sensitivity */
if(iw_get_ext(skfd, ifname, SIOCGIWSENS, &wrq) >= 0)
{
    info->has_sens = 1;
    memcpy(&(info->sens), &(wrq.u.sens), sizeof(iwparam));
}

if((info->has_range) && (info->range.we_version_compiled > 10))
{
    /* Get retry limit/lifetime */
    if(iw_get_ext(skfd, ifname, SIOCGIWRETRY, &wrq) >= 0)
    {
        info->has_retry = 1;
        memcpy(&(info->retry), &(wrq.u.retry), sizeof(iwparam));
    }
}

/* Lhpsh RTS threshold */
if(iw_get_ext(skfd, ifname, SIOCGIWRTS, &wrq) >= 0)
{
    info->has_rts = 1;
    memcpy(&(info->rts), &(wrq.u.rts), sizeof(iwparam));
}

/* Lhpsh fragmentation threshold */

```

```

if(iw_get_ext(skfd, ifname, SIOCGIWFRAG, &wrq) >= 0)
{
    info->has_frag = 1;
    memcpy(&(info->frag), &(wrq.u.frag), sizeof(iwparam));
}
#endif /* WE_ESSENTIAL */

return(0);
}

//-----Emfanish kai Diathrhsh parametrwn-----

static void
display_info(struct wireless_info *    info,
             char *                    ifname)
{

printf("\n\n=====
=====\\n\\n");

/* Onoma interface */
printf("%s ", ifname); //Gia na vlepw apo poio interface lamvanei

void IWCONFIG_MAC();
{
const char *MAC_PROTOCOL;
if(info->b.has_mode)
{

```

```

        MAC_PROTOCOL = info->b.name;
    }
else
    {
        MAC_PROTOCOL = "None";
    }
printf("MAC PROTOCOL: %s\n\n", MAC_PROTOCOL);
} //iwconfig_mac()

```

```

void IWCONFIG_ESSID();

```

```

{
const char *ESSID;

if (info->b.has_essid)
{
if (info->b.essid_on)
{
    ESSID = info->b.essid;
}
else
{
    ESSID = "Off/ Any";
}
} //end 1st if
else
{
    ESSID = "None";
}
printf("ESSID: %s\n\n", ESSID);

```

```
} //iwconfig_essid()
```

```
void IWCONFIG_MODE();
```

```
{
```

```
const char *MODE;
```

```
if(info->b.has_mode)
```

```
{
```

```
MODE = iw_operation_mode[info->b.mode];
```

```
}
```

```
else
```

```
{
```

```
MODE = "None";
```

```
}
```

```
printf("MODE: %s\n\n", MODE);
```

```
} //iwconfig_mode()
```

```
void IWCONFIG_FREQUENCY();
```

```
{
```

```
double FREQUENCY;
```

```
double freq = info->b.freq;
```

```
int channel = -1;
```

```
if(info->b.has_freq)
```

```
{
```

```
if(info->has_range && (freq < KILO))
```

```
{channel = iw_channel_to_freq((int) freq, &freq, &info->range);}
```



```

    FREQUENCY = freq;
}
else
{
    FREQUENCY = 0;
}
printf("FREQUENCY :%f\n\n", FREQUENCY);
} //iwconfig_frequency()

void IWCONFIG_AP();
{
const char *AP;
char ap_buffer[128];

if(info->has_ap_addr)
{
    AP = iw_sawap_ntop(&info->ap_addr, ap_buffer);
}
else
{
    AP = "None";
}
printf("ACCESS POINT :%s\n\n", AP);
} //end iwconfig_ap()

void IWCONFIG_BITRATE();
{
const char *BITRATE;

```

```

char bitrate_buffer[128];

if(info->has_bitrate)
{
iw_print_bitrate(bitrate_buffer, sizeof(bitrate_buffer), info->bitrate.value);
BITRATE = bitrate_buffer;
}
else
{
BITRATE = "Null";
}
printf("BITRATE :%s\n\n", BITRATE);
} //iwconfig_bitrate()

void IWCONFIG_TXPOWER();
{
const char *TXPOWER;
char txpower_buffer[128];

if(info->has_txpower)
{
iw_print_txpower(txpower_buffer, sizeof(txpower_buffer), &info->txpower);
TXPOWER = txpower_buffer;
}
else
{
TXPOWER = "Null";
}
printf("TX POWER :%s\n\n", TXPOWER);
} // iwconfig_txpower

```

```

void IWCONFIG_SENSITIVITY();
{
int SENSITIVITY;

if (info->has_sens)
{
SENSITIVITY = info->sens.value;
}
else
{
SENSITIVITY = 0;
}
printf("SENSITIVITY :%d\n\n", SENSITIVITY);
}

```

```

void IWCONFIG_RTS();
{
/*metatroph tou info->rts.value apo int se string wste na lamvanei ton idio tupou. */
/*To info->rts.value einai tupou int*/
char rts[128];
const char *RTS;

if (info->has_rts)
{
if (info->rts.disabled)
{
RTS = "Off";
}
}
}

```

```

else
{
    sprintf(rts, "%d", info->rts.value); //Metatroph tou rts apo int se string

    RTS = rts;
}
}
else
{
    RTS = "None";
}
printf("RTS THRESHOLD :%s\n\n", RTS);
} //iwconfig_rts()

void IWCONFIG_FRAGM();
{
    /*metatroph tou info->frag.value apo int se string wste na lamvanei ton idio tupo. */
    /*To info->frag.value einai tupou int*/
    char frag[128];
    const char *FRAG;

    if (info->has_frag)
    {
        if (info->frag.disabled)
        {
            FRAG = "Off";
        }
    }
    else
    {
        sprintf(frag, "%d", info->frag.value);
        FRAG = frag;
    }
}

```

```

    }
}
else
{
    FRAG = "None";
}
printf("FRAGMENTATION THRESHOLD :%s\n\n", FRAG);
} //iwconfig_frag()

void IWCONFIG_POWMANAG();
{
const char *POWMANAG;
const char *POWER_MODE;
const char *PERIOD;
char period_buffer[128];
char mode_buffer[128];

//De vazw if(info->has_power, epeidh an den exei, tote den uparxei suskeuh na deis tis
methseis, xairw polu.

    if (info->power.disabled)
    {
        POWMANAG = "Off";
        printf("POWER MANAGEMENT :%s\n\n", POWMANAG);
    }
else
    {
        if(info->power.flags & IW_POWER_TYPE)
        {
            iw_print_pm_value(period_buffer, sizeof(period_buffer),info->power.value, info-
            >power.flags, info->range.we_version_compiled);

            PERIOD = period_buffer; // epistrefei to time period se seconds.

```

```

    }

    iw_print_pm_mode(mode_buffer, sizeof(mode_buffer), info->power.flags);
    POWER_MODE = mode_buffer; // epistrefei to mode. den exw idea ti einai...
    POWMANAG = "On";
    printf("POWER MANAGEMENT  :%s\n\n", POWMANAG);
    }
} //iwconfig_powmanag

void IWCONFIG_LINKSTATS();

{

char stats_buffer[128];
const char *LINKSTATS;

if(info->has_stats)
{
    iw_print_stats(stats_buffer, sizeof(stats_buffer),
        &info->stats.qual, &info->range, info->has_range);
    LINKSTATS = stats_buffer;
}
else
{
    LINKSTATS = "Null";
}

printf("QUALITY: %s\n\n",    LINKSTATS);

} //iwconfig_linkstats()

void IWCONFIG_INVNWDID();

{

```

```

int INVNWID = info->stats.discard.nwid;

printf("RX INVALID NWID  :%d\n\n", INVNWID);

}    //iwconfig_invnwid

void IWCONFIG_INVCRYPT();

{

int INVCRYPT = info->stats.discard.code;

printf("RX INVALID CRYPT  :%d\n\n", INVCRYPT);

}    //iwconfig_invcrypt

void IWCONFIG_INVFRAG();

{

int INVFRAG = info->stats.discard.fragment;

printf("RX INVALID FRAG  :%d\n\n", INVFRAG);

}    //iwconfig_invfrag

void IWCONFIG_TXRET();

{

int TXRET = info->stats.discard.retries;

printf("TX EXCESSIVE RETRIES  :%d\n\n", TXRET);

}    //iwconfig_txret()

void IWCONFIG_INVMISC();

{

int INVMISC = info->stats.discard.misc;

printf("INVALID MISC  :%d\n\n", INVMISC);

}    // iwconfig_invmisc()

void IWCONFIG_BEACON();

{

```

```

int BEACON = info->stats.miss.beacon;

printf("MISSED BEACON  :%d\n\n", BEACON);

}    //iwconfig_beacon()
}    //end display_info()

// -----klhsh ths display_info()-----

static int
print_info(int      skfd,
           char * ifname,
           char * args[],
           int      count)
{
    struct wireless_info  info;
    int                   rc;

    /* Gia na mhn epistrefei "unused parameter" */
    args = args; count = count;

    rc = get_info(skfd, ifname, &info);
    display_info(&info, ifname);

    return 0;
}

// -----MAIN-----

int
main(intargc,
     char ** argv)
{
    argv = argv; //epeidh leei oti einai unused parameter, to dhlwnw na uparxei.

    int skfd;          /* generic raw socket desc. */

```



```
int goterr = 0;

/* Anoigei kanali sto NET kernel */
if((skfd = iw_sockets_open()) < 0)
{
    perror("socket");
    exit(-1);
}

argc = 1;
if(argc == 1)
iw_enum_devices(skfd, &print_info, NULL, 0);

/* Kleisimo socket */
iw_sockets_close(skfd);

return(goterr);
} //end main()
```

callbacks.c

```
#ifndef HAVE_CONFIG_H // Includes
# include <config.h>
#endif

#include <gtk/gtk.h>

#include "callbacks.h"
#include "interface.h"
#include "support.h"

//--Arxikopoihsh metavlhtwn gia callbacks parathurwn kai widgets ----
GtkWidget *window0;
GtkWidget *window1;
GtkWidget *window2;
GtkWidget *window3;
GtkWidget *window4;
GtkWidget *window5;
GtkWidget *optionmenu_info1;

void overview()
{
    const char *MAC_PROTOCOL = "IEEE 802.11g";
    const char *ESSID = "AP-307";
    const char *AP = "00:16:B6:D9:27:54";
    const char *QUALITY = "56/100";
    const char *MODE = "Managed";
    const char *POWMANAG = "Off";
    GtkWidget *listitem11; //Mac Protocol
    GtkWidget *list11;
    GtkWidget *listitem12; //ESSID
    GtkWidget *list12;
    GtkWidget *listitem13; //Access Point
    GtkWidget *list13;
    GtkWidget *listitem14; //Link Quality
    GtkWidget *list14;
    GtkWidget *listitem15; //Mode
    GtkWidget *list15;
    GtkWidget *listitem16; //Power Management
    GtkWidget *list16;

    GtkWidget *cur_list1 = lookup_widget(GTK_WIDGET(window1), "list11");
    listitem11 = gtk_list_item_new_with_label(MAC_PROTOCOL);
    gtk_container_add(GTK_CONTAINER(cur_list1), listitem11);
    gtk_widget_show(listitem11);

    GtkWidget *cur_list2 = lookup_widget(GTK_WIDGET(window1), "list12");
    listitem12 = gtk_list_item_new_with_label(ESSID);
    gtk_container_add(GTK_CONTAINER(cur_list2), listitem12);
    gtk_widget_show(listitem12);

    GtkWidget *cur_list3 = lookup_widget(GTK_WIDGET(window1), "list13");
    listitem13 = gtk_list_item_new_with_label(AP);
    gtk_container_add(GTK_CONTAINER(cur_list3), listitem13);
}
```

```

gtk_widget_show(listitem13);

GtkWidget *cur_list4 = lookup_widget(GTK_WIDGET(window1), "list14");
listitem14 = gtk_list_item_new_with_label(QUALITY);
gtk_container_add(GTK_CONTAINER(cur_list4), listitem14);
gtk_widget_show(listitem14);

GtkWidget *cur_list5 = lookup_widget(GTK_WIDGET(window1), "list15");
listitem15 = gtk_list_item_new_with_label(MODE);
gtk_container_add(GTK_CONTAINER(cur_list5), listitem15);
gtk_widget_show(listitem15);

GtkWidget *cur_list6 = lookup_widget(GTK_WIDGET(window1), "list16");
listitem16 = gtk_list_item_new_with_label(POWMANAG);
gtk_container_add(GTK_CONTAINER(cur_list6), listitem16);
gtk_widget_show(listitem16);
}

void measurements()
{
    const char *QUALITY = "56/100"; // Link Quality
    const char *BITRATE = "54 Mb/s"; //Bit Rate
    const char *SIGNAL = "-60 dBm"; // Signal Level
    const char *NOISE = "-96 dBm"; //Noise Level
    const char *FREQUENCY = "2.432 GHz"; //Frequency
    const char *TXPOWER = "20 dBm"; //Tx Power
    GtkWidget *listitem21;
    GtkWidget *list21;
    GtkWidget *listitem22;
    GtkWidget *list22;
    GtkWidget *listitem23;
    GtkWidget *list23;
    GtkWidget *listitem24;
    GtkWidget *list24;
    GtkWidget *listitem25;
    GtkWidget *list25;
    GtkWidget *listitem26;
    GtkWidget *list26;

    GtkWidget *cur_list1 = lookup_widget(GTK_WIDGET(window2), "list21");
    listitem21 = gtk_list_item_new_with_label(QUALITY);
    gtk_container_add(GTK_CONTAINER(cur_list1), listitem21);
    gtk_widget_show(listitem21);

    GtkWidget *cur_list2 = lookup_widget(GTK_WIDGET(window2), "list22");
    listitem22 = gtk_list_item_new_with_label(BITRATE);
    gtk_container_add(GTK_CONTAINER(cur_list2), listitem22);
    gtk_widget_show(listitem22);

    GtkWidget *cur_list3 = lookup_widget(GTK_WIDGET(window2), "list23");
    listitem23 = gtk_list_item_new_with_label(SIGNAL);
    gtk_container_add(GTK_CONTAINER(cur_list3), listitem23);
    gtk_widget_show(listitem23);

    GtkWidget *cur_list4 = lookup_widget(GTK_WIDGET(window2), "list24");
    listitem24 = gtk_list_item_new_with_label(NOISE);
    gtk_container_add(GTK_CONTAINER(cur_list4), listitem24);
    gtk_widget_show(listitem24);
}

```

```

GtkWidget *cur_list5 = lookup_widget(GTK_WIDGET(window2), "list25");
listitem25 = gtk_list_item_new_with_label(FREQUENCY);
gtk_container_add(GTK_CONTAINER(cur_list5), listitem25);
gtk_widget_show(listitem25);

GtkWidget *cur_list6 = lookup_widget(GTK_WIDGET(window2), "list26");
listitem26 = gtk_list_item_new_with_label(TXPOWER);
gtk_container_add(GTK_CONTAINER(cur_list6), listitem26);
gtk_widget_show(listitem26);
}

void losses()
{

const char *INVNWID = "0"; //Rx Invalid NWID
const char *INVCRYPT = "0"; //Rx Invalid Crypt
const char *INVFRAG = "0"; // Rx Invalid Fragn
const char *TXRET = "0"; //Tx Excessive Retries
const char *BEACON = "0"; //Missed Beacon
const char *INVMISC = "0"; //Invalid Misc
GtkWidget *listitem31;
GtkWidget *list31;
GtkWidget *listitem32;
GtkWidget *list32;
GtkWidget *listitem33;
GtkWidget *list33;
GtkWidget *listitem34;
GtkWidget *list34;
GtkWidget *listitem35;
GtkWidget *list35;
GtkWidget *listitem36;
GtkWidget *list36;

GtkWidget *cur_list1 = lookup_widget(GTK_WIDGET(window3), "list31");
listitem31 = gtk_list_item_new_with_label(INVNWID);
gtk_container_add(GTK_CONTAINER(cur_list1), listitem31);
gtk_widget_show(listitem31);

GtkWidget *cur_list2 = lookup_widget(GTK_WIDGET(window3), "list32");
listitem32 = gtk_list_item_new_with_label(INVCRYPT);
gtk_container_add(GTK_CONTAINER(cur_list2), listitem32);
gtk_widget_show(listitem32);

GtkWidget *cur_list3 = lookup_widget(GTK_WIDGET(window3), "list33");
listitem33 = gtk_list_item_new_with_label(INVFRAG);
gtk_container_add(GTK_CONTAINER(cur_list3), listitem33);
gtk_widget_show(listitem33);

GtkWidget *cur_list4 = lookup_widget(GTK_WIDGET(window3), "list34");
listitem34 = gtk_list_item_new_with_label(TXRET);
gtk_container_add(GTK_CONTAINER(cur_list4), listitem34);
gtk_widget_show(listitem34);

GtkWidget *cur_list5 = lookup_widget(GTK_WIDGET(window3), "list35");
listitem35 = gtk_list_item_new_with_label(BEACON);
gtk_container_add(GTK_CONTAINER(cur_list5), listitem35);
gtk_widget_show(listitem35);

GtkWidget *cur_list6 = lookup_widget(GTK_WIDGET(window3), "list36");

```

```

listitem36 = gtk_list_item_new_with_label(INVMISC);
gtk_container_add(GTK_CONTAINER(cur_list6), listitem36);
gtk_widget_show(listitem36);
}

void thresholds()
{
    const char *SENSITIVITY = "-121"; //Sensitivity
    const char *INVCRYPT = "2347"; //RTS Threshold
    const char *INVFRAG = "2346"; // Fragmentation Threshold
    GtkWidget *listitem41;
    GtkWidget *list41;
    GtkWidget *listitem42;
    GtkWidget *list42;
    GtkWidget *listitem43;
    GtkWidget *list43;

    GtkWidget *cur_list1 = lookup_widget(GTK_WIDGET(window4), "list41");
    listitem41 = gtk_list_item_new_with_label(SENSITIVITY);
    gtk_container_add(GTK_CONTAINER(cur_list1), listitem41);
    gtk_widget_show(listitem41);

    GtkWidget *cur_list2 = lookup_widget(GTK_WIDGET(window4), "list42");
    listitem42 = gtk_list_item_new_with_label(INVCRYPT);
    gtk_container_add(GTK_CONTAINER(cur_list2), listitem42);
    gtk_widget_show(listitem42);

    GtkWidget *cur_list3 = lookup_widget(GTK_WIDGET(window4), "list43");
    listitem43 = gtk_list_item_new_with_label(INVFRAG);
    gtk_container_add(GTK_CONTAINER(cur_list3), listitem43);
    gtk_widget_show(listitem43);
}

void show_description (const char *description)
{
    GtkWidget *listitem; // Orizei ena item tupou GList
    GtkWidget *list_info_description; //Orizei to list_info_description pou exoume kai sto window5
    GtkWidget *list = lookup_widget(GTK_WIDGET(window5), "list_info_description");//Ftiaxnei
    ena pointer, kai to vazei na deixnei sto list_info_description
    listitem = gtk_list_item_new_with_label(description);//Dhmiourgei to neo listitem, me to label
    "description"
    gtk_container_add(GTK_CONTAINER(list), listitem); //Prosthetei to list item, sto container tou
    list (to opoio list pleon = list_info_description)
    gtk_widget_show(listitem); //Emfanizei to listitem
}

void show_cur_value(const char *cur_value)
{
    GtkWidget *listitem1;
    GtkWidget *list_info;
    GtkWidget *cur_list = lookup_widget(GTK_WIDGET(window5), "list_info");
    listitem1 = gtk_list_item_new_with_label(cur_value);
    gtk_container_add(GTK_CONTAINER(cur_list), listitem1);
    gtk_widget_show(listitem1);
}

```

```

}

// -----Ploghsh Sto HOME (window0)-----

void
on_toolbutton_home1_clicked      (GtkToolButton *toolbutton,
                                  gpointer        user_data)
{
    GtkWidget *hidewin;
    hidewin = GTK_WIDGET(lookup_widget(window1, "window1"));
    gtk_widget_hide(hidewin);
}

void //test
on_toolbutton_home2_clicked      (GtkToolButton *toolbutton,
                                  gpointer        user_data)
{
    GtkWidget *hidewin;
    hidewin = GTK_WIDGET(lookup_widget(window2, "window2"));
    gtk_widget_hide(hidewin);
}

void
on_toolbutton_home3_clicked      (GtkToolButton *toolbutton,
                                  gpointer        user_data)
{
    GtkWidget *hidewin;
    hidewin = GTK_WIDGET(lookup_widget(window3, "window3"));
    gtk_widget_hide(hidewin);
}

void
on_toolbutton_home4_clicked      (GtkToolButton *toolbutton,
                                  gpointer        user_data)
{
    GtkWidget *hidewin;
    hidewin = GTK_WIDGET(lookup_widget(window4, "window4"));
    gtk_widget_hide(hidewin);
}

// -----Teratismos Susthmatos-----

//Kleisimo Susthmatos, X tou window0
void
on_window0_destroy               (GtkObject *object,
                                  gpointer   user_data)
{
    gtk_main_quit();
}

//Kleisimo Susthmatos, X tou window1
void

```

```

on_window1_destroy      (GtkObject  *object,
                        gpointer     user_data)
{
    gtk_main_quit();
}

//Kleisimo Susthmatos, X tou window2
void
on_window2_destroy      (GtkObject  *object,
                        gpointer     user_data)
{
    gtk_main_quit();
}

//Kleisimo Susthmatos, X tou window3
void
on_window3_destroy      (GtkObject  *object,
                        gpointer     user_data)
{
    gtk_main_quit();
}

//Kleisimo Susthmatos, X tou window4
void
on_window4_destroy      (GtkObject  *object,
                        gpointer     user_data)
{
    gtk_main_quit();
}

//Kleisimo susthmatos, toolbuttons "exit"
void
on_toolbutton_exit1_clicked (GtkToolButton *toolbutton,
                            gpointer     user_data)
{
    gtk_main_quit();
}

void
on_toolbutton_exit2_clicked (GtkToolButton *toolbutton,
                            gpointer     user_data)
{
    gtk_main_quit();
}

void
on_toolbutton_exit3_clicked (GtkToolButton *toolbutton,
                            gpointer     user_data)
{
    gtk_main_quit();
}

void
on_toolbutton_exit4_clicked (GtkToolButton *toolbutton,
                            gpointer     user_data)
{
    gtk_main_quit();
}

```

```

}

// -----Window0, arxikes epiloges-----

// Anoigma "Overview"
void
on_button1_clicked      (GtkButton  *button,
                          gpointer    user_data)
{
window1 = create_window1 (); //Dhmiourgei to window1
gtk_widget_show (window1); // Anoigei to window1

overview(); //Fortwnei oles tis times twn parametrwn tou Overview
}

// Anoigma "Measurements"
void
on_button2_clicked      (GtkButton  *button,
                          gpointer    user_data)
{
window2 = create_window2 ();
gtk_widget_show (window2);
measurements();
}

// Anoigma "Losses"
void
on_button3_clicked      (GtkButton  *button,
                          gpointer    user_data)
{
window3 = create_window3 ();
gtk_widget_show (window3);
losses();
}

// Anoigma "Thresholds"
void
on_button4_clicked      (GtkButton  *button,
                          gpointer    user_data)
{
window4 = create_window4 ();
gtk_widget_show (window4);
thresholds();
}

// -----Window1, Overview-----

//Refresh
void
on_button_refresh1_clicked      (GtkButton  *button,
                                  gpointer    user_data)
{
GtkWidget *hidewin; //Kruvei to window1

```



```

hidewin = GTK_WIDGET(lookup_widget(window1, "window1"));
gtk_widget_hide(hidewin);
window1 = create_window1 ();
gtk_widget_show (window1);

overview();
}

void
on_toolbutton_forward1_clicked (GtkToolButton *toolbutton,
                                gpointer        user_data)
{
window2 = create_window2 ();
gtk_widget_show (window2);
measurements();
GtkWidget *hidewin; // Arxikopoihsh
hidewin = GTK_WIDGET(lookup_widget(window1, "window1")); // Vriskei to window1 kai
vazei to pointer ekei
gtk_widget_hide(hidewin);
}

// -----Window2, Thresholds-----

void
on_button_refresh2_clicked (GtkButton *button,
                             gpointer   user_data)
{
GtkWidget *hidewin;
hidewin = GTK_WIDGET(lookup_widget(window2, "window2"));
gtk_widget_hide(hidewin);
window2 = create_window2 ();
gtk_widget_show (window2);
measurements();
}

void
on_toolbutton_back2_clicked (GtkToolButton *toolbutton,
                              gpointer      user_data)
{
window1 = create_window1 ();
gtk_widget_show (window1);
overview();
GtkWidget *hidewin; // Arxikopoihsh
hidewin = GTK_WIDGET(lookup_widget(window2, "window2"));
gtk_widget_hide(hidewin);
}

void
on_toolbutton_forward2_clicked (GtkToolButton *toolbutton,
                                 gpointer      user_data)
{
window3 = create_window3 ();
gtk_widget_show (window3);
}

```

```

losses();
GtkWidget *hidewin; // Arxikopoihsh
hidewin = GTK_WIDGET(lookup_widget(window2, "window2")); // Vriskei to window2 kai
vazei to pointer ekei
gtk_widget_hide(hidewin);
}

```

```
// -----Window3, Losses-----
```

```

void
on_button_refresh3_clicked (GtkWidget *button,
                             gpointer   user_data)
{
GtkWidget *hidewin;
hidewin = GTK_WIDGET(lookup_widget(window3, "window3"));
gtk_widget_hide(hidewin);
window3 = create_window3 ();
gtk_widget_show (window3);
losses();
}

```

```

void
on_toolbutton_back3_clicked (GtkToolButton *toolbutton,
                              gpointer      user_data)
{
window2 = create_window2 ();
gtk_widget_show (window2);
measurements();
GtkWidget *hidewin; // Arxikopoihsh
hidewin = GTK_WIDGET(lookup_widget(window3, "window3")); // Vriskei to window3 kai
vazei to pointer ekei
gtk_widget_hide(hidewin);
}

```

```

void
on_toolbutton_forward3_clicked (GtkToolButton *toolbutton,
                                 gpointer      user_data)
{
window4 = create_window4 ();
gtk_widget_show (window4);
thresholds();
GtkWidget *hidewin; // Arxikopoihsh
hidewin = GTK_WIDGET(lookup_widget(window3, "window3")); // Vriskei to window3 kai
vazei to pointer ekei
gtk_widget_hide(hidewin);
}

```

```
// -----Window4, Thresholds-----
```

```

void
on_button_refresh4_clicked (GtkWidget *button,

```

```

                gpointer      user_data)
{
    GtkWidget *hidewin;
    hidewin = GTK_WIDGET(lookup_widget(window4, "window4"));
    gtk_widget_hide(hidewin);
    window4 = create_window4 ();
    gtk_widget_show (window4);
    thresholds();
}

void
on_toolbutton_back4_clicked (GtkToolButton *toolbutton,
                             gpointer      user_data)
{
    window3 = create_window3 ();
    gtk_widget_show (window3);
    losses();
    GtkWidget *hidewin; // Arxikopoihsh
    hidewin = GTK_WIDGET(lookup_widget(window4, "window4")); // Vriskei to window4 kai
    vazei to pointer ekei
    gtk_widget_hide(hidewin);
}

// -----Window5, Info-----

//Kleisimo tou parathourou info
void
on_button_info_close_clicked (GtkButton *button,
                              gpointer   user_data)
{
    gtk_widget_destroy(window5);
}

//-----Anoigma tou Info window (window5)-----

//Information tou window1, anoigei to Window5
void
on_button_info1_clicked (GtkButton *button,
                        gpointer   user_data)
{
    window5 = create_window5 ();
    gtk_widget_show (window5);
}

//Information tou window2, anoigei to Window5
void
on_button_info2_clicked (GtkButton *button,
                        gpointer   user_data)
{
    window5 = create_window5 ();
    gtk_widget_show (window5);
}

```

```

//Information tou window3, anoigei to Window5
void
on_button_info3_clicked      (GtkButton   *button,
                              gpointer     user_data)
{
    window5 = create_window5 ();
    gtk_widget_show (window5);
}

//Information tou window4, anoigei to Window5
void
on_button_info4_clicked      (GtkButton   *button,
                              gpointer     user_data)
{
    window5 = create_window5 ();
    gtk_widget_show (window5);
}

//----- "Overview" tou Info-----

void refresh_window5()
{
    gtk_widget_hide(window5);
    window5 = create_window5 ();
    gtk_widget_show (window5);
}

//info gia to overview
void
on_overview_info1_activate   (GtkMenuItem *menuitem,
                              gpointer     user_data)
{
}

void
on_mac_protocol1_activate    (GtkMenuItem *menuitem,
                              gpointer     user_data)
{
    refresh_window5();
    const gchar *text;
    const gchar *value;
    text = "MAC Protocol: Displays the MAC Protocol used.";
    value = "IEEE 802.11g";
    show_description(text);
    show_cur_value(value);
}

void
on_essid1_activate           (GtkMenuItem *menuitem,
                              gpointer     user_data)

```

```

        gpointer      user_data)
{
    refresh_window5();
    const gchar *text;
    const gchar *value;
    text = "ESSID: Displays the Network Name or the\nDomain ID. Used to identify cells which
are\npart of the same virtual network.";
    value = "AP-307";
    show_description(text);
    show_cur_value(value);
}

void
on_access_point1_activate      (GtkMenuItem *menuitem,
        gpointer      user_data)
{
    refresh_window5();
    const gchar *text;
    const gchar *value;
    text = "Access Point: Displays the address of the\nAccess Point to which the wireless card is\
ncurrently registered.";
    value = "00:16:B6:D9:27:54";
    show_description(text);
    show_cur_value(value);
}

void
on_link_quality1_activate      (GtkMenuItem *menuitem,
        gpointer      user_data)
{
    refresh_window5();
    const gchar *text;
    const gchar *value;
    text = "Link Quality: Overall quality of the link.\nMay be based on the level of contention
or\ninterference, the BER or FER, how good the\nreceived signal is, or other parameters.";
    value = "56/100";
    show_description(text);
    show_cur_value(value);
}

void
on_mode1_activate              (GtkMenuItem *menuitem,
        gpointer      user_data)
{
    refresh_window5();
    const gchar *text;
    const gchar *value;
    text = "Mode: Displays the operating mode of the\ndevice, which depends on the
network\ntopology (Ad-Hoc, Managed, Master, Repeater,\nSecondary, Monitor or Auto).";
    value = "Managed";
    show_description(text);
    show_cur_value(value);
}

void
on_power_management1_activate  (GtkMenuItem *menuitem,

```

```

                                gpointer      user_data)
{
refresh_window5();
const gchar *text;
const gchar *value;
text = "Power Management: Displays whether the\npower management settings are enabled
or\ndisabled.";
value = "Off";
show_description(text);
show_cur_value(value);
}

```

```
//-----Menu "Measurements" tou Info-----
```

```

void
on_measurements_info2_activate      (GtkMenuItem *menuItem,
                                gpointer      user_data)
{
}

```

```

void
on_link_quality2_activate           (GtkMenuItem *menuItem,
                                gpointer      user_data)
{
refresh_window5();
const gchar *text;
const gchar *value;
text = "Link Quality: Overall quality of the link.\nMay be based on the level of contention
or\ninterference, the BER or FER, how good the\nreceived signal is, or other parameters.";
value = "56/100";
show_description(text);
show_cur_value(value);
}

```

```

void
on_bit_rate2_activate              (GtkMenuItem *menuItem,
                                gpointer      user_data)
{
refresh_window5();
const gchar *text;
const gchar *value;
text = "Bit Rate: The bit rate is the speed at which\nbits are transmitted over the
medium.\nMeasured in bits per second.\n";
value = "54 Mb/s";
show_description(text);
show_cur_value(value);
}

```

```

void
on_signal_level2_activate          (GtkMenuItem *menuItem,
                                gpointer      user_data)
{
refresh_window5();
const gchar *text;
const gchar *value;

```

```

text = "Signal Level: Displays how strong the\nreceived signal is. May be arbitrary units\nor
dBm.";
value = "-60 dBm";
show_description(text);
show_cur_value(value);
}

```

```

void
on_noise_level2_activate      (GtkMenuItem *menuitem,
                               gpointer      user_data)
{
refresh_window5();
const gchar *text;
const gchar *value;
text = "Noise Level: Displays how strong the\nreceived noise is. May be arbitrary units\nor
dBm.";
value = "-96 dBm";
show_description(text);
show_cur_value(value);
}

```

```

void
on_frequency2_activate      (GtkMenuItem *menuitem,
                              gpointer      user_data)
{
refresh_window5();
const gchar *text;
const gchar *value;
text = "Frequency: Displays the operating frequency\nin the device, in Hz.";
value = "2.432 GHz";
show_description(text);
show_cur_value(value);
}

```

```

void
on_tx_power2_activate      (GtkMenuItem *menuitem,
                             gpointer      user_data)
{
refresh_window5();
const gchar *text;
const gchar *value;
text = "Tx Power: Displays the transmit power,\nin dBm.";
value = "20 dBm";
show_description(text);
show_cur_value(value);
}

```

//-----Menu "Losses" tou Info-----

```

void
on_losses_info3_activate      (GtkMenuItem *menuitem,
                               gpointer      user_data)
{
}

```

```

void
on_rx_invalid_nwid3_activate (GtkMenuItem *menuItem,
                             gpointer      user_data)
{
    refresh_window5();
    const gchar *text;
    const gchar *value;
    text = "Rx Invalid NWID: Displays the number of \npackets received with a different NWID
\nor ESSID.\n";
    value = "0";
    show_description(text);
    show_cur_value(value);
}

void
on_rx_invalid_crypt3_activate (GtkMenuItem *menuItem,
                              gpointer      user_data)
{
    refresh_window5();
    const gchar *text;
    const gchar *value;
    text = "Rx Invalid Crypt.: Displays the number\nof packets that the hardware was unable \nto
decrypt.";
    value = "0";
    show_description(text);
    show_cur_value(value);
}

void
on_rx_invalid_fragm3_activate (GtkMenuItem *menuItem,
                              gpointer      user_data)
{
    refresh_window5();
    const gchar *text;
    const gchar *value;
    text = "Rx Invalid Fragm.: Displays the number of \npackets for which the hardware was not
able\nto properly re-assemble the link layer \nfragments.";
    value = "0";
    show_description(text);
    show_cur_value(value);
}

void
on_tx_excessive_retries3_activate (GtkMenuItem *menuItem,
                                   gpointer      user_data)
{
    refresh_window5();
    const gchar *text;
    const gchar *value;
    text = "Tx Excessive Retries: Displays the number of\npackets that the hardware failed to de-
liver.";
    value = "0";
    show_description(text);
    show_cur_value(value);
}

```



```

void
on_missed_beacon3_activate      (GtkMenuItem *menuItem,
                                gpointer      user_data)
{
refresh_window5();
const gchar *text;
const gchar *value;
text = "Missed Beacon: Displays the number of \nperiodic beacons from the Cell or the Ac-
cess\nPoint that have been missed.";
value = "0";
show_description(text);
show_cur_value(value);
}

```

```

void
on_invalid_misc3_activate      (GtkMenuItem *menuItem,
                                gpointer      user_data)
{
refresh_window5();
const gchar *text;
const gchar *value;
text = "Invalid Misc: Displays the number of packets\nthat have been lost in relation
with\nspecific wireless operations. ";
value = "0";
show_description(text);
show_cur_value(value);
}

```

//-----Menu "Thresholds" tou Info-----

```

void
on_thresholds_info4_activate   (GtkMenuItem *menuItem,
                                gpointer      user_data)
{
}

```

```

void
on_sensitivity4_activate      (GtkMenuItem *menuItem,
                                gpointer      user_data)
{
refresh_window5();
const gchar *text;
const gchar *value;
text = "Sensitivity: Displays the lower signal level\nfor which the hardware will consider re-
ceive\npackets usable. Negative values are assumed\nto be dBm.\n";
value= "-121";
show_description(text);
show_cur_value(value);
}

```

```

void
on_rts_threshold4_activate     (GtkMenuItem *menuItem,
                                gpointer      user_data)
{

```

```

refresh_window5();
const gchar *text;
const gchar *value;
text = "RTS Threshold: Displays the size of the \nsmallest packet for which the node
sends\nthe Request To Send message.";
value = "2347";
show_description(text);
show_cur_value(value);
}

```

```

void
on_fragmentation_thr4_activate (GtkMenuItem *menuItem,
                                gpointer user_data)
{
refresh_window5();
const gchar *text;
const gchar *value;
text = "Fragmentation Threshold: Displays the\nmaximum fragment size. Fragm/tion
allows\nto split an IP packet in smaller fragments\ntransmitted on the medium.";
value = "2346";
show_description(text);
show_cur_value(value);
}

```

//-----MENU EPILOGH

WINDOWS-----

//MENU tou window1

```

void
on_overview1_activate (GtkMenuItem *menuItem,
                       gpointer user_data)
{
GtkWidget *hidewin; //Metavlth gia na fugei to trexon parathuro, epeidh tha 3anafortwthei to
idio
hidewin = GTK_WIDGET(lookup_widget(window1, "window1")); // Vriskei to parathuro
gtk_widget_hide(hidewin); // To kruvei
window1 = create_window1 (); //Emfanizei to idio, se kainouria morfh (refreshed)
gtk_widget_show (window1);

```

```

overview();
}

```

```

void
on_measurements1_activate (GtkMenuItem *menuItem,
                            gpointer user_data)
{
window2 = create_window2 ();
gtk_widget_show (window2);
measurements();
GtkWidget *hidewin;
hidewin = GTK_WIDGET(lookup_widget(window1, "window1"));
gtk_widget_hide(hidewin);
}

```

```

void
on_losses1_activate (GtkMenuItem *menuItem,
                    gpointer user_data)

```

```

{
window3 = create_window3 ();
gtk_widget_show (window3);
losses();
GtkWidget *hidewin;
hidewin = GTK_WIDGET(lookup_widget(window1, "window1"));
gtk_widget_hide(hidewin);
}

void
on_thresholds1_activate      (GtkMenuItem *menuitem,
                             gpointer      user_data)
{
window4 = create_window4 ();
gtk_widget_show (window4);
thresholds();
GtkWidget *hidewin;
hidewin = GTK_WIDGET(lookup_widget(window1, "window1"));
gtk_widget_hide(hidewin);
}

//MENU tou window2
void
on_overview2_activate      (GtkMenuItem *menuitem,
                             gpointer      user_data)
{
window1 = create_window1 ();
gtk_widget_show (window1);
overview();
GtkWidget *hidewin;
hidewin = GTK_WIDGET(lookup_widget(window2, "window2"));
gtk_widget_hide(hidewin);
}

void
on_measurements2_activate  (GtkMenuItem *menuitem,
                             gpointer      user_data)
{
GtkWidget *hidewin;
hidewin = GTK_WIDGET(lookup_widget(window2, "window2"));
gtk_widget_hide(hidewin);
window2 = create_window2 ();
gtk_widget_show (window2);
measurements();
}

void
on_losses2_activate      (GtkMenuItem *menuitem,
                             gpointer      user_data)
{
window3 = create_window3 ();
gtk_widget_show (window3);
losses();
GtkWidget *hidewin;
hidewin = GTK_WIDGET(lookup_widget(window2, "window2"));
gtk_widget_hide(hidewin);
}

```

```

void
on_thresholds2_activate      (GtkMenuItem *menuItem,
                             gpointer      user_data)
{
    window4 = create_window4 ();
    gtk_widget_show (window4);
    thresholds();
    GtkWidget *hidewin;
    hidewin = GTK_WIDGET(lookup_widget(window2, "window2"));
    gtk_widget_hide(hidewin);
}

//MENU tou window3
void
on_overview3_activate      (GtkMenuItem *menuItem,
                             gpointer      user_data)
{
    window1 = create_window1 ();
    gtk_widget_show (window1);
    overview();
    GtkWidget *hidewin;
    hidewin = GTK_WIDGET(lookup_widget(window3, "window3"));
    gtk_widget_hide(hidewin);
}

void
on_measurements3_activate  (GtkMenuItem *menuItem,
                             gpointer      user_data)
{
    window2 = create_window2 ();
    gtk_widget_show (window2);
    measurements();
    GtkWidget *hidewin;
    hidewin = GTK_WIDGET(lookup_widget(window3, "window3"));
    gtk_widget_hide(hidewin);
}

void
on_losses3_activate       (GtkMenuItem *menuItem,
                             gpointer      user_data)
{
    GtkWidget *hidewin;
    hidewin = GTK_WIDGET(lookup_widget(window3, "window3"));
    gtk_widget_hide(hidewin);
    window3 = create_window3 ();
    gtk_widget_show (window3);
    losses();
}

void
on_thresholds3_activate    (GtkMenuItem *menuItem,
                             gpointer      user_data)
{
    window4 = create_window4 ();
    gtk_widget_show (window4);
    thresholds();
    GtkWidget *hidewin;
}

```

```
hidewin = GTK_WIDGET(lookup_widget(window3, "window3"));
gtk_widget_hide(hidewin);
}
```

```
//MENU tou window4
```

```
void
on_overview4_activate      (GtkMenuItem *menuitem,
                             gpointer     user_data)
{
    window1 = create_window1 ();
    gtk_widget_show (window1);
    overview();
    GtkWidget *hidewin;
    hidewin = GTK_WIDGET(lookup_widget(window4, "window4"));
    gtk_widget_hide(hidewin);
}
```

```
void
on_measurements4_activate  (GtkMenuItem *menuitem,
                             gpointer     user_data)
{
    window2 = create_window2 ();
    gtk_widget_show (window2);
    measurements();
    GtkWidget *hidewin;
    hidewin = GTK_WIDGET(lookup_widget(window4, "window4"));
    gtk_widget_hide(hidewin);
}
```

```
void
on_losses4_activate        (GtkMenuItem *menuitem,
                             gpointer     user_data)
{
    window3 = create_window3 ();
    gtk_widget_show (window3);
    losses();
    GtkWidget *hidewin;
    hidewin = GTK_WIDGET(lookup_widget(window4, "window4"));
    gtk_widget_hide(hidewin);
}
```

```
void
on_thresholds4_activate    (GtkMenuItem *menuitem,
                             gpointer     user_data)
{
    GtkWidget *hidewin;
    hidewin = GTK_WIDGET(lookup_widget(window4, "window4"));
    gtk_widget_hide(hidewin);
    window4 = create_window4 ();
    gtk_widget_show (window4);
    thresholds();
}
```

ΠΑΡΑΡΤΗΜΑ II

Iwconfig.c Compile & Run

Linux Εντολές

Τα Iwtools, προσφέρουν automake αρχεία ώστε με την εντολή «make» στο τερματικό να μεταγλωττίζουν και να εγκαθιστούν τα προγράμματα, αυτόματα.

Επειδή, όμως, το Scratchbox δεν αναγνωρίζει τα προγράμματα αν γίνουν compiled μέσω της make, η μεταγλώττισή τους πρέπει να γίνει χειρωνακτικά και με άλλες εκδόσεις βιβλιοθηκών τις οποίες αυτό υποστηρίζει. Ακολουθούν οι εντολές που πρέπει να εισαχθούν στην κονσόλα των Linux για το compile των iwtools προγραμμάτων, ώστε να είναι συμβατά με το Scratchbox και κατ' επέκταση με το Maemo:

```
gcc -O2 -W -Wall -Wstrict-prototypes -c iwlib.c
rm -f libiw.a
ar cru libiw.a iwlib.o
ranlib libiw.a
gcc -O2 -W -Wall -Wstrict-prototypes -fPIC -c -o iwlib.so iwlib.c
gcc -shared -o libiw.so.26 -Wl,-soname,libiw.so.26 -lm -lc iwlib.so
gcc -O2 -W -Wall -Wstrict-prototypes -c iwconfig.c
gcc -O2 -W -Wall -Wstrict-prototypes -o iwconfig iwconfig.o libiw.a -lm
gcc -O2 -W -Wall -Wstrict-prototypes -c iwlist.c
gcc -O2 -W -Wall -Wstrict-prototypes -o iwlist iwlist.o libiw.a -lm
gcc -O2 -W -Wall -Wstrict-prototypes -c iwpriv.c
gcc -O2 -W -Wall -Wstrict-prototypes -o iwpriv iwpriv.o libiw.a -lm
gcc -O2 -W -Wall -Wstrict-prototypes -c iwspy.c
```

```
gcc -O2 -W -Wall -Wstrict-prototypes -o iwspy iwspy.o libiw.a -lm
gcc -O2 -W -Wall -Wstrict-prototypes -c iwgetid.c
gcc -O2 -W -Wall -Wstrict-prototypes -o iwgetid iwgetid.o -lm
gcc -O2 -W -Wall -Wstrict-prototypes -c iwevent.c
gcc -O2 -W -Wall -Wstrict-prototypes -o iwevent iwevent.o libiw.a -lm
```

Σημείωση: Για το compilation μόνο του iwconfig.c αρκούν οι πρώτες 8 εντολές.