ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ

ΙΔΡΥΜΑ ΠΑΤΡΑΣ

ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ
ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΙΑΣ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Αριθμός 1159

ΘΕΜΑ : << **ΜΕΛΕΤΗ ΚΑΙ ΑΝΑΠΤΥΞΗ ΕΦΑΡΜΟΓΗΣ** VOIP (VOICE OVER IP) **Η΄/ΚΑΙ** VVOIP(VOICE AND VIDEO OVER IP) ΜΕ ΤΗ **ΧΡΗΣΗ ΤΟΥ** SIP **ΠΡΩΤΟΚΟΛΛΟΥ** >>

ΕΙΣΗΓΗΤΗΣ :

ΔΗΜΗΤΡΗΣ ΚΑΡΕΛΗΣ

ΣΠΟΥΔΑΣΤΗΣ :

ΓΙΑΝΝΗ ΓΚΕΝΤΙΑΝ

ΠΑΤΡΑ 2011

ΠΕΡΙΛΗΨΗ

Τα τελευταία χρόνια με την ραγδαία ανάπτυξη του διαδικτύου έγινε εφικτή η χρησιμοποίηση νέων τεχνολογιών και υπηρεσιών στον κλάδο των επικοινωνιών. Η παρούσα διπλωματική αναφέρεται σε τεχνολογίες VoIP / VVoIP, όπου αναμένεται να είναι οι αντικαταστάτες του παραδοσιακού τηλεφωνικού συστήματος μεταγωγής κυκλώματος PSTN. Ως VoIP / VVoIP ονομάζουμε μια ομάδα τεχνολογιών και πρωτοκόλλων με τις οποίες είναι εφικτή η επικοινωνία δύο ή περισσοτέρων ,απομακρυσμένων μεταξύ τους, χρηστών με μεταφορά ήχου/εικόνας μέσω IP δικτύων όπως το Internet.Οι VoIP / VVoIP εφαρμογές συνήθως ταξινομούνται σύμφωνα με τα υποστηριζόμενα πρωτόκολλα που υλοποιούν όπως τα H.323, SIP, Megaco H.248,MGCP.Ανάμεσα σε αυτά τα πρωτόκολλα το SIP θεωρείται από τα πλέον ιδανικά για VoIP συστήματα.
Σκοπός αυτής της διπλωματικής είναι η μελέτη του SIP πρωτοκόλλου καθώς και η σχεδίαση, ανάπτυξη και υλοποίηση VVoIP εφαρμογής που θα επιτρέπει τη διαχείριση συνόδων μεταξύ δύο μερών με χρήση σηματοδοσίας βασισμένης στο πρωτόκολλο SIP. Η εφαρμογή περιλαμβάνει τόσο την πλευρά του εξυπηρετητή, όσο και του πελάτη.
Η διπλωματική εργασία χωρίζεται σε δύο σκέλη:

I. Μελέτη των VoIP τεχνολογιών και ανάλυση του SIP πρωτοκόλλου.

II. Παρουσίαση και ανάλυση της VVoIP εφαρμογής.

Το πρώτο σκέλος περιλαμβάνει το πρώτο κεφάλαιο το οποίο περιλαμβάνει μια ιστορική αναδρομή της τηλεφωνίας και τα βασικά χαρακτηριστικά των VoIP τεχνολογιών και συστημάτων. Στο δεύτερο κεφάλαιο γίνεται μια αναλυτική περιγραφή του SIP πρωτοκόλλου και στα κεφάλαια τρία και τέσσερα παρουσιάζονται δύο πρωτόκολλα στενά συνδεδεμένα με το SIP, τα RTP/RTCP και SDP αντίστοιχα.
Το δεύτερο σκέλος αποτελείται από τα κεφάλαια 5 έως 8.Στο κεφάλαιο 5 παρουσιάζεται η JMF βιβλιοθήκη δίνοντας έμφαση στη σχέση της με το RTP για την μετάδοση δεδομένων πραγματικού χρόνου. Στο κεφάλαιο 6 προχωρούμε στην ανάλυση της σχεδίασης της εφαρμογής και στο κεφάλαιο 7 παρουσιάζονται τα βασικά πακέτα και κλάσεις της Java εφαρμογής. Τέλος στο όγδοο κεφάλαιο παραθέτουμε κάποια παραδείγματα εκτέλεσης.

ΠΕΡΙΕΧΟΜΕΝΑ

ΠΕΡΙΛΗΨΗ....2
ΠΕΡΙΕΧΟΜΕΝΑ....3
ΛΙΣΤΑ ΕΙΚΟΝΩΝ....5
ΛΙΣΤΑ ΣΧΗΜΑΤΩΝ....6
ΛΙΣΤΑ ΠΙΝΑΚΩΝ....7
ΚΕΦΑΛΑΙΟ 1: **ΕΙΣΑΓΩΓΗ**.... 10
1.1 **ΤΗΛΕΦΩΝΙΑ-ΙΣΤΟΡΙΚΗ ΑΝΑΔΡΟΜΗ**....10
1.2 **ΑΠΟ ΤΑ ΑΝΑΛΟΓΙΚΑ ΣΤΑ ΨΗΦΙΑΚΑ ΣΥΣΤΗΜΑΤΑ ΕΠΙΚΟΙΝΩΝΙΩΝ**....12
1.3 **ΤΙ ΕΙΝΑΙ** VOIP-VVOIP....14
1.4 **ΣΤΟΙΒΑ ΠΡΩΤΟΚΟΛΛΩΝ** VOIP....16
1.5 **ΤΗΛΕΔΙΑΣΚΕΨΗ** (VIDEOCONFERENCE)....18
1.6 **ΣΥΜΠΕΡΑΣΜΑΤΑ**....21
ΚΕΦΑΛΑΙΟ 2: **ΤΟ** SIP **ΠΡΩΤΟΚΟΛΛΟ**....23
2.1 **ΓΕΝΙΚΑ**....23
2.2 SIP **ΟΝΤΟΤΗΤΕΣ**....24
2.2.1 **ΠΡΑΚΤΟΡΕΣ ΧΡΗΣΤΗ**-USER AGENT....24
2.2.2 PROXY SERVER....24
2.2.2.1 STATEFUL PROXY SERVERS....25
2.2.2.2 STATELESS PROXY SERVERS....25
2.2.3 REDIRECT SERVERS....26
2.2.4 REGISTRAR SERVERS....26
2.3 SIP **ΜΗΝΥΜΑΤΑ**....27
2.3.1 **ΓΕΝΙΚΗ ΔΟΜΗ**....27
2.3.2 **ΕΠΙΚΕΦΑΛΙΔΕΣ** (HEADER FIELDS)....28
ΕΠΙΚΕΦΑΛΙΔΑ....29
ΠΕΡΙΓΡΑΦΗ....29
2.3.3 **ΟΙ ΕΝΝΟΙΕΣ** DIALOG **ΚΑΙ** TRANSACTION....32
2.3.4 SIP REQUESTS....33
2.3.4.1 REGISTER....34
2.3.4.2 INVITE....36
2.3.4.3 ACK....36
2.3.4.4 CANCEL....37
2.3.4.5 BYE....37
2.3.4.6 OPTIONS....38
2.3.5 SIP RESPONSES....38
2.3.5.1 PROVISIONAL (1xx)....39
2.3.5.2 SUCCESS (2xx)....39
2.3.5.3 REDIRECTION (3xx)....40
2.3.5.4 CLIENT ERROR (4xx)....40
2.3.5.5 SERVER ERROR (5xx)....41
2.3.5.6 GLOBAL FAILURE (6xx)....42

2.4 SIP **ΔΙΕΥΘΥΝΣΕΙΣ**……42
2.4.1 SIP URI……42
2.4.2 SIPS URI……43
2.4.3 TELEPHONE URI……43
2.5 **ΠΑΡΑΔΕΙΓΜΑΤΑ**……43
2.5.1 REGISTRATION……43
2.5.2 INVITE WITH PROXY SERVERS……45
2.5.3 INVITE WITH PROXY AND REDIRECT SERVERS……46
ΚΕΦΑΛΑΙΟ 3: RTP/RTCP……49
3.1 **ΔΟΜΗ** RTP **ΠΑΚΕΤΟΥ**……50
3.2 **ΔΟΜΗ** RTCP **ΠΑΚΕΤΟΥ**……53
ΚΕΦΑΛΑΙΟ 4: SDP……60
4.1 ARXITEKTONIKH SDP……62
4.2 PAYLOAD TYPES FOR STANDARD AUDIO/VIDEO……64
ΚΕΦΑΛΑΙΟ 5: JMF……66
5.1 **ΑΡΧΙΤΕΚΤΟΝΙΚΗ**……67
5.2 JMF/RTP……70
ΚΕΦΑΛΑΙΟ 6: **ΥΛΟΠΟΙΗΣΗ**……74
6.1 **ΑΝΑΛΥΣΗ ΣΧΕΔΙΑΣΗΣ**……74
6.1.1 **ΓΕΝΙΚΑ**……74
6.1.2 **ΒΙΒΛΙΟΘΗΚΕΣ**……74
6.1.3 **ΓΙΑΤΙ** MYSQL……75
6.1.4 **ΑΠΑΙΤΗΣΕΙΣ ΣΥΣΤΗΜΑΤΟΣ**……75
6.2 CLIENT……76
6.2.1 CONFIGURATION……76
6.2.2 **ΕΚΚΙΝΗΣΗ ΕΦΑΡΜΟΓΗΣ**……78
6.2.3 **ΛΕΙΤΟΥΡΓΙΕΣ**……79
6.3 REGISTRAR SERVER……89
6.3.1 CONFIGURATION……89
6.3.2 **ΕΚΚΙΝΗΣΗ ΕΦΑΡΜΟΓΗΣ**……90
6.3.3 **ΛΕΙΤΟΥΡΓΙΕΣ**……91
6.4 LOCATION SERVICE ……94
6.4.1 **ΕΓΚΑΤΑΣΤΑΣΗ**……94
6.4.2 **ΕΚΚΙΝΗΣΗ**……96
6.4.3 **ΠΕΡΙΓΡΑΦΗ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ**……96
ΚΕΦΑΛΑΙΟ 7: **ΠΕΡΙΓΡΑΦΗ ΤΩΝ ΚΛΑΣΕΩΝ**……99
7.1 PACKAGE CLIENT……99
7.2 PACKAGE SERVER……102
ΚΕΦΑΛΑΙΟ 8: **ΠΑΡΑΔΕΙΓΜΑΤΑ ΕΚΤΕΛΕΣΗΣ**……105
8.1 **ΕΓΓΡΑΦΗ**(REGISTER)……105
8.2 **ΚΛΗΣΗ**(INVITE……109
ΒΙΒΛΙΟΓΡΑΦΙΑ-ΑΝΑΦΟΡΕΣ……113

ΛΙΣΤΑ ΕΙΚΟΝΩΝ

Εικόνα 1.1: Φρυκτωρίες.....10
Εικόνα 1.2: Τηλέγραφοι.....10
Εικόνα 1.3:Τηλεφωνικές συσκευές από τα χρόνια του Μπέλ μέχρι τις αρχές του 20ου αιώνα.....11
Εικόνα 1.4: ENIAC.....12
Εικόνα 1.5: Τηλεφωνική εφαρμογή 3CX.....14
Εικόνα 1.6: Ένα τηλέφωνο USB. 14
Εικόνα 1.7: Ένα τηλέφωνο SIP υλισμικού.....15
Εικόνα 5.1: Μοντέλο VCR. 67
Εικόνα 6.1: ΕΓΓΡΑΦΗ -Βήμα 1.....79
Εικόνα 6.2: ΕΓΓΡΑΦΗ -Βήμα 2.....80
Εικόνα 6.3: ΕΓΓΡΑΦΗ -Βήμα 3.....80
Εικόνα 6.4: ΕΓΓΡΑΦΗ - Επιτυχής Εγγραφή. 81
Εικόνα 6.5: ΕΓΓΡΑΦΗ -Ανεπιτυχής Εγγραφή. 81
Εικόνα 6.6: ΑΠΕΓΓΡΑΦΗ -Βήμα 1.....82
Εικόνα 6.7: ΑΠΕΓΓΡΑΦΗ -Βήμα 2.....83
Εικόνα 6.8: Έξοδος. 83
Εικόνα 6.9: Προβολή λεπτομερειών-Debug.....84
Εικόνα 6.10: Προβολή πληροφοριών Client.....84
Εικόνα 6.11: Προβολή.....85
Εικόνα 6.12: Είσοδος στις Επιλογές.....85
Εικόνα 6.13: Επιλογές πολυμεσικών συσκευών. 86
Εικόνα 6.14: Διαχείρηση λογαριασμών.....86
Εικόνα 6.15: ΔΙΑΧΕΙΡΗΣΗ ΕΠΑΦΩΝ(ΣΥΝΔΕΣΗ ΗΧΟΥ ΚΛΗΣΗΣ ΑΝΑ ΕΠΑΦΗ) 87
Εικόνα 6.16: Κλήση Βήμα 1.....87
Εικόνα 6.17: Κλήση Βήμα 2.....88
Εικόνα 6.18: Κλήση Βήμα 3.....88
Εικόνα 6.19: ΓΕΝΙΚΗ ΠΕΡΙΓΡΑΦΗ ΑΡΧΙΚΟΥ ΠΑΡΑΘΥΡΟΥ. 89

ΛΙΣΤΑ ΣΧΗΜΑΤΩΝ

Σχήμα 1.1: Αρχιτεκτονική τηλεφωνικού συστήματος. 15
Σχήμα 1.2: Μικρό VoIP δίκτυο. 16
Σχήμα 2.1: Ιεραρχικό μοντέλο του Sip 33
Σχήμα 2.2: Εγγραφή μέσω Registrar και κλήση. 44
Σχήμα 2.3: Κλήση με δύο Proxy Servers. 45
Σχήμα 2.4: Κλήση με Proxy και Redirect Servers. 47
Σχήμα 3.1 RTP δεδομένα σε ένα IP πακέτο. 49
Σχήμα 3.2 Επικεφαλίδα ενός RTP πακέτου. 50
Σχήμα 3.3: Δομή πακέτων Sender / Receiver reports. 56
Σχήμα 3.4: Source description 58
Σχήμα 3.5: Δομή BYE πακέτου. 59
Σχήμα 3.6: Δομή APP Πακέτο. 59
Σχήμα 5.1: Media proccesing model of JMF. 68
Σχήμα 5.2: Ιεραρχικό μοντέλο JMF. 68
Σχήμα 5.3: Μοντέλο JMF/RTP. 70
Σχήμα 5.4: RECEIVE. 71
Σχήμα 5.5: TRANSMIT. 72
Σχήμα 6.1: Παράδειγμα αρχείου contacts.txt 76
Σχήμα 6.2: Παράδειγμα ενός uaconfig.txt αρχείου. 77
Σχήμα 6.3: Παράδειγμα ενός SipStack_cl.txt αρχείου. 78
Σχήμα 6.4:Παράδειγμα αρχείου SipStack_serv.txt 90
Σχήμα 6.5: παράδειγμα αρχείου 192.168.2.2.5060_events.log 92
Σχήμα 6.6: παράδειγμα αρχείου 192.168.2.2.5060_messages.log 93
Σχήμα 6.7: Διάγραμμα ER της βάσης δεδομένων. 96
Σχήμα 6.8: Διάγραμμα σχήματος για το σχήμα της σχεσιακής βάσης δεδομένων Location Service 97
Σχήμα 7.1: Διάγραμμα κλάσεων του package Client. 101
Σχήμα 7.2: Διάγραμμα κλάσεων του package Server. 103

ΛΙΣΤΑ ΠΙΝΑΚΩΝ

Πίνακας 2.1: Επικεφαλίδες SIP Μηνυμάτων. 29
Πίνακας 3.1: Payload Types για ήχο. 52
Πίνακας 3.2: Payload Types για βίντεο. 52
Πίνακας 3.3: Τα είδη πληροφορίας ενός SDES/RTCP πακέτου. 55
Πίνακας 3.4: Οι τιμές του CNAME 58
Πίνακας 4.1: Payload Types. 64

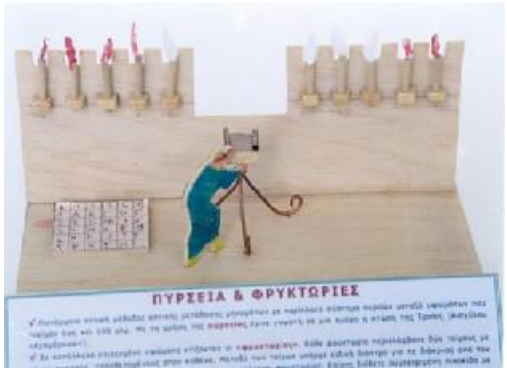
ΚΕΦΑΛΑΙΟ 1

ΕΙΣΑΓΩΓΗ

ΚΕΦΑΛΑΙΟ 1: ΕΙΣΑΓΩΓΗ

1.1 ΤΗΛΕΦΩΝΙΑ-ΙΣΤΟΡΙΚΗ ΑΝΑΔΡΟΜΗ

Από τα προϊστορικά ακόμα χρόνια ήταν φανερή η ανάγκη των ανθρώπων για επικοινωνία, η οποία μέχρι και τον Μεσαίωνα βασιζόταν σε δύο βασικά είδη τηλεπικοινωνιών: την οπτική τηλεπικοινωνία και την ακουστική. Η ανάγκη του ανθρώπου για πιο άμεσους και εξελιγμένους τρόπους επικοινωνίας οδήγησε στην εμφάνιση του παλιότερου γνωστού συστήματος ψηφιακής τηλεπικοινωνίας τις Φρυκτωρίες.



Εικόνα 1.1: **Φρυκτωρίες**

Οι Φρυκτωρίες χρησιμοποιήθηκαν στην αρχαία Ελλάδα για την μεταφορά μηνυμάτων σε μεγάλες χιλιομετρικές αποστάσεις. Τα μηνύματα μεταδίδονταν με αναμμένους δαυλούς κατά την διάρκεια της νύχτας ή με πυκνούς καπνούς κατά την διάρκεια της ημέρας. Υπάρχουν γραπτές μαρτυρίες σύμφωνα με τις οποίες, οι αρχαίοι Έλληνες χρησιμοποιούσαν πυρσούς φωτιάς για να μεταδώσουν προσυμφωνημένα μηνύματα. Παράδειγμα αποτελεί η είδηση για την είσοδο του Δούρειου Ίππου στην Τροία από τον Σίνωνα προς τον Αγαμέμνονα και το μήνυμα με πυρσό που έστειλε ο Αγαμέμνονας προς τον ελληνικό στόλο στην Τένεδο, δίνοντάς του το σήμα της επιστροφής και κατάληψης της ανοχύρωτης Τροίας.

Εφευρέσεις όπως το ακουστικό κέρας, ο οπτικός τηλέγραφος (ή πυρσεία), ο υδραυλικός τηλέγραφος και το σύστημα των φρυκτωριών έπαιξαν σημαντικό ρόλο στην εξέλιξη των τηλεπικοινωνιών.

Πολλοί, λοιπόν, ήταν εκείνοι που στα ύστερα χρόνια τις βελτίωσαν ή έκαναν εφευρέσεις βασισμένες πάνω σε αυτές. Αρκετά χρόνια αργότερα, με την βιομηχανική επανάσταση η ανάγκη για ένα γρήγορο και αξιόπιστο μέσο επικοινωνίας είχε γίνει πλέον επιτακτική. Έτσι δεν άργησε να εμφανιστεί ο σπουδαιότερος πρόδρομος του τηλεφώνου ο τηλέγραφος (Εικόνα 1.2).

Εικόνα 1.2: **Τηλέγραφοι**

Η ιδέα του τηλεγράφου αν και προέρχεται, όπως είδαμε προηγουμένως, από τα αρχαία χρόνια υλοποιήθηκε το 1774 από τον Ελβετό George Luis που κατασκεύασε μια πρώιμη μορφή τηλεγράφου, αργότερα εμφανίστηκαν οι τηλέγραφοι του Semmering (1810), του Ampere και των Cooke και Wheaton. Ο Αμερικανός, όμως, Samuel Morse (1791-1872) το 1837 παρουσίασε τον τηλέγραφό του που είχε την δυνατότητα να μεταδίδει μηνύματα σε πολύ μακρινές αποστάσεις γρήγορα και χωρίς μεγάλο κόστος. Το πρώτο μήνυμα από αυτόν τον τηλέγραφο στάλθηκε το 1844 από την Ουάσιγκτον στην Βαλτιμόρη.

Εικόνα 1.3:**Τηλεφωνικές συσκευές από τα χρόνια του Μπέλ μέχρι τις αρχές του** 20**ου αιώνα**.

Το έτος 1875 επαναλήφθηκε η ιστορία με τον Morse και τον τηλέγραφο: Ένας φαινομενικά άσχετος με την Τεχνική, ο καθηγητής φυσιολογίας της φωνής, Alexander Graham Bell (Μπελ, 1847-1922), παρουσίασε ένα σύστημα τηλεμετάδοσης της ανθρώπινης ομιλίας, ένα τηλέφωνο(Εικόνα 1.3). Για την ακρίβεια, ο Μπελ επινόησε τον ηλεκτρομαγνητικό μετατροπέα ήχου· στο στενό σημείο ενός χωνιού τοποθέτησε μια λεπτή μεταλλική μεμβράνη και ακριβώς δίπλα της βρισκόταν ένα πηνίο, τυλιγμένο σε μία μαγνητική ράβδο. Οι ταλαντώσεις της μεμβράνης από τα ηχητικά κύματα παρήγαγαν στο πηνίο ασθενή ηλεκτρική τάση. Στην άλλη άκρη η ίδια διάταξη λειτουργούσε ως μεγάφωνο. Οι μεταβολές του ρεύματος στο πηνίο προκαλούσαν ταλαντώσεις στη μεμβράνη, η οποία δημιουργούσε έτσι ηχητικά κύματα.

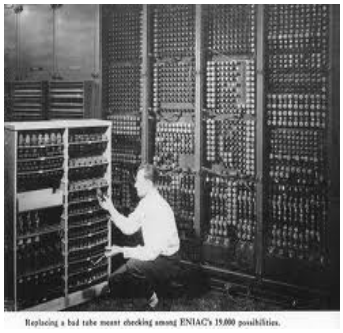
Μετά την εφεύρεση όμως του μικροφώνου από τον Αμερικανό Χίγκες το 1877, το τηλέφωνο άρχισε να εξελίσσεται και να χρησιμοποιείται για τη σύνδεση μακρινών αποστάσεων. Το μικρόφωνο αυτό περιλάμβανε μικρή ράβδο από άνθρακα η οποία περιβαλλόταν από δυο στρώματα άνθρακα. Στην αρχή μικρόφωνο και ακουστικό ήταν τοποθετημένα μαζί. Το τηλέφωνο πέρασε διάφορες εξελίξεις για να φτάσει στη σημερινή του μορφή.

Η εφεύρεση του Μπελ ήταν η αρχή για μία ραγδαία εξέλιξη της τηλεφωνίας: Ήδη το έτος 1878 ιδρύθηκε στο New Haven των ΗΠΑ το πρώτο δημόσιο τηλεφωνικό δίκτυο με 21 συνδρομητές. Το 1881 εγκαταστάθηκε το πρώτο τηλεφωνικό δίκτυο της Ευρώπης στο 'Αμστερνταμ με 49 συνδρομητές. Στα πρώτα χρόνια λειτουργούσαν όλα τα τηλεφωνικά δίκτυα με κλήση στο κέντρο και παραγγελία της σύνδεσης. Συγκεκριμένα, αν ήθελε κάποιος να τηλεφωνήσει, σήκωνε το ακουστικό και γύριζε τη μανιβέλα του επαγωγέα. Με αυτό τον τρόπο έφτανε ένα σήμα στην τηλεφωνήτρια, η οποία μετέτρεπε την παραγγελία για επικοινωνία με κάποιο συνδρομητή σε σύνδεση στον κεντρικό τηλεφωνικό πίνακα. Η αυτόματη τηλεφωνία άρχισε να λειτουργεί κάποια χρόνια αργότερα και διαδόθηκε σταδιακά. Από το 1895 λειτούργησε η πρώτη διεθνής σύνδεση μεταξύ Ολλανδίας και Βελγίου.

Στην Ελλάδα το πρώτο τηλεφωνικό κέντρο κατασκευάστηκε το 1931 και με σύμβαση που έκανε το κράτος με τη γερμανική εταιρεία Siemens κατασκευάστηκαν τα πρώτα αυτόματα τηλεφωνικά κέντρα. Στις 23 Οκτωβρίου του 1949 λαμβάνει χώρα η πρώτη επίσημη κρατική εισαγωγή της τηλεφωνίας στην Ελλάδα. Είναι η μέρα όπου ιδρύεται ο Οργανισμός Τηλεπικοινωνιών Ελλάδας - Ο.Τ.Ε. και 10 Νοεμβρίου 1949 γίνονται τα επίσημα εγκαίνια των εργασιών του Ο.Τ.Ε. Το 1965 αυτοματοποιείται το υπεραστικό τηλεφωνικό δίκτυο της χώρας και το 1968 ποντίζεται το υποβρύχιο καλώδιο Ελλάδας - Ιταλίας, MED-3. Το 1970 τοποθετείται η πρώτη κεραία του Κέντρου Δορυφορικών Επικοινωνιών Θερμοπυλών (η 6η στην Ευρώπη) και στις 20 Νοεμβρίου 1989 λειτουργεί στην Πάτρα το πρώτο πλήρες ψηφιακό τηλεφωνικό κέντρο του συστήματος AXE-10/ERICSSON.

1.2 ΑΠΟ ΤΑ ΑΝΑΛΟΓΙΚΑ ΣΤΑ ΨΗΦΙΑΚΑ ΣΥΣΤΗΜΑΤΑ ΕΠΙΚΟΙΝΩΝΙΩΝ

Καθώς το τηλέφωνο εξελισσόταν και έπαιρνε μια ολοένα και σημαντικότερη θέση στην καθημερινότητα του απλού ανθρώπου, είχαν ήδη αρχίσει να γίνονται τα πρώτα βήματα για την μια από τις σημαντικότερες ανακαλύψεις του 20^{ού} αιώνα το Διαδίκτυο (Internet) .



Εικόνα 1.4: ENIAC.

Το 1946, μετά το τέλος του Β' Παγκοσμίου Πολέμου, οι Ηνωμένες Πολιτείες χρειάζονταν μια συσκευή η οποία να βοηθά τους στρατιωτικούς στους υπολογισμούς για να βρίσκουν τα όπλα τους το στόχο με μεγαλύτερη ακρίβεια. Για πρώτη φορά δημιουργήθηκε ένα τεράστιο μηχάνημα που αντί για μηχανικά μέρη χρησιμοποιούσε ηλεκτρονικές λυχνίες , κατασκευασμένες από τον Λι Ντε Φορέ (Lee DeForest). (Εικόνα1.4) Ο πρώτος ηλεκτρονικός υπολογιστής ονομάστηκε ENIAC (Electronic Numerical Integrator And Calculator). Ο ENIAC που κατασκευάστηκε από τους Μόκλι και Έκερτ ήταν τεράστιος σε μέγεθος (καταλάμβανε έναν ολόκληρο όροφο), και έπρεπε να τον ελέγχουν συνεχώς ειδικοί επιστήμονες. Συχνά, επίσης, καίγονταν οι λυχνίες του και έπρεπε να τις αντικαθιστούν. Ακόμα και ο πιο ταπεινός σημερινός υπολογιστής είναι χιλιάδες φορές καλύτερος από τον ENIAC ως προς τις δυνατότητες. Ήταν, όμως, η πρώτη σοβαρή προσπάθεια δημιουργίας υπολογιστικής μηχανής. Τα χρόνια που ακολούθησαν χαρακτηρίστηκαν από μια συνεχή εξέλιξη των υπολογιστών όπως:

- 1^{ή} Γενιά Υπολογιστών (1946-1956) στην οποία δημιουργήθηκε για πρώτη φορά ένα τεράστιο μηχάνημα που αντί για μηχανικά μέρη χρησιμοποιούσε ηλεκτρονικές λυχνίες (κατασκευή ENIAC).

- 2^{ή} Γενιά υπολογιστών (1956-1963) την περίοδο αυτή οι λυχνίες αντικαθίστανται από τρανζίστορ. Οι ηλεκτρονικές αυτές κατασκευές (κρυσταλλοτριόδοι, όπως τις

ονομάζουν οι ηλεκτρονικοί), επιτρέπουν τη δημιουργία μικρότερων και ταχύτερων υπολογιστών. Το 1956 στο Ίδρυμα Τεχνολογίας της Μασαχουσέτης (M.I.T.) κατασκευάστηκε ο πρώτος Ηλεκτρονικός Υπολογιστής που λειτουργούσε με τρανζίστορ, ο TX-0.

- 3ή Γενιά υπολογιστών (1964-1971) Το 1958, ο Jack Kilby της εταιρείας Texas Instruments, κατάφερε να δημιουργήσει κάτι που θα άλλαζε τον κόσμο των ηλεκτρονικών για πάντα. Κατασκεύασε το πρώτο Ολοκληρωμένο Κύκλωμα συνδυάζοντας τρανζίστορ, πυκνωτές, αντιστάτες και άλλα ηλεκτρονικά εξαρτήματα όλα τοποθετημένα στο ίδιο κομμάτι από πυρίτιο. Το δημιούργημα του Kilby επέτρεψε στους επιστήμονες να κατασκευάσουν υπολογιστές τόσο μικρούς ώστε να μπορούμε ακόμη και να τους μεταφέρουμε.

- 4ή Γενιά υπολογιστών (1971-Σήμερα) Οι υπολογιστές που έχουμε σήμερα ανήκουν στην 4η Γενιά. Ο κάθε ένας από αυτούς είναι εφοδιασμένος με Επεξεργαστή (CPU), έχει τη δική του Μνήμη, μονάδα αποθήκευσης πληροφοριών, οθόνη, και κάποιο είδος μέσου για να δίνουμε πληροφορίες στον υπολογιστή (πληκτρολόγιο, πενάκι, ποντίκι κλπ). Σύμφωνα με το νόμο του Moore, κάθε 18 περίπου μήνες, η ισχύς των παραγόμενων υπολογιστών διπλασιάζεται. Έτσι, γίνεται αντιληπτό γιατί ένας υπολογιστής που αγοράζεται σήμερα είναι (περίπου) δύο φορές ταχύτερος από έναν υπολογιστή της ίδιας "κατηγορίας" που αγοράστηκε πριν ενάμιση χρόνο.

Παράλληλα με την παραπάνω εξέλιξη των υπολογιστών σημειώθηκε ταυτόχρονα και ο εκσυγχρονισμός των ενσύρματων(από το καλώδιο σύστροφου ζεύγους στην οπτική ίνα) και ασύρματων(από την εκπομπή και λήψη ραδιοκυμάτων στην εκτόξευση δορυφόρων και επικοινωνία μέσω μικροκυμάτων) δικτύων.

Η ραγδαία εξέλιξη των ηλεκτρονικών υπολογιστών και δικτύων τον περασμένο αιώνα έκανε πρόσφορο το έδαφος για την εισαγωγή σε πιο εξελιγμένες μορφές επικοινωνίας όπως ηλεκτρονικό ταχυδρομείο(e-mail), εφαρμογές άμεσων μηνυμάτων(Instant Messaging), συνδιάλεξη μέσω βίντεο(Video Conference), με την αντικατάσταση των αναλογικών τηλεφωνικών συσκευών σε ψηφιακά συστήματα επικοινωνίας.

Στην παρούσα πτυχιακή εργασία αναλύεται η τηλεφωνία μέσω διαδικτύου (IP Telephony) και συγκεκριμένα η ανάλυση του SIP πρωτοκόλλου σηματοδοσίας και ανάπτυξη VoIP και VVoIP εφαρμογής με χρήση του SIP. (βλέπε παρακάτω)

1.3 TI EINAI VOIP-VVOIP

Το Voice over IP ή VoIP (VVOIP **ή** Video and Voice over IP αποτελεί την επέκταση του VoIP ώστε να περιλαμβάνει και την αποστολή εικόνας) ή τηλεφωνία μέσω διαδικτύου ή σωστότερα **ΦεΔΠ** δηλαδή "Φωνή επί διαδικτυακού πρωτοκόλλου", χαρακτηρίζει μια ομάδα πρωτοκόλλων-τεχνολογιών (H.323, SIP), η οποία προσφέρει φωνητική συνομιλία σε πραγματικό χρόνο με σχετικά καλή ποιότητα πλέον και στην ουσία χωρίς κόστος.

Εικόνα 1.5: **Τηλεφωνική εφαρμογή** 3CX.

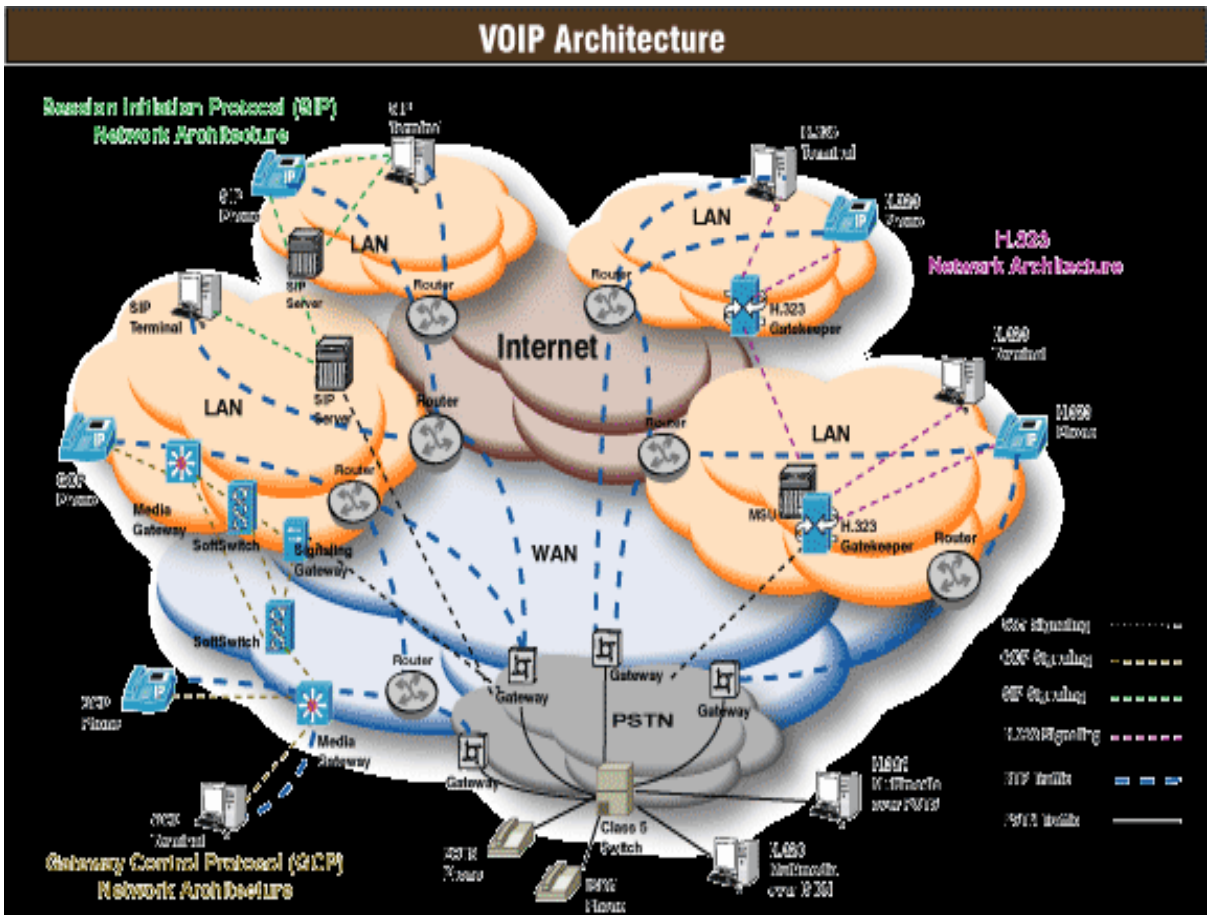
Οι συνομιλίες αυτές παραδοσιακά γίνονταν αποκλειστικά μέσω προσωπικών υπολογιστών που ήταν συνδεδεμένοι με το Διαδίκτυο (Internet) και διέθετε μικρόφωνο, ακουστικά και το κατάλληλο λογισμικό. Η κλήση κατέληγε σε ένα άλλο, ανάλογα εξοπλισμένο, προσωπικό υπολογιστή χωρίς να υπάρχει κάποια επιπλέον χρέωση, εκτός από αυτή της πρόσβασης στο Διαδίκτυο, αφού στη συγκεκριμένη επικοινωνία δεν μεσολαβεί κάποιος παραδοσιακός φορέας τηλεπικοινωνιών (π.χ. ΟΤΕ) παρά μόνο το Διαδίκτυο.Για την πραγματοποίηση κλήσεων μέσω VoIP, ο χρήστης χρειάζεται ένα πρόγραμμα τηλεφώνου (Εικόνα1.5) με βάση λογισμικό ή ένα τηλέφωνο VoIP με βάση υλισμικό (εικόνα 1.7).Οι τηλεφωνικές κλήσεις μπορούν να πραγματοποιηθούν προς οποιοδήποτε

Εικόνα 1.6: **Ένα τηλέφωνο** USB.

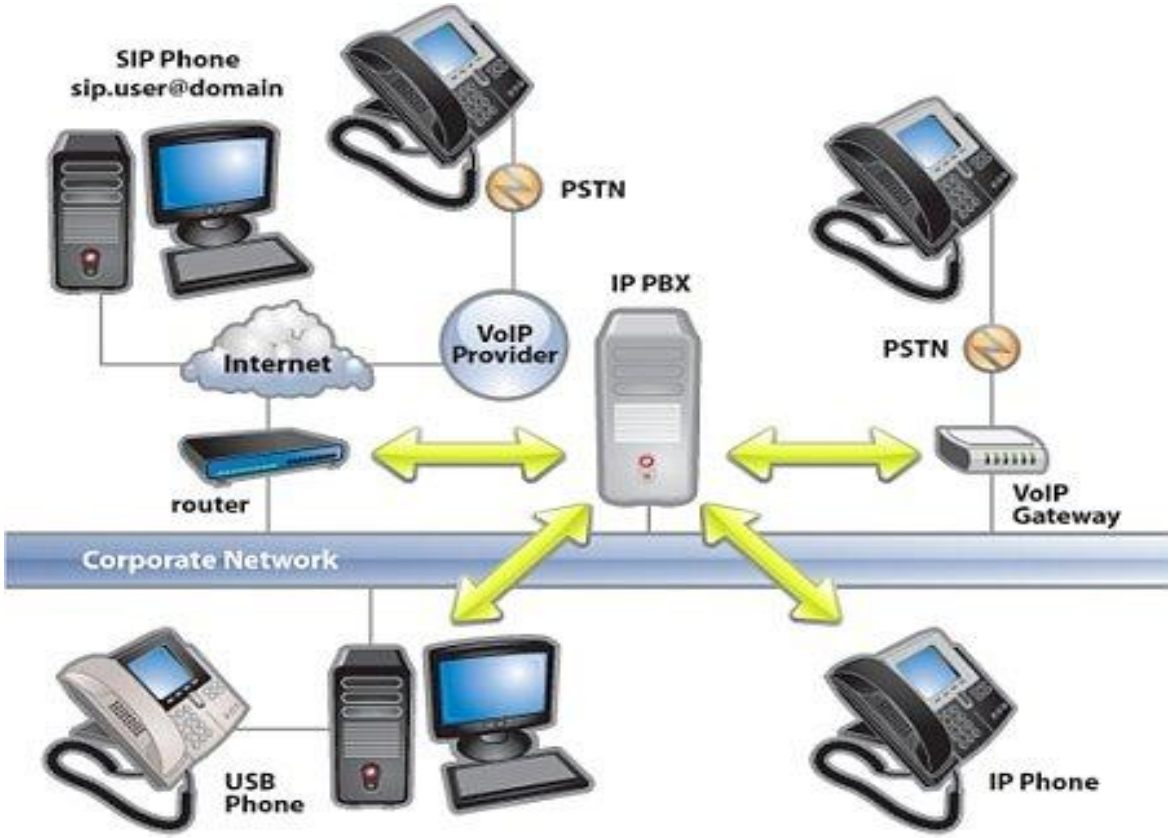
προορισμό/ άτομο: προς αριθμούς VoIP, καθώς και προς άτομα που διαθέτουν κανονικούς αριθμούς τηλεφώνου. Τον τελευταίο καιρό έχουν εμφανιστεί οι λεγόμενοι εναλλακτικοί τηλεπικοινωνιακοί φορείς, οι οποίοι προσφέρουν προώθηση των κλήσεων VoIP σε σταθερά δίκτυα τηλεπικοινωνιών σε εξαιρετικά χαμηλό κόστος, αλλά όχι το αντίστροφο. Μερικοί εξ αυτών έχουν παρουσιάσει και ειδικές τηλεφωνικές συσκευές USB VoIP(εικόνα 1.6), οι οποίες συνεργάζονται με το αντίστοιχο λογισμικό στον Η/Υ και καθιστούν τις κλήσεις μέσω Διαδικτύου σαφώς πιο λειτουργικές.

Εικόνα 1.7: **Ένα τηλέφωνο** SIP **υλισμικού**.

Στα σχήματα 1.1 και 1.2 μπορούμε να δούμε την αρχιτεκτονική ενός μητροπολιτικού VoIP συστήματος και ενός μικρού γραφείου με IP-PBX.



Σχήμα 1.1: **Αρχιτεκτονική τηλεφωνικού συστήματος**.



Σχήμα 1.2: **Μικρό** VoIP **δίκτυο**.

1.4 ΣΤΟΙΒΑ ΠΡΩΤΟΚΟΛΛΩΝ VOIP

Τα πιο συχνά αναφερόμενα VoIP πρωτόκολλα είναι τα πρωτόκολλα σηματοδοσίας (signaling protocols). Σε VoIP δίκτυα κάνουμε χρήση αυτών των πρωτοκόλλων για να εντοπίσουμε τη συσκευή στο άλλο άκρο της επικοινωνίας, και στη συνέχεια να διαπραγματευτεί την ανταλλαγή μεταξύ των συσκευών αποστολής και λήψης. Τα δύο πιο γνωστά πρωτόκολλα σηματοδοσίας είναι:

- Session Initiation Protocol (SIP), που ορίζεται από το Internet Engineering Task Force (IETF).
- H.323, που ορίζεται από τη Διεθνή Ένωση Τηλεπικοινωνιών (ITU).

Αυτά τα δύο πρωτόκολλα βασικά κάνουν το ίδιο πράγμα και οι περισσότερες συσκευές VoIP χρησιμοποιούν το ένα ή το άλλο. Τα δύο πρωτόκολλα εργάζονται με διαφορετικό τρόπο για να επιτευχθεί η δημιουργία μιας VoIP σύνδεσης, το SIP είναι ASCII-based και το H.323 είναι Binary-based. Αν και το H.323 ήταν πιο δημοφιλές στην αρχή, και πολλοί θεωρούν ότι είναι ανώτερο στην ικανότητά του να συνεργαστεί με το δημόσιο τηλεφωνικό δίκτυο μεταγωγής (PSTN) και για τη μετάδοση βίντεο, το SIP έχει γίνει ολοένα και πιο δημοφιλής λόγω της υποστήριξής του από πολλούς κατασκευαστές συσκευών VoIP.

Το H.323 είναι μια σουίτα που αποτελείται από μια σειρά από πολλά και διάφορα πρωτόκολλα που εκτελούν ειδικά καθήκοντα μαζί. Ορισμένα μέλη της σουίτας περιλαμβάνουν:

- H.225.0, το οποίο ορίζει τη σύνδεση.
- H.332, το οποίο χρησιμοποιείται για μεγάλες συνεδρίες.
- H.235, το οποίο παρέχει λειτουργίες ασφάλειας και αυθεντικότητας.
- H.245, το οποίο διαπραγματεύεται την χρήση του καναλιού.
- RAS, που χειρίζεται τα μηνύματα καταχώρισης, εισαγωγής, και κατάστασης (registration, admission, and status messages).

Μια VoIP πύλη (VoIP gateway) συνδέει μια διεύθυνση IP στο PSTN δίκτυο ή σε ένα αναλογικό τηλέφωνο. Οι VoIP πύλες έχουν δύο μέρη:

- Το media gateway controller (MGC).
- Το media gateway (MG).

Ένα άλλο σύνολο πρωτοκόλλων, που ονομάζεται πρωτόκολλα συσκευής ελέγχου, ξεχωρίζουν λογικά το κομμάτι του ελέγχου κλήσης από το λογικό κομμάτι της επεξεργασίας πολυμέσων στις VoIP πύλες. Παραδείγματα αυτών των πρωτοκόλλων αποτελούν τα παρακάτω:

- Media Gateway Control Protocol (MGCP).
- H.248 (γνωστό ως Media Gateway Controller, or Megaco).

Χρησιμοποιεί έναν call agent που κατευθύνει και ελέγχει το MG και την σηματοδοσία της πύλης. The MGC uses MGCP to find the locations and capabilities of the VoIP endpoints. Το MGC χρησιμοποιεί το MGCP για να βρεί τις θέσεις και τις παραμέτρους των VoIP τερματικών.

Η IETF χρησιμοποιεί το όνομα Megaco, και η ITU το όνομα H.248 για το ίδιο πρωτόκολλο το οποίο δημιουργήθηκε από κοινού από τους δύο οργανισμούς. Το MGCP έχει σχεδιαστεί για να παρέχει εξ αποστάσεως έλεγχο της VoIP πύλης και άλλες συσκευές συνόδου. Τα MGCP και Megaco είναι παρόμοια, αλλά το Megaco υποστηρίζει περισσότερα είδη δικτύων, καθώς και τα δίκτυα ATM.

Τα VoIP δίκτυα βασίζονται σε μια κεντρική αρχιτεκτονική και συνήθως χρησιμοποιούν τα Megaco και MGCP. Το MGC/call agent είναι η κεντρική συσκευή που επικοινωνεί με τις media πύλες. Τα δίκτυα που βασίζονται σε μια κατανεμημένη αρχιτεκτονική χρησιμοποιούν τα SIP και H.323.

Το RTP είναι ένα πρωτόκολλο για τη μετάδοση ήχου και βίντεο σε IP δίκτυα. Στο RFC 3550 ορίζεται το RTP και λειτουργεί σε συνδυασμό με το SIP ή το H.323. Μια κλήση VoIP χρησιμοποιεί δύο RTP ρεύματα, ένα προς κάθε κατεύθυνση.

Στο RTP συνήθως ορίζονται οι πόρτες μεταξύ 16384 και 32767, αλλά δεν υπάρχει κάποιο πρότυπο που να ορίζει τις πόρτες για το RTP.Επίσης το RTP δεν προβλέπει ποιότητα υπηρεσιών (Quality of Service-QoS). Το RTP συνεργάζεται με το πρωτόκολλο ελέγχου (RTPC), το οποίο παρέχει πληροφορίες για τον έλεγχο των RTP επικοινωνιών. Το RTP διαχειρίζεται την μεταβίβαση των δεδομένων το ίδιο και το RTPC μπορεί να συλλέξει πληροφορίες (τα πακέτα που αποστέλλονται, τα πακέτα που χάθηκαν, κλπ.) που αφορούν την ποιότητα υπηρεσιών (QoS).

Το Secure Real Time Transport Protocol (SRTP) παρέχει τις λειτουργίες κρυπτογράφησης, γνησιότητας και ακεραιότητας των δεδομένων. Το Secure RTCP (SRTCP) παρέχει τις ίδιες υπηρεσίες ασφαλείας για το RTPC. Τα SRTP και SRTCP κάνουν χρήση του Advanced Encryption Standard (παλαιότερα γνωστή ως Rijndael), το οποίο εγκρίθηκε από την κυβέρνηση των ΗΠΑ να αντικαταστήσει το Data Encryption Standard.

1.5 ΤΗΛΕΔΙΑΣΚΕΨΗ (VIDEOCONFERENCE)

Ως τηλεδιάσκεψη ή βίντεο-διάσκεψη (videoconference–videoteleconference-visual collaboration) νοείται η από απόσταση διάσκεψη, η αμφίδρομη δηλαδή επικοινωνία μεταξύ δύο ή περισσότερων ομάδων ή ατόμων μέσω συστημάτων ήχου, βίντεο ή υπολογιστών. Με την τηλεδιάσκεψη δίνεται η δυνατότητα σε ένα ή περισσότερους απομακρυσμένους χρήστες, να συμμετέχουν σε μία διάσκεψη χωρίς να είναι απαραίτητη η φυσική τους παρουσία στον χώρο συνεδρίασης.Η τηλεδιάσκεψη μπορεί να πραγματοποιηθεί με τρεις τρόπους:

- Η από σημείο σε σημείο τηλεδιάσκεψη (Point to Point Conferencing Audio και Video) απευθύνεται στην επικοινωνία δύο χρηστών και αρκεί μια απλή σύνδεση σε κάποιο δίκτυο. Αυτή είναι και η πιο απλή μορφή τηλεδιάσκεψης.

- Η τηλεδιάσκεψη πολλαπλών σημείων (Multipoint Conferencing Audio και Video) όπου πολλοί χρήστες συνδέονται σε κάποιο κεντρικό εξυπηρετητή (multipoint conference unit – MCU) που αναλαμβάνει να μεταδώσει την επικοινωνία από όλους σε όλους. Στην τηλεδιάσκεψη πολλαπλών σημείων έχουμε ηχητική και οπτική συνδιάσκεψη μεταξύ περισσότερων από 2 σταθμών εργασίας και την ολοκλήρωσή τους στο περιβάλλον μιας εικονικής αίθουσας. Όλοι οι σταθμοί λαμβάνουν εικόνα και ήχο από όλους. Χωροταξικά οι σταθμοί είναι σε διαφορετικές τοποθεσίες και η οπτική επικοινωνία μεταξύ τους προσομοιώνει κάθε λειτουργία ανάλογη με τις λειτουργίες που διεξάγονται σε πραγματικές αίθουσες διασκέψεων.

- Η τηλεδιάσκεψη από σημείο σε πολλά σημεία (Point to Multipoint Conferencing Audio και Video), όπου ένα σύνολο χρηστών παρακολουθεί την μετάδοση ενός και μόνο χρήστη χωρίς να επιτρέπεται αμφίδρομη επικοινωνία. Στην τηλεδιάσκεψη σημείου προς πολλαπλά σημεία έχουμε ηχητική και οπτική συνδιάσκεψη από ένα σταθμό σε άλλους σταθμούς. Στην περίπτωση αυτή ο πομπός βρίσκεται σε ένα σταθμό και οι διάφορες ομάδες ληπτών στους υπόλοιπους σταθμούς. Η ροή είναι μονόδρομη και οι σταθμοί των ληπτών λαμβάνουν την εικόνα και ήχο από το σταθμό του πομπού χωρίς να μεταδίδουν την δική τους εικόνα και ήχο. Παράδειγμα του τρόπου αυτού τηλεδιάσκεψης αποτελεί ο τρόπος με τον οποίο δουλεύει η τηλεόραση.

Εκτός των παραπάνω κατηγοριών μπορούμε να διαχωρίσουμε περαιτέρω τους τρόπους τηλεδιάσκεψης όπως:

- Στην τηλεδιάσκεψη με εικόνα, όπου αποτελεί οπτική και ηχητική επικοινωνία πραγματικού χρόνου μεταξύ ατόμων που βρίσκονται σε διαφορετικές τοποθεσίες. Χρησιμοποιείται από μία ή περισσότερες ομάδες που επικοινωνούν με άλλα άτομα προς ανταλλαγή ιδεών, απόκτηση πληροφοριών, σύγχρονη εκπαίδευση και διαχείριση οργανισμών. Η τηλεδιάσκεψη συγκεκριμένα συνδυάζει:
    - εικόνες video (των συμμετεχόντων κατά την διάρκειας της σύσκεψης)
    - υψηλής ποιότητας ήχου επικοινωνία (μουσική, ήχοι, φωνή)

  Ο απαιτούμενος εξοπλισμός αποτελείται από μία βιντεοκάμερα, ένα μόνιτορ και κάποια στοιχεία που επιτρέπουν τον έλεγχο τους είναι απαραίτητα για την αποστολή και λήψη πληροφοριών. Τα συστήματα τηλεδιάσκεψης βρίσκουν εφαρμογή στη σύγχρονη τηλε-εκπαίδευση και τηλε-συνεργασία. Βασικά πλεονεκτήματα των συστημάτων τηλεδιάσκεψης είναι το ότι επιτρέπουν οπτική επικοινωνία, παρέχουν την δυνατότητα στους εκπαιδευόμενους για φυσικές και αυθόρμητες αντιδράσεις, ενώ δίνουν τη δυνατότητα της παρακολούθησης εκπαιδευτικών προγραμμάτων σε άτομα που βρίσκονται σε απομακρυσμένες τοποθεσίες. Μειονεκτήματα αυτών των συστημάτων αποτελεί το μεγάλο κόστος του εξοπλισμού και χρήσης.

- Στην τηλεδιάσκεψη μόνο με ήχο, όπου αποτελεί την ταυτόχρονη σύνδεση πολλών διαφορετικών ατόμων μέσω τηλεφωνικών γραμμών, γραμμών ISDN ή εφαρμογών του Internet. Με τον τρόπο αυτό μια ομάδα ατόμων μπορεί να επικοινωνήσει μεταξύ των μελών της. Όσο αφορά τις δικτυακές απαιτήσεις αυτές διακρίνονται σε:
    - Απαιτήσεις χωρητικότητας γραμμής που είναι από 16 έως 64 Kbps ανά συνεδρία ανάλογα με το σχήμα κωδικοποίησης ήχου που ακολουθείται.
    - Απαιτήσεις στο μικρό χρόνο απόκρισης (<< 0,2 sec) που απαιτείται για την αλληλεπίδραση μεταξύ των χρηστών.

Στην πιο απλή της μορφή η τηλεδιάσκεψη μπορεί να γίνει μέσω τηλεφώνου, με χρήση συσκευής ανοικτής ακρόασης. Στην περίπτωση αυτή δίνεται σε όλους τους συμμετέχοντες ένας κοινός αριθμός, τον οποίο καλούν την ώρα που έχει ορισθεί ως χρονικό σημείο έναρξης της συνεδρίας. Ο εξοπλισμός που απαιτείται για την ανωτέρω μορφή διάσκεψης είναι απλός και χαμηλού κόστους Η πλέον προηγμένη τεχνολογικά μορφή τηλεδιάσκεψης, είναι αυτή που παρέχει ήχο και εικόνα, μέσω οθόνης που εγκαθίσταται στο χώρο που βρίσκεται ένας, ή και περισσότεροι εκ των συμμετεχόντων στη διάσκεψη. Η οθόνη αυτή μπορεί να είναι μικρή (εικονο-τηλέφωνο) ή μεγαλύτερη (οθόνη υπολογιστή με εγκατεστημένη κάμερα, ή μεγάλη οθόνη τηλεδιασκέψεων).

Ο εξοπλισμός που χρειαζόμαστε για μια επιτυχημένη τηλεδιάσκεψη ποικίλη ανάλογα με την ποιότητα σε εικόνα και ήχο, την ταχύτητα και τον αριθμώ των ατόμων που μπορούν να λάβουν μέρος. Μερικά βασικά στοιχεία του εξοπλισμού είναι:

- Βιντεοκάμερα για να συλλάβει το βίντεο του ομιλητή. Διαφέρει ανάλογα με την ποιότητα λήψης ,από το αν είναι κινητή (PTZ) ή σταθερή, ο φακός να έχει ευρεία γωνία και όχι στενή, να έχει αυτόματη αυξομείωση των χρωμάτων και της φωτεινότητας κ.α.
- Συσκευές απεικόνισης όπου θα εμφανίζεται το βίντεο του άλλου ομιλητή. Μπορεί να είναι βιντεοπροβολέας (projector) ή οθόνες με διάφορα μεγέθη (14'',20'',30'')και διαφορετική ποιότητα αναπαράστασης.
- Ηχεία για να ακούγεται ο ήχος από τον εκάστοτε ομιλητή και Μικρόφωνο για να συλλάβει τον ήχο του ομιλητή. Ο ήχος αποτελεί το σημαντικότερο κομμάτι σε μια τηλεδιάσκεψη λαμβάνοντας υπόψην ότι εάν χάσουμε το βίντεο ή έχουμε κακή εικόνα σε μια τηλεδιάσκεψη αυτό δεν μας περιορίζει να ολοκληρώσουμε την επικοινωνία μας, αντίθετα αν χαθεί ο ήχος τότε η επικοινωνία καθίσταται αδύνατη. Σύμφωνα με τα παραπάνω γίνεται κατανοητό πως τα μικρόφωνα θα πρέπει να έχουν κάποια συγκεκριμένα χαρακτηριστικά όπως: πλήρη αμφίδρομη (*full- duplex*) μετάδοση του ήχου, ακύρωση της ηχούς (*echo cancellation*), καταστολή θορύβου, και ικανότητα μίξης. Στα ηχεία επειδή κυρίως θα αναπαράγουν φωνή(εύρος φάσματος 300-4000Hz) το μόνο που χρειάζεται να προσέξουμε είναι η ένταση και η παρεμβολή τους στα μικρόφωνα.
- Κωδικοποιητής – αποκωδικοποιητής(Codecs) που είναι αρμόδιος για την συμπίεση - αποσυμπίεση των τηλεοπτικών και ακουστικών σημάτων για να είναι δυνατή η αποστολή τους μέσω των ακριβών δικτυακών συνδέσεων. Μερικά χαρακτηριστικά είναι: αν αποτελεί κομμάτι λογισμικού ή υλικό(τα υλικά Codec είναι γρηγορότερα κάτι που κάνει πιθανότερη την επικοινωνία σε πραγματικό χρόνο) και πόσα διαφορετικά codec υποστηρίζει ώστε να είναι δυνατή η επικοινωνία με διάφορα συστήματα τηλεδιάσκεψης.
- Λογισμικό κλήσης το οποίο είναι υπεύθυνο για την έναρξη, την συντήρηση και τον τερματισμό της τηλεδιάσκεψης καθώς και την αρμονική λειτουργία των παραπάνω συσκευών .Μερικά κρίσημα χαρακτηριστικά του λογισμικού σε μια εφαρμογή τηλεδιάσκεψης είναι:να είναι φιλικό προς τον χρήστη, να υποστηρίζει πλήθος των παραπάνω συσκευών, να είναι γρήγορο, φορητό(ανεξαρτησία πλατφόρμας) καθώς και να καλύπτει ικανοποιητικά όλα τα χαρακτηριστικά ποιότητας λογισμικού(λειτουργικότητα, ευχρηστία, συντηρησιμότητα, ελεγξιμότητα, αξιοπιστία κ.α.).

1.6 ΣΥΜΠΕΡΑΣΜΑΤΑ

Η επικοινωνία αποτελούσε και αποτελεί αναπόσπαστο κομμάτι της καθημερινής ζωής του ανθρώπου. Αυτό συνετέλεσε στην αλματώδη ανάπτυξη των επικοινωνιών και ιδιαίτερα της τηλεφωνίας τον τελευταίο αιώνα. Έτσι από τις Φρυκτωρίες της Αρχαίας Ελλάδας περάσαμε στον τηλέγραφο και το τηλέφωνο. Η επιστήμη των υπολογιστών ήταν η κατάλληλη για να υποστηρίξει το τηλέφωνο τα επόμενα χρόνια και να το εξελίξει από μια απρόσωπη συσκευή επικοινωνίας ,κυρίως δύο προσώπων, σε ένα ολοκληρωμένο σύστημα επικοινωνιών που κάνει απαραίτητη την χρήση του όχι μόνο σε κοινωνικό επίπεδο αλλά και επιχειρησιακό-οικονομικό, πολιτικό.

Σήμερα η τηλεδιάσκεψη αποτελεί τον αντικαταστάτη του απλού τηλεφώνου παρέχοντας έναν αμεσότερο τρόπο επικοινωνίας και την δυνατότητα υπηρεσιών όπως: Μάθηση από απόσταση, Τηλε-εργασία, Τηλε-Συνεργασία. Τέλος, το μέλλον της τηλεδιάσκεψης αποτελεί η Τηλε-παρουσία (Telepresence).Το 2007 η εταιρία Cisco παρουσίασε ένα σύστημα τηλε-παρουσίας το οποίο επιτρέπει σε έναν ομιλητή κυριολεκτικά να διακτινιστεί ψηφιακά και να εμφανιστεί σε μια απομακρυσμένη αίθουσα ως τρισδιάστατο ολόγραμμα(Διαθέσιμο βίντεο στην ιστοσελίδα http://www.musion.co.uk/Cisco_TelePresence.html).

ΚΕΦΑΛΑΙΟ 2

ΤΟ SIP ΠΡΩΤΟΚΟΛΛΟ

ΚΕΦΑΛΑΙΟ 2: ΤΟ SIP ΠΡΩΤΟΚΟΛΛΟ

2.1 ΓΕΝΙΚΑ

Το Session Initiation Protocol (SIP) είναι ένα πρωτόκολλο σηματοδοσίας που χρησιμοποιείται για τη δημιουργία συνόδων σε ένα IP δίκτυο. Η σύνοδος θα μπορούσε να είναι μια απλή αμφίδρομη τηλεφωνική κλήση ή θα μπορούσε να είναι μια συνδιάσκεψη συνόδου πολυμέσων. Η ικανότητα για τη δημιουργία αυτών των συνόδων σημαίνει ότι μια σειρά από καινοτόμες υπηρεσίες μπορούν να καταστούν δυνατές ,όπως voice-enriched e-commerce, web page click-to-dial, Instant Messaging with buddy lists, και IP Centrex services.

Το SIP δημιουργήθηκε από την Internet Engineering Task Force (IETF-συγκεκριμένα από την MMUSIC Working Group) ,το όργανο που είναι υπεύθυνο για τη διαχείριση και την ανάπτυξη των μηχανισμών που συνθέτουν το διαδίκτυο. Αρχικά δημοσιεύτηκε το 1996 ως RFC 2543 και αργότερα το 2002 εξελίχθηκε ως στο RFC 3261 . Τα τελευταία δύο χρόνια, η Voice over IP κοινότητα ενέκρινε το SIP, ως το κύριο πρωτόκολλο σηματοδοσίας και συνεχίζει να εξελίσσεται και επεκτείνεται όπως ωριμάζει η τεχνολογία ενώ συγχρόνως κερδίζει συνεχώς έδαφος στην αγορά.

Το SIP διακρίνεται λίγο πολύ από αυτή την φιλοσοφία. Έχοντας αναπτυχθεί αποκλειστικά ως ένας μηχανισμός για τη θέσπιση συνεδριών, δεν γνωρίζει τις λεπτομέρειες της συνεδρίας παρά μόνο αρχίζει, τερματίζει και τροποποιεί συνεδρίες. Αυτό η απλότητα σημαίνει ότι το SIP διακρίνεται από:

- Την επεκτασιμότητά του
- Και προσαρμόζεται εύκολα σε διαφορετικές αρχιτεκτονικές και σενάρια..

Το SIP είναι ένα request-response (αίτημα-απόκριση) πρωτόκολλο που μοιάζει με δύο άλλα πρωτόκολλα του διαδικτύου ,τα HTTP και SMTP, με συνέπεια να ταιριάζει άνετα δίπλα σε άλλες Internet εφαρμογές. Χρησιμοποιώντας το SIP, η τηλεφωνία γίνεται άλλη μια Web εφαρμογή και ενσωματώνεται εύκολα σε άλλες υπηρεσίες του Internet. Το SIP είναι μια απλή εργαλειοθήκη όπου οι πάροχοι υπηρεσιών μπορούν να χρησιμοποιήσουν για την οικοδόμηση υπηρεσιών φωνής και πολυμέσων. Τέλος για να καταστεί δυνατή η τηλεφωνική επικοινωνία, το SIP χρειάζεται να συνεργαστεί με άλλα πρωτόκολλα όπως:

- Για την εξασφάλιση μεταφοράς (RTP/RTCP).
- Για την συμφωνία των παραμέτρων της κλήσης(SDP).
- Για τον έλεγχο ταυτότητας των χρηστών (ακτίνα, διάμετρος).
- Να παρέχουν καταλόγους (LDAP),
- Να είναι σε θέση να εγγυάται ποιότητα φωνής (RSVP, YESSIR) και διασυνεργασίας με τη σημερινή του τηλεφωνικού δικτύου.

2.2 SIP **ΟΝΤΟΤΗΤΕΣ**

Για να καταφέρει το SIP να υλοποιήσει τις απαραίτητες λειτουργίες για την εγκατάσταση συνόδων διατηρώντας παράλληλα τα ιδιαίτερα χαρακτηριστικά του(όπως η απλότητα και επεκτασιμότητα) είναι διαιρεμένο σε ένα σύνολο οντοτήτων (SIP Entities) από τις οποίες οι πιο κύριες είναι οι:

- Πράκτορες Χρήστη -USER AGENTS(UA).
- SIP Εξυπηρετητές -SIP SERVERS.

2.2.1 ΠΡΑΚΤΟΡΕΣ ΧΡΗΣΤΗ-USER AGENT

Οι User Agents εκκινούν και τερματίζουν συνόδους ανταλλάζοντας αιτήσεις (requests) και απαντήσεις(responses). Όπως υπονοείται από το όνομα, ένας User Agent παίρνει κάποια είσοδο από το χρήστη και λειτουργεί ως πράκτορας για λογαριασμό του χρήστη για την εγκατάσταση και τον τερματισμό διαλόγων με άλλους χρήστες. Κάθε UA πρέπει να διατηρεί πληροφορίες για την κατάσταση των κλήσεων, τις οποίες εκκινεί ή στις οποίες συμμετέχει. Οι πληροφορίες αυτές χρησιμοποιούνται και για την αποθήκευση των πληροφοριών για κάθε διάλογο, αλλά και για λόγους αξιοπιστίας.

Σύμφωνα με το RFC 3261 ο User Agent αποτελείται από έναν User Agent Client (UAC) και έναν User Agent Server (UAS).Ένας UAC μπορεί να δημιουργεί αιτήσεις οι οποίες προκύπτουν από έναν εξωτερικό παράγοντα (π.χ. το πάτημα από τον χρήστη ενός κουμπιού ή κάποιο σήμα στην τηλεφωνική γραμμή) και να επεξεργάζεται απαντήσεις. Ανάλογα ένας UAS είναι ικανός να λαμβάνει αιτήσεις και να δημιουργεί απαντήσεις οι οποίες μπορεί να προκύπτουν από τον χρήστη, το αποτέλεσμα της εκτέλεσης ενός προγράμματος ή μέσω ενός άλλου μηχανισμού. Η επεξεργασία της κάθε αίτησης γίνεται σε ατομικό επίπεδο και αν γίνει αποδεκτή τότε όλες οι αλλαγές κατάστασης που θα προκύψουν στον UAS πρέπει να εκτελεστούν αλλιώς καμία αλλαγή κατάστασης δεν πρέπει να γίνει.

2.2.2 PROXY SERVER

Ο Proxy Server ή αλλιώς Πληρεξούσιος Εξυπηρετητής είναι η οντότητα του πρωτοκόλλου, που δρομολογεί τις SIP αιτήσεις στους UAS και τις SIP απαντήσεις στους UAC. Μία αίτηση μπορεί να περάσει από πολλούς Proxies μέχρι να φτάσει στον τελικό της στόχο. Κάθε Proxy παίρνει με τη σειρά του αποφάσεις για τη σωστή δρομολόγηση της αίτησης, μεταβάλλοντας την αίτηση προτού την προωθήσει. Οι απαντήσεις δρομολογούνται μέσω των ίδιων Proxies προς την αντίθετη κατεύθυνση αυτή τη φορά.

Είναι χρήσιμο να θεωρήσει κανείς τους Proxies ως δρομολογητές του SIP-επιπέδου, που προωθούν τις SIP αιτήσεις και απαντήσεις. Η λογική που χρησιμοποιούν είναι όμως πιο πολύπλοκη από μια απλή προώθηση μηνυμάτων, που βασίζεται σε έναν πίνακα δρομολόγησης. Οι προδιαγραφές του SIP επιτρέπουν στους Proxies να προβαίνουν σε

ενέργειες, όπως είναι η εγκυρότητα των αιτήσεων , η ταυτοποίηση των χρηστών και ο εντοπισμός και χειρισμός ατέρμονων βρόχων. Η πολλαπλή χρησιμότητά τους επιτρέπει στο διαχειριστή του συστήματος να τους χρησιμοποιεί για διάφορους λόγους και σε διάφορες τοποθεσίες του δικτύου.

Ένας Proxy είναι σχεδιασμένος με τέτοιο τρόπο, ώστε η λειτουργία του να είναι όσο πιο διαφανή γίνεται στους UAs. Στους Proxy Servers επιτρέπεται να αλλάξουν οποιοδήποτε μήνυμα μόνο με κάποιους καθορισμένους και περιορισμένους τρόπους. Για παράδειγμα, δε μπορούν σε καμία περίπτωση να αλλάξουν το SDP σώμα ενός INVITE. Πλην κάποιων εξαιρέσεων , οι Proxies μπορούν να εκκινήσουν αιτήσεις με δική τους πρωτοβουλία. Δε μπορούν όμως να τερματίσουν έναν ενεργό διάλογο στέλνοντας ένα BYE Request οι ίδιοι. Ένας Proxy Server διαχωρίζεται σε δύο κατηγορίες οι οποίες παρουσιάζονται σε επόμενες ενότητες:

- STATEFUL PROXY SERVER
- STATELESS PROXY SERVER

2.2.2.1 STATEFUL PROXY SERVERS

Ένας Δυναμικός Πληρεξούσιος Εξυπηρετητής (STATEFUL PROXY SERVER) δε χειρίζεται ανεξάρτητα μηνύματα, αλλά SIP συναλλαγές. Γνωρίζει την κατάσταση των συναλλαγών και των μηνυμάτων που έχουν σταλεί και μπορεί κατά συνέπεια να επεξεργάζεται καλύτερα τα εισερχόμενα μηνύματα. Χρησιμοποιεί την αποθηκευμένη πληροφορία επηρεάζοντας την επεξεργασία μελλοντικών εισερχόμενων αιτήσεων που σχετίζονται με μια αίτηση. Έχει την ικανότητα να διακλαδώσει μια αίτηση δρομολογώντας την σε πολλαπλούς προορισμούς. Κάθε αίτηση η οποία προωθείται σε περισσότερους από έναν προορισμούς πρέπει να χειρίζεται δυναμικά. Μπορεί να αλλάξει την λειτουργία του σε στατική οποιαδήποτε στιγμή κατά την διάρκεια της επεξεργασίας μιας αίτησης αρκεί να μην εκτελέσει κάποια δυναμική λειτουργία(δημιουργία απάντησης με κωδικό 100-100 response).Ένας STATEFUL PROXY δεν μπορεί να δημιουργήσει μια αίτηση-Cancel(CANCEL request). Παρουσιάζει όμως και κάποια μειονεκτήματα, όπως είναι η κατανάλωση μνήμης και η πολυπλοκότητα της υλοποίησής του.

2.2.2.2 STATELESS PROXY SERVERS

Ένας Στατικός Πληρεξούσιος Εξυπηρετητής (STATELESS PROXY SERVER) ενεργεί ως μια απλή μηχανή προώθησης SIP μηνυμάτων. Προωθεί κάθε εισερχόμενη αίτηση σε μια οντότητα της οποίας την διεύθυνση υπολογίζει παίρνοντας κάποιες αποφάσεις δρομολόγησης οι οποίες βασίζονται στις παραμέτρους της ίδιας της αίτησης(του μυνήματος-αίτηση), και απλά προωθεί κάθε απάντηση που λαμβάνει. Μετά από την προώθηση των αιτήσεων, απαντήσεων ένας stateless proxy παύει να διατηρεί πληροφορίες για τη συναλλαγή, στην οποία αναφέρεται το μήνυμα.

Σημειώνεται επίσης ότι ως Outbound Proxy αναφέρεται ο Proxy που λαμβάνει αιτήσεις από ένα πελάτη ανεξαρτήτως του προορισμού των μηνυμάτων που στέλνει ο πελάτης (Request-URI). Οι πελάτες μπορεί να επιλέξουν να στέλνουν κάθε εξερχόμενο μήνυμα μέσω ενός Outbound Proxy.

2.2.3 REDIRECT SERVERS

Ο Εξυπηρετητής Ανακατεύθυνσης (Redirect Server) είναι το απλούστερο είδος SIP Server. Ένας Εξυπηρετητής Ανακατεύθυνσης λαμβάνει SIP αιτήσεις και δίνει μια απάντηση της κλάσης 3xx, κατευθύνοντας τον πελάτη να επικοινωνήσει με μια εναλλακτική ομάδα SIP διευθύνσεων. Οι εναλλακτικές διευθύνσεις υπάρχουν στο πεδίο Contact της επικεφαλίδας της απάντησης.

Η ανακατεύθυνση δίνει τη δυνατότητα στους Servers να επιστρέφουν πληροφορίες δρομολόγησης στους πελάτες, βοηθώντας στον εντοπισμό του στόχου, ενώ οι ίδιοι δε λαμβάνουν πλέον μέρος στη SIP συναλλαγή. Κατά συνέπεια, οι Redirect Servers δεν κρατάνε πληροφορίες για την κατάσταση των διαλόγων, αλλά μόνο για την πορεία ανεξάρτητων συναλλαγών που χειρίζονται οι ίδιοι. Η ανακατεύθυνση είναι μια απλή και γρήγορη διαδικασία, που επιτρέπει στους Redirect Servers να επιτυγχάνουν υψηλή απόδοση.

2.2.4 REGISTRAR SERVERS

Ο Registrar ή Εξυπηρετητής Εγγραφών είναι ένα είδος εξυπηρετητή , που δέχεται αιτήσεις εγγραφής (REGISTER Requests) και τοποθετεί όσες πληροφορίες λαμβάνει στο Location Service για το domain που χειρίζεται. Κάθε Registrar χειρίζεται τα REGISTER Requests για ένα συγκεκριμένο domain ή σύνολο domains. Χρησιμοποιεί το Location Service (μια αφηρημένη βάση με τοποθεσίες) για την αποθήκευση και την ανάκτηση πληροφοριών για τη θέση των χρηστών. Ο Location Service μπορεί να τρέχει σε ένα απομακρυσμένο μηχάνημα και η επικοινωνία μαζί του να γίνεται με τη χρήση ενός κατάλληλου πρωτοκόλλου (π.χ. με το LDAP). Η επιλογή εξαρτάται από την εκάστοτε υλοποίηση. Μερικές υλοποιήσεις μπορούν να τοποθετήσουν το Location Service και το Registrar στο ίδιο μηχάνημα.

Ο Registrar μπορεί να ζητήσει ταυτοποίηση των εισερχόμενων αιτήσεων χρησιμοποιώντας την 401 (Unauthenticated) απάντηση, ενώ απορρίπτει την αίτηση, αν λάβει κάποιο μήνυμα με μέθοδο άλλη από τη REGISTER δίνοντας μια 501 (Not Implemented) απάντηση. Συνήθως ένας Registrar συνυπάρχει στο ίδιο σύστημα με έναν Proxy και έναν Redirect Server και η διαφοροποίηση υφίσταται μόνο λογικά και όχι φυσικά.

2.3 SIP ΜΗΝΥΜΑΤΑ

Η εγκατάσταση, η συντήρηση και ο τερματισμός μιας συνόδου επιτυγχάνεται μέσω της ανταλλαγής συγκεκριμένων μηνυμάτων. Λόγω τις ομοιότητας του SIP με τα HTTP και SMTP τα SIP μηνύματα έχουν κληρονομήσει χαρακτηριστικά από αυτά τα πρωτόκολλα.

Υπάρχουν δύο κατηγορίες SIP μηνυμάτων: οι SIP αιτήσεις (SIP Requests) που στέλνονται από τον πελάτη στον εξυπηρετητή και οι SIP απαντήσεις (SIP Responses) που στέλνονται από τον εξυπηρετητή στον πελάτη. Στις επόμενες ενότητες περιγράφονται όλα τα είδη των δύο παραπάνω κατηγοριών SIP μηνυμάτων για να γίνει κατανοητή η χρήση τους στις διάφορες συνόδους.

2.3.1 ΓΕΝΙΚΗ ΔΟΜΗ

Το SIP είναι ένα text-based πρωτόκολλο που χρησιμοποιεί το Uniform Transformation Format-8 (UTF-8 - http://www.utf-8.com RFC 2279) σύνολο χαρακτήρων. Τόσο οι αιτήσεις όσο οι απαντήσεις χρησιμοποιούν τη βασική δομή, που περιγράφεται στο RFC 2822 αν και στη σύνταξη υπάρχουν διαφορές παραδείγματος χάρη το SIP περιέχει κεφαλίδες οι οποίες δεν είναι έγκυρες σύμφωνα με το RFC 2822.

Όλα τα μηνύματα αποτελούνται από μία γραμμή έναρξης (start-line), που μπορεί να είναι είτε μία γραμμή αίτησης (Request-line), είτε μια γραμμή κατάστασης (Status-line) αναλόγως αν πρόκειται για αίτηση ή απάντηση, ένα ή περισσότερα πεδία επικεφαλίδας (header fields), που χρησιμοποιούνται για να μεταφέρουν τα χαρακτηριστικά του μηνύματος, μία κενή γραμμή που καθορίζει το τέλος των πεδίων της επικεφαλίδας και ένα προαιρετικό σώμα (body), όπως φαίνεται και στο παρακάτω παράδειγμα μηνύματος:

```
INVITE sip:tpu@hp.com SIP/2.0                          | START-LINE
Via: SIP/2.0/UDP local.hp.com                          |
From: OC <sip:OpenCall.SIP@hp.com>                     |
To: TPU <sip:tpu@hp.com>                               |
Subject: Confcall                                      |
Call-ID: 132059753@local.hp.com                        | HEADER-FIELDS
Content-Type: application/sdp                          |
CSeq: 1 INVITE                                         |
Contact: <sip:telecom@16.188.155.140>                  |
Content-Length: 187                                    |

v=0                                                            |
o=user1 51633745 1348648134 IN IP4 16.188.155.140      |
s=Interactive Conference                               |
c=IN IP4 224.2.4.4/127                                         | BODY
t=0 0                                                          |
m=audio 3456 RTP/AVP 0 22                              |
a=rtpmap:22 application/g723.1                         |
```

Το σώμα του μηνύματος χρησιμοποιείται για την περιγραφή της συνόδου που θα εγκατασταθεί (για παράδειγμα σε μια σύνοδο πολυμέσων περιγράφει τα είδη

κωδικοποίησης του ήχου και του video κτλ.) ή εναλλακτικά μπορεί να χρησιμοποιηθεί για να περιέχει δυαδικά δεδομένα ή δεδομένα κειμένου οποιουδήποτε τύπου, που σχετίζονται με κάποιο τρόπο με τη σύνοδο. Στο SIP υπάρχει μια σαφής διάκριση μεταξύ της πληροφορίας σηματοδοσίας που υπάρχει στη γραμμή έναρξης και την επικεφαλίδα, και των πληροφοριών περιγραφής του session που είναι ανεξάρτητες από το πρωτόκολλο.

Η γραμμή έναρξης, κάθε γραμμή επικεφαλίδας του μηνύματος και η κενή γραμμή τερματίζονται με μια αλλαγή γραμμής και ένα χαρακτήρα επαναφοράς (CRLF).Η κενή γραμμή πρέπει να είναι παρούσα στο μήνυμα, ακόμα και αν δεν υπάρχει σώμα. Μεγάλο μέρος της σύνταξης των SIP μηνυμάτων και των πεδίων της επικεφαλίδας είναι ίδιο με αυτό του πρωτοκόλλου HTTP/1.1.

2.3.2 ΕΠΙΚΕΦΑΛΙΔΕΣ (HEADER FIELDS)

Οι SIP επικεφαλίδες είναι παρόμοιες με αυτές του HTTP όσον αφορά την σύνταξη και την σημασιολογία. Οι επικεφαλίδες κατασκευάζονται σύμφωνα με την παρακάτω γραμματική σε BNF μορφή όπως ορίζεται και στην παράγραφο 25 του RFC 3261:

```
headers        =  "?" header *( "&" header )
header         =  hname "=" hvalue
hname          =  1*( hnv-unreserved / unreserved / escaped )
hvalue         =  *( hnv-unreserved / unreserved / escaped )
hnv-unreserved =  "[" / "]" / "/" / "?" / ":" / "+" / "$"
```

Το SIP επιτρέπει την ύπαρξη πολλών επικεφαλίδων με το ίδιο όνομα και διαφορετικές τιμές καθώς και την κατασκευή σύνθετων επικεφαλίδων (multiple header) που ορίζονται σύμφωνα με τον παρακάτω κανόνα:

```
header  =  "header-name" HCOLON header-value *(COMMA header-value)
```

Οπότε οι 3 παρακάτω επικεφαλίδες είναι ίδιες και μπορούμε να τις συναντήσουμε σε κάποιο μήνυμα SIP:

- Route: <sip:alice@atlanta.com>
  Subject: Lunch
  Route: <sip:bob@biloxi.com>
  Route: <sip:carol@chicago.com>

- Route: <sip:alice@atlanta.com>, <sip:bob@biloxi.com>
  Route: <sip:carol@chicago.com>
  Subject: Lunch

- Subject: Lunch
  Route: <sip:alice@atlanta.com>, <sip:bob@biloxi.com>,
  <sip:carol@chicago.com>

Η σειρά με την οποία εμφανίζονται οι επικεφαλίδες σε ένα μήνυμα SIP δεν ορίζεται όμως συνιστάται οι επικεφαλίδες οι οποίες χρειάζονται για την επεξεργασία στους proxy servers (π.χ. Via, Route, Record-Route, Proxy-Require, Max-Forwards, Proxy-Authorization) να εμφανίζονται στην αρχή για την αποφυγή άσκοπης σάρωσης ολόκληρου του μηνύματος.

Μια επικεφαλίδα μπορεί επίσης να συνοδεύετε από μια ή περισσότερες παραμέτρους οι οποίες προστίθενται στο τέλος της τιμής μετά από ένα ερωτηματικό «;» όπως περιγράφεται από τον παρακάτω κανόνα:

field-name: field-value *(;parameter-name=parameter-value)

Αν και μπορούν να προστεθούν πολλές παράμετροι σε μια επικεφαλίδα δεν μπορούν να εμφανίζονται παραπάνω από μια παράμετροι με το ίδιο όνομα. Τα ονόματα και οι τιμές των επικεφαλίδων ,τα ονόματα και οι τιμές των παραμέτρων καθώς και τα αλφαριθμητικά που ορίζονται μέσα σε εισαγωγικά είναι Case-Insensitive εκτός αν ορίζεται αλλιώς για κάποια επικεφαλίδα. Παραδείγματος χάρη οι παρακάτω επικεφαλίδες δεν διαφέρουν:

- Contact: <sip:alice@atlanta.com>;expires=3600
CONTACT: <sip:alice@atlanta.com>;ExPiReS=3600

- Content-Disposition: session;handling=optional
content-disposition: Session;HANDLING=OPTIONAL

Το SIP διαθέτει επίσης έναν μηχανισμό (Compact Form)για να μειώνει το μέγεθος ενός μηνύματος όταν είναι απαραίτητο(π.χ. όταν χρησιμοποιείται το πρωτόκολλο UDP για να μην υπερβαίνει το μέγιστο επιτρεπόμενο μέγεθος πακέτου-Maximum Transmission Unit<MTU>) ορίζοντας κάποιες συντομογραφίες στα ονόματα κάποιων επικεφαλίδων(π.χ. Contact = m και Content-Encoding = e).Μερικές επικεφαλίδες που χρησιμοποιούνται σήμερα φαίνονται στον παρακάτω πίνακα(Πίνακας 2.1):

Πίνακας 2.1: **Επικεφαλίδες** SIP **Μηνυμάτων**.

ΕΠΙΚΕΦΑΛΙΔΑ	ΠΕΡΙΓΡΑΦΗ
Accept	καθορίζει τους αποδεκτούς τύπους μέσων (media types) στο σώμα του μηνύματος (π.χ. κείμενο, ήχος κτλ.).
Accept-Encoding	καθορίζει τις αποδεκτές κωδικοποιήσεις των μέσων.
Accept-Language	καθορίζει τις προτιμώμενες γλώσσες.
Alert-Info	καθορίζει έναν εναλλακτικό ήχο για τις εισερχόμενες κλήσεις του χρήστη.
Allow	δηλώνει το σύνολο μεθόδων που υποστηρίζει ο UA.
Authentication-Info	Δίνει τη δυνατότητα αμοιβαίας πιστοποίησης με χρήση του HTTP Digest (http://www.ietf.org/rfc/rfc2617.txt).

Authorization	περιέχει πληροφορίες για την ταυτοποίηση του UA.
Call-id	παράμετρος που καθορίζει μονοσήμαντα μια αίτηση για έναρξη συνεδρίας από ένα UA.
Call-Info	περιέχει επιπρόσθετες πληροφορίες για τον αποστολέα του μηνύματος.
Contact	περιέχει ένα URI, του οποίου το νόημα εξαρτάται από τον τύπο του μηνύματος.
Content-Disposition	περιγράφει τον τρόπο με τον οποίο πρέπει να ερμηνευτεί το σώμα του μηνύματος.
Content-Encoding	αναφέρει τους μηχανισμούς αποκωδικοποίησης/συμπίεσης που έχουν εφαρμοστεί στο σώμα του μηνύματος.
Content-Language	αναφέρει τη γλώσσα στην οποία είναι γραμμένο το σώμα του μηνύματος.
Content-Length	αναφέρει το μέγεθος του σώματος του μηνύματος. Η τιμή 0 δηλώνει πως δεν υπάρχει σώμα στο μήνυμα.
Content-Type	καθορίζει τον τύπο του μέσου στο σώμα του μηνύματος.
CSeq	περιέχει ένα δεκαδικό αριθμό, που αυξάνεται για κάθε αίτηση και τη μέθοδο της αίτησης που χρησιμοποιείται στο μήνυμα.
Date	η ημερομηνία και ώρα.
Error-Info	περιέχει δείκτη σε επιπρόσθετες πληροφορίες για το σφάλμα που συνέβη.
Expires	το χρονικό διάστημα (σε δευτερόλεπτα) μετά το οποίο το μήνυμα δεν έχει πια ισχύ.
From	πληροφορίες για την οντότητα που ξεκίνησε την αρχική αίτηση, στην οποία αναφέρεται το μήνυμα.
In-Reply-To	περιέχει το call-ID, στο οποίο αναφέρεται το συγκεκριμένο μήνυμα ή το οποίο επιστρέφει το μήνυμα (υπάρχει μόνο στα SIP Requests).
Max-Forwards	καθορίζει το μέγιστο αριθμό των κόμβων που μπορούν να προωθήσουν το συγκεκριμένο Request.

Min-Expires	καθορίζει τον ελάχιστο χρόνο, για τον οποίο στοιχεία της information base ενός SIP Server δεν μπορούν να αλλαχτούν.
MIME-Version	Μηνύματα που έχουν δημιουργηθεί με βάση το Multipurpose Internet Mail Extensions πρωτόκολλο (MIME - http://www.mhonarc.org/~ehood/MIME) περιέχουν αυτό το header field που καθορίζει την έκδοσή του.
Organization	περιέχει το όνομα του οργανισμού στον οποίο ανήκει η SIP οντότητα που εξέδωσε το μήνυμα.
Priority	περιέχει την προτεραιότητα που πρέπει να δοθεί στο συγκεκριμένο μήνυμα σε περιπτώσεις συμφόρησης. Ορίζονται τέσσερις κλάσεις προτεραιότητας: όχι επείγον (non-urgent), κανονικό (normal), άμεσης προτεραιότητας (urgent) και επείγον (emergency).
Proxy-Authenticate	περιέχει δεδομένα απαραίτητα για την ταυτοποίηση του UA που έστειλε το μήνυμα από τον Πληρεξούσιο Εξυπηρετητή (Proxy Server - ένα είδος SIP server), που πρόκειται να το επεξεργαστεί.
Proxy-Authorization	το περιεχόμενο του είναι στις περισσότερες περιπτώσεις ίδιο με του Proxy-Authenticate header field.
Proxy-Require	περιέχει τις προδιαγραφές που πρέπει να πληρεί ο Proxy Server για να επεξεργαστεί το μήνυμα.
Record-Route	προστίθεται από κάποιον Proxy Server για να δηλώσει ότι όλα τα μηνύματα που αφορούν σε έναν συγκεκριμένο διάλογο πρέπει να δρομολογηθούν μέσω αυτού.
Reply-To	περιέχει ένα URI, όχι κατά ανάγκη ίδιο με αυτό του From header field. Μπορεί να χρησιμοποιηθεί για να αποστέλλονται λίστες με χαμένες ή ανεπιτυχείς κλήσεις.
Require	χρησιμοποιείται από τους UAC για να ενημερώσουν τους UAS για τις επιλογές που πρέπει να υποστηρίζουν, ώστε να επεξεργαστούν επιτυχώς το μήνυμα.
Retry-After	πρόκειται για ένα ακέραιο που ισούται

	με το χρόνο (σε δευτερόλεπτα) που πρέπει ο καλών να περιμένει πριν ξανακαλέσει σε περίπτωση που λάβει κάποιο συγκεκριμένο μήνυμα αποτυχίας.
Route	περιέχει μια λίστα από Proxy Servers, μέσα από τους οποίους πρέπει να δρομολογηθεί το συγκεκριμένο μήνυμα.
Server	περιέχει πληροφορίες σχετικά με το software που χρησιμοποιεί ο UAS για να επεξεργαστεί το request.
Subject	το θέμα του μηνύματος.
Supported	όλες τις επεκτάσεις που υποστηρίζει ο UA.
Timestamp	καθορίζει την ακριβή ώρα που στάλθηκε το Request μήνυμα.
To	περιέχει το URI και ίσως και άλλες πληροφορίες για τον τελικό παραλήπτη του μηνύματος.
Unsupported	περιέχει λίστα με χαρακτηριστικά που δεν υποστηρίζονται από τον UAS.
User-Agent	περιέχει πληροφορίες για τον UA που δημιούργησε το request.
Via	περιέχει τη διαδρομή που ακολούθησε το μήνυμα μέχρι τώρα (ως ακολουθία IP διευθύνσεων), αν πρόκειται για Request μήνυμα, ή καθορίζει τη διαδρομή που θα πρέπει να ακολουθηθεί, αν πρόκειται για μήνυμα απάντησης.
Warning	περιέχει επιπρόσθετες πληροφορίες για την κατάσταση της απάντησης.
WWW-Authenticate	χρησιμοποιείται για την πιστοποίηση του UAC από τον UAS.

2.3.3 ΟΙ ΕΝΝΟΙΕΣ DIALOG ΚΑΙ TRANSACTION

Σύμφωνα με το RFC 3261 ένας SIP Διάλογος (SIP Dialog) είναι μια peer-to-peer σχέση ανάμεσα σε δύο User Agents η οποία υφίσταται για κάποια χρονική περίοδο. Ένας SIP Dialog καθορίζεται από συγκεκριμένες μεθόδους οι οποίες σηματοδοτούν την αρχή και το τέλος του ,π.χ. από μια Invite αίτηση έως μια επιτυχής απάντηση 2xx, και περιλαμβάνει όλες τις διαδικασίες και τα μηνύματα ανάμεσα από τα δύο αυτά όρια.

Ένα SIP Dialog χαρακτηρίζεται μοναδικά από 3 τιμές οι οποίες είναι απαραίτητες σε κάθε μήνυμα το οποίο μπορεί να είναι μέρος ενός Dialog:

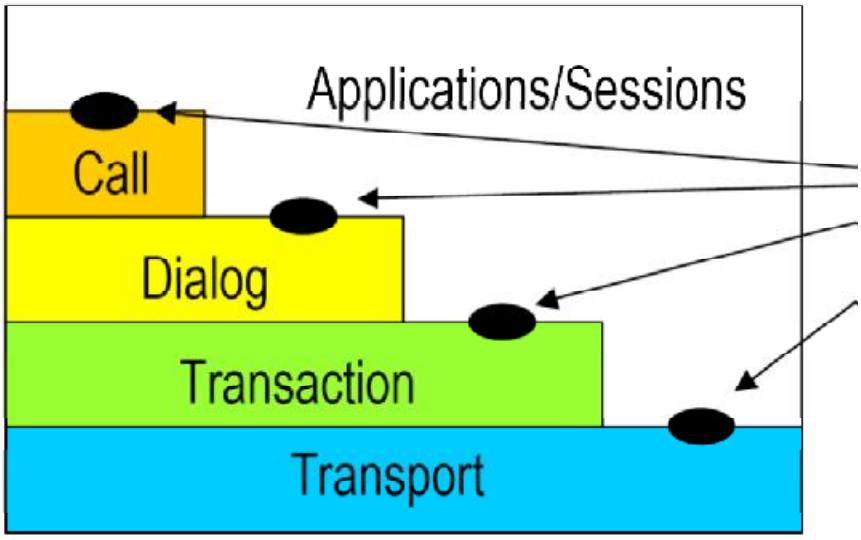
- Call-Identifier(Call-id Header)
- Local Tag
- Remote Tag

Ένα SIP Transaction είναι η διαδικασία μεταξύ ενός Client και ενός Server και οριοθετείται από την αποστολή της αρχικής αίτησης από τον Client στον Server μέχρι την αποστολή της τελικής απάντησης (όχι της κλάσης 1xx) από τον Server στον Client.

SIP Transactions υπάρχουν μέσα σε SIP Dialog αλλά ορίζονται και ανεξάρτητα σε μερικές περιπτώσεις όπως στην διαδικασία Register ενός User Agent με ένα Registrar όπου η όλη διαδικασία περιγράφεται από ένα Transaction.

Οι Transaction και Dialog αποτελούν μέρος του ιεραρχικού μοντέλου του SIP ορίζοντας 2 επίπεδα(**Σχήμα** 2.1):

- Transaction Layer
- Dialog Layer



Σχήμα 2.1: **Ιεραρχικό μοντέλο του** Sip

2.3.4 SIP REQUESTS

Οι SIP αιτήσεις (SIP REQUESTS) αποτελούν την μια από τις 2 κατηγορίες μηνυμάτων. Αναγνωρίζονται από τη γραμμή αίτησης (Request-line) που έχουν ως γραμμή έναρξης. Η Request-line περιέχει το όνομα της μεθόδου της αίτησης, ένα ζητούμενο URI (Request-URI) και την έκδοση του πρωτοκόλλου, χωρισμένα μεταξύ τους με ένα κενό χαρακτήρα. Ένα παράδειγμα μιας γραμμής αίτησης είναι το παρακάτω όπου είναι μια αίτηση εγγραφής

(REGISTER) προς τον Registrar Server με SIP URI: ceid.upatras.gr με έκδοση SIP πρωτοκόλλου 2:

REGISTER sip:192.168.2.2:5060 SIP/2.0

Το SIP πρωτόκολλο διαθέτει έξι μεθόδους οι οποίες αναλύονται παρακάτω:

- REGISTER
- INVITE
- ACK
- CANCEL
- BYE
- και OPTIONS.

2.3.4.1 REGISTER

Η μέθοδος REGISTER χρησιμοποιείται από τον User Agent για να δηλώσει στο SIP δίκτυο την τρέχουσα διεύθυνση επικοινωνίας του (Contact URI, IP address) και το URI (ουσιαστικά το AOR του χρήστη), για το οποίο όλες οι αιτήσεις θα πρέπει να κατευθύνονται στη συγκεκριμένη διεύθυνση. Ένα REGISTER Request μήνυμα μπορεί να περιέχει σώμα, αν και η λειτουργία του δεν καθορίζεται στις προδιαγραφές του πρωτοκόλλου.

Ένα REGISTER Request μήνυμα έχει ως προορισμό τον αντίστοιχο Registrar Server ο οποίος είναι υπεύθυνος για το συγκεκριμένο Domain που ανήκει ο χρήστης. Το μήνυμα είτε στέλνεται κατευθείαν στον Registrar εφόσον είναι γνωστή η διεύθυνσή του, είτε σε έναν Proxy Server ο οποίος αναλαμβάνει να ανακαλύψει τον Registrar που χρειαζόμαστε .Ένα παράδειγμα ενός μηνύματος REGISTER φαίνεται παρακάτω:

ΜΗΝΥΜΑ REGISTER

REGISTER sip:registrar.ceid.teipat.gr SIP/2.0
Via: SIP/2.0/UDP pc32.ceid.teipat.gr:5060;branch=z9hG4bKnashds7
Max-Forwards: 70
To: GENTIAN <sip:gentian@ceid.teipat.gr>
From: GENTIAN <sip: gentian@ceid.teipat.gr>;tag=456248
Call-ID: 843817637684230@998sdasdh09
CSeq: 1826 REGISTER
Contact: <sip:gentian@192.168.2.4>
Expires: 7200
Content-Length: 0

Οι βασικές επικεφαλίδες που πρέπει να εμφανίζονται σε κάθε REGISTER request είναι:

- Γραμμή Έναρξης ,που περιέχει το όνομα της μεθόδου(REGISTER),το domain του Registrar που είναι υπεύθυνος για το χειρισμό του συγκεκριμένου Request(Request-URI) και την έκδοση του SIP.
- Via, που περιέχει την μέθοδο, την διεύθυνση του Registrar και την έκδοση του πρωτοκόλλου.
- From, περιέχει το AOR του UA που στέλνει την αίτηση για να δηλώσει την παρουσία του στη συγκεκριμένη θέση στο δίκτυο.
- To, περιέχει την ίδια πληροφορία με την επικεφαλίδα From..
- Call-id, παραμένει η ίδια για κάθε REGISTER Request που στέλνεται από τον ίδιο UA.
- C-seq, αυξάνεται για κάθε νέα αίτηση REGISTER που στέλνεται για το ίδιο AOR.
- Max-Forwards, μειώνεται για κάθε hop που θα κάνει το μήνυμα και φανερώνει την λήξη του μηνύματος όταν πάρει την τιμή 0.
- Contact, που δείχνει την διεύθυνση του UA(ή την τιμή * βλέπε παρακάτω).

Μια επικεφαλίδα η οποία μπορεί να μην εμφανίζεται πάντα αλλά αλλάζει την σημασία του μηνύματος είναι η Expires η οποία αν υπάρχει φανερώνει την λήξη του μηνύματος καθώς και την λήξη της εγγραφής στο Location Service και παίρνει τιμές μεγαλύτερες ή ίσον του μηδέν .Εκτός της επικεφαλίδας υπάρχει και μια παράμετρος Expires στην επικεφαλίδα Contact η οποία φανερώνει την λήξη της εγγραφής στο Location Service και παρακάμπτει την επικεφαλίδα αν υπάρχει.

Συνεπώς ανάλογα των επικεφαλίδων Contact και Expires(μπορεί να είναι και παράμετρος της επικεφαλίδας Contact) ένας Registrar μπορεί να προβεί στις παρακάτω ενέργειες:

- Contact=URI,Expires>0. Σ' αυτήν την περίπτωση ο εξυπηρετητής εγγραφών θα πρέπει να δημιουργήσει μια εγγραφή στο Location Service όπου θα συνδέει το AOR του χρήστη με όλες τις διευθύνσεις που περιέχονται στην επικεφαλίδα Contact και θα λήγει σε όσα δευτερόλεπτα δηλώνει η επικεφαλίδα Expires.
- Contact=URI,Expires=0.Διαγράφει την εγγραφή που έχει Expires =0.
- Contact=*,Expires= 0. Διαγράφει όλες τις εγγραφές που σχετίζονται με αυτό το AOR.
- Contact=URI. Όταν δεν υπάρχει επικεφαλίδα Expires τότε ορίζει ο Registrar ορίζει τον χρόνο λήξης της εγγραφής.

Μια ακόμη παράμετρος η οποία πρέπει να αναγνωρίζεται από έναν Registrar είναι η "q" η οποία δηλώνει την προτεραιότητα του κάθε Contact σε σχέση με τα υπόλοιπα. Τέλος ο κάθε UA είναι υπεύθυνος να ανανεώνει την εγγραφή του με ένα νέο Register Request πριν αυτό λήξει. Στην περίπτωση που όλα πάνε καλά ο Registrar απαντά με 200 OK προσθέτοντας μια παράμετρο Expires δίπλα από κάθε Contact έγινε επιτυχημένα η εγγραφή και δηλώνει τον χρόνο λήξης της εγγραφής.

2.3.4.2 INVITE

Ένας UA χρησιμοποιεί τη μέθοδο INVITE, όταν θέλει να ξεκινήσει τη διαδικασία για επικοινωνία με έναν άλλο UA. Το μήνυμα μίας αίτησης INVITE μπορεί να περιέχει σώμα(Body Part) με τις πληροφορίες μέσου (media information) του αποστολέα. Μπορεί επίσης να περιέχει άλλες πληροφορίες για τη συνεδρία, όπως την ποιότητα υπηρεσίας (Quality of Service - QoS) ή πληροφορίες ασφάλειας. Ένα επιτυχές INVITE Request εγκαθιστά ένα διάλογο μεταξύ των δύο UAs, που διαρκεί, ώσπου μια αίτηση BYE να σταλεί από έναν από τους δύο για να τερματίσει τη συνεδρία. Ο χρήστης που στέλνει το αρχικό INVITE κατασκευάζει ένα μοναδικό Call-ID, που χρησιμοποιείται από όλα τα μέλη καθ' όλη τη διάρκεια της συνεδρίας.

Τα πεδία επικεφαλίδας που πρέπει τουλάχιστον να εμφανίζονται σε ένα INVITE Request είναι: Call-ID, CSeq, From, To, Via, Contact, Content-Length και Max-Forwards, όπως φαίνεται και στο παρακάτω δείγμα:

INVITE sip:george@yahoo.comSIP/2.0
Via: SIP/2.0/UDP pc32.ceid.teipat.gr;rport=5090;branch=z9hG4bK13344
Max-Forwards: 70
To: <sip: george@yahoo.com>
From: <sip:gentian@ceid.teipat.gr>;tag=z9hG4bK26797963
Call-ID: 862664514648@192.168.2.2
CSeq: 1 INVITE
Contact: <sip:gentian@pc32.ceid.upatras.gr>
Content-Length: 0

2.3.4.3 ACK

ACK είναι η μέθοδος που χρησιμοποιείται για την επιβεβαίωση των τελικών απαντήσεων που στέλνονται για ένα INVITE Request. Το πεδίο CSeq της επικεφαλίδας δεν αυξάνεται για ένα ACK Request, αλλά αλλάζει μόνο η μέθοδος που εμφανίζεται. Με αυτόν τον τρόπο μπορεί ένας UA να ταιριάξει τον αριθμό του CSeq του ACK με τον αντίστοιχο αριθμό του INVITE στο οποίο αναφέρεται. Για 2xx απαντήσεις (μία ομάδα απαντήσεων, που θα αναφερθεί και παρακάτω), το ACK στέλνεται άκρο-άκρο (end-to-end), ενώ για όλες τις άλλες τελικές απαντήσεις στέλνεται ανά κόμβο (hop-by-hop).

Τα πεδία επικεφαλίδας που πρέπει τουλάχιστον να εμφανίζονται σε ένα ACK Request είναι: Call-ID, CSeq, From, To, Via, Content-Length και Max-Forwards.

ACK sip:maria@192.168.2.2:5090 SIP/2.0
Via: SIP/2.0/UDP 192.168.2.2:5080;rport;branch=z9hG4bK78308
Max-Forwards: 70
To: <sip:192.168.2.2:5090>;tag=350bac123730b714
From: <sip:gentian@teipat.gr>;tag=z9hG4bK26797963

Call-ID: 862664514648@192.168.2.2
CSeq: 1 ACK
Content-Length: 0

2.3.4.4 CANCEL

Η μέθοδος CANCEL χρησιμοποιείται για τον τερματισμό αναζητήσεων ή κλήσεων που εκκρεμούν. Μπορεί να κατασκευαστεί είτε από έναν UA, είτε από έναν Proxy Server, εφόσον έχει ληφθεί μια 1xx(PROVISIONAL) απάντηση. Ο UA χρησιμοποιεί το CANCEL για να ακυρώσει μια κλήση την οποία άρχισε και δεν έχει εγκατασταθεί μέχρι εκείνη την στιγμή.

Τα πεδία επικεφαλίδας που πρέπει να εμφανίζονται σε ένα CANCEL Request είναι: Call-ID, CSeq, From, To, Via, Content-Length και Max-Forwards όπως φαίνεται στο παρακάτω παράδειγμα:

CANCEL sip:maria@192.168.2.2:5090 SIP/2.0
Via: SIP/2.0/UDP 192.168.2.2:5080;branch=z9hG4bKfae8cb69f547b8cb
Max-Forwards: 70
To: < sip:maria@192.168.2.2:5090 >
From: < sip:gentian@192.168.2.2:5080 >;tag=102
Call-ID: 862664514648@192.168.2.2
CSeq: 5000 CANCEL
Content-Length: 0

2.3.4.5 BYE

Η μέθοδος BYE χρησιμοποιείται για τον τερματισμό μιας συνεδρίας. Μπορεί να σταλεί μόνο από τους UAs που συμμετέχουν στο διάλογο και σε καμία περίπτωση από κάποιον SIP Server ή από κάποιον τρίτο. Η αίτηση στέλνεται απευθείας στον άλλο UA που συμμετέχει στη σύνοδο και η απάντηση επίσης.

Τα πεδία επικεφαλίδας που πρέπει να εμφανίζονται είναι: Call-ID, CSeq, From, To, Via και Max-Forwards, Content-Length όπως φαίνεται στο παρακάτω παράδειγμα:

BYE sip:gentian@192.168.2.2:5090 SIP/2.0
Via: SIP/2.0/UDP 192.168.2.2:5080;rport;branch=z9hG4bK51006
Max-Forwards: 70
To: <sip:192.168.2.2:5090>;tag=99f9a35fa7a9120a
From: <sip:maria@teipat.gr>;tag=z9hG4bK06835992
Call-ID: 726349630045@192.168.2.2
CSeq: 2 BYE
Content-Length: 0

2.3.4.6 OPTIONS

Η τελευταία μέθοδος χρησιμοποιείται για να ανακαλύψει τις δυνατότητες και τη διαθεσιμότητα ενός UA ή SIP Server. Η απάντηση στην αίτηση απαριθμεί όλες τις δυνατότητες και τα χαρακτηριστικά της οντότητας.

Τα πεδία επικεφαλίδας που πρέπει να εμφανίζονται είναι: Call-ID, CSeq, From, To, Via και Max-Forwards, Content-Length όπως φαίνεται στο παρακάτω παράδειγμα:

OPTIONS sip: gentian@ceid.teipat.gr SIP/2.0
Via: SIP/2.0/UDP ceid.teipat.gr;rport;branch=z9hG4bK51006
Max-Forwards: 70
To: <ceid.teipat.gr>
From: <sip: gentian@ceid.teipat.gr >
Call-ID: 726349630045@192.168.2.2
CSeq: 1 OPTIONS
Accept: text/xml
Content-Length: 0

2.3.5 SIP RESPONSES

Μία SIP απάντηση (SIP Response) είναι ένα μήνυμα, που δημιουργείται από έναν UAS ή από έναν SIP Server για να απαντήσει σε μια αίτηση που ξεκίνησε από έναν UAC. Η γραμμή έναρξης σε κάθε μήνυμα απάντησης είναι μια γραμμή κατάστασης (Status-line). Η Status-line αποτελείται από την έκδοση του πρωτοκόλλου, τον τριψήφιο αριθμητικό κωδικό της κατάστασης (status code) και την επεξηγηματική φράση που αντιστοιχεί σε αυτόν (reason).

Τα SIP μηνύματα χωρίζονται σε έξι κλάσεις οι οποίες διαχωρίζονται από το πιο σημαντικό ψηφίο του 3-ψήφιου κωδικού κατάστασης και κάθε κλάση αντιπροσωπεύει μια διαφορετική κατηγορία απαντήσεων. Οι έξι αυτές κλάσεις διαχωρίζονται περαιτέρω σε 2 κατηγορίες απαντήσεων τις προσωρινές απαντήσεις(PROVISIONAL RESPONSES) που αποτελούνται από την κλάση 1xx ενώ οι κλάσεις 2xx έως 6xx ανήκουν στην κατηγορία των τελικών απαντήσεων (FINAL RESPONSES). Στις ενότητες που ακολουθούν αναλύονται οι έξι κλάσεις των SIP απαντήσεων.

2.3.5.1 PROVISIONAL (1xx)

Αποτελούν προσωρινές απαντήσεις οι οποίες φανερώνουν την κατάσταση και πρόοδο μιας κλήσης αλλά δεν μπορούν να την τερματίσουν. Παραδείγματα αποτελούν οι απαντήσεις:

- 100 TRYING που στέλνεται από έναν Proxy με το που λαμβάνει μια αίτηση INVITE .
- 180 RINGING που στέλνεται από έναν UAS όταν αναμένει απόφαση από τον καλούντα.
- 181 CALL IS BEING FORWARDED που ενημερώνει ότι η κλήση προωθείται.
- 182 CALL QUEUED που δηλώνει ότι ο UA δεν είναι διαθέσιμος. Ωστόσο η κλήση μπήκε σε ουρά προτεραιότητας και δεν απορρίφθηκε.
- 183 SESSION PROGGRESS Η απάντηση αυτή περιέχει πληροφορίες για την κατάσταση της κλήσης που δεν περιγράφονται από άλλα status codes.

Πριν από κάποια τελική απάντηση μπορούμε να λάβουμε από μια έως πολλές Provisional Responses. Ένας UAC μπορεί να δημιουργήσει "Early Dialogs" όταν χρειαστεί να στείλει ένα Provisional Response στον συνομιλητή του μέσα σε ένα Dialog πριν ολοκληρωθεί το Transaction της αρχικής INVITE μεθόδου.

Τα πεδία επικεφαλίδας που πρέπει να εμφανίζονται είναι: Call-ID, CSeq, From, To, Via και Max-Forwards, Content-Length όπως φαίνεται στο παρακάτω παράδειγμα:

SIP/2.0 100 Trying
Via: SIP/2.0/UDP 192.168.2.2:5080;branch=z9hG4bK51688;rport=5080
To: <sip:192.168.2.2:5090>;tag=68e841da1ac059b4
From: <sip:maria@teipat.gr>;tag=z9hG4bK24289724
Call-ID: 713082948951@192.168.2.2
CSeq: 1 INVITE
Content-Length: 0

2.3.5.2 SUCCESS (2xx)

Ανήκει στις τελικές απαντήσεις και φανερώνει την επιτυχία της αίτησης στην οποία αναφέρεται. Στην μοντέλο 200(OK)/ACK πρέπει για κάθε 200 απάντηση που φθάνει να στέλνεται πίσω ένα μήνυμα ACK.

Αυτή η κλάση διαθέτει τις αιτήσεις:

- 200(OK) όπου δηλώνει ότι η αίτηση έγινε δεκτή ή ότι επεξεργάστηκε επιτυχώς.

2.3.5.3 REDIRECTION (3xx)

Χρησιμοποιείται όταν χρειάζεται να γίνει ανακατεύθυνση της κλήσης ή κάποιου αιτήματος. Συνήθως στέλνεται από έναν Redirect Server είτε για να επιστρέψει μια διεύθυνση ενός χρήστη είτε για να δηλώσει ένα λάθος στον εντοπισμό ενός συγκεκριμένου χρήστη(και από Proxy Server) και διαθέτει τις παρακάτω απαντήσεις:

- 300 MULTIPLE CHOICES: Αυτή η απάντηση περιλαμβάνει πολλά Contact πεδία επικεφαλίδας, που δείχνουν ότι ο Location Service επέστρεψε πολλές πιθανές διευθύνσεις για το AOR που υπάρχει στο Request-URI.
- 301 MOVED PERMANENTLY: Δηλώνει ότι ο χρήστης έχει μετακινηθεί μόνιμα σε κάποια άλλη διεύθυνση την οποία και περιέχει και μπορεί ο UA να την αποθηκεύσει για μελλοντική χρήση.
- 302 MOVED TEMPORARILY: Περιέχει μια προσωρινή διεύθυνση για τον εντοπισμό του χρήστη, αλλά αντίθετα με την 301 όχι μόνιμη.
- 305 USE PROXY: Δείχνει ότι ο χρήστης πρέπει να ξαναστείλει την αίτηση, χρησιμοποιώντας τον Proxy Server που ορίζεται στην επικεφαλίδα Contact.
- 380 ALTERNATIVE SERVICE: Η απάντηση επιστρέφει ένα URI, που δείχνει το είδος της υπηρεσίας, που θα ήθελε ο καλούμενος.

2.3.5.4 CLIENT ERROR (4xx)

Η 4xx κλάση απαντήσεων χρησιμοποιείται από έναν UAS ή από ένα SIP Server για να δείξει ότι η εξυπηρέτηση της αίτησης δεν μπορεί να ολοκληρωθεί. Το είδος του σφάλματος ή η ύπαρξη κάποιων header fields μπορεί να υποδείξει στον UAC πώς θα μπορούσε να ανακατασκευαστεί η αίτηση. Ο UAC δε θα πρέπει να δοκιμάσει να ξαναστείλει το ίδιο Request χωρίς να το αλλάξει σύμφωνα με τις πληροφορίες της απάντησης. Εδώ ανήκουν οι απαντήσεις:

- 400 Bad Request
- 401 Unauthorized
- 402 Payment Required
- 403 Forbidden
- 404 Not Found
- 405 Method Not Allowed
- 406 Not Acceptable
- 407 Proxy Authentication Required
- 408 Request Timeout
- 410 Gone
- 412 Conditional Request Failed
- 413 Request Entity Too Large
- 414 Request-URI Too Long
- 415 Unsupported Media Type
- 416 Unsupported URI Scheme
- 417 Unknown Resource-Priority

- 420 Bad Extension
- 421 Extension Required
- 422 Session Interval Too Small
- 423 Interval Too Brief
- 428 Use Identity Header
- 429 Provide Referrer Identity
- 433 Anonymity Disallowed
- 436 Bad Identity-Info
- 437 Unsupported Certificate
- 438 Invalid Identity Header
- 440 Max-Breadth Exceeded
- 470 Consent Needed
- 480 Temporarily Unavailable
- 481 Call/Transaction Does Not Exist
- 482 Loop Detected
- 483 Too Many Hops
- 484 Address Incomplete
- 485 Ambiguous
- 486 Busy Here
- 487 Request Terminated
- 488 Not Acceptable Here
- 489 Bad Event
- 491 Request Pending
- 493 Undecipherable
- 494 Security Agreement Required

2.3.5.5 SERVER ERROR (5xx)

Η κλάση 5xx περιέχει απαντήσεις που δηλώνουν ότι η αίτηση δε μπορεί να εξυπηρετηθεί, λόγω ενός σφάλματος που υπάρχει στο εξυπηρετητή (Server). Οι πιθανές απαντήσεις είναι:

- 500 Server Internal Error
- 501 Not Implemented
- 502 Bad Gateway
- 503 Service Unavailable
- 504 Server Time-out
- 505 Version Not Supported
- 513 Message Too Large
- 580 Precondition Failure [RFC3312]

2.3.5.6 GLOBAL FAILURE (6xx)

Αυτή η κλάση περιέχει απαντήσεις που δείχνουν πως ο Server γνωρίζει ότι η εξυπηρέτηση της αίτησης θα αποτύχει, σε όποια θέση και αν δοκιμαστεί. Εδώ ανήκουν οι απαντήσεις:

- 600 Busy Everywhere
- 603 Decline
- 604 Does Not Exist Anywhere
- 606 Not Acceptable

2.4 SIP ΔΙΕΥΘΥΝΣΕΙΣ

Το SIP υποστηρίζει 3 τύπους διευθύνσεων τα Sip URI(Uniform Resource Indicators), Sips URI(Secure Sip URI) και tel(Telephone URI). Μια SIP διεύθυνση μπορούμε να την συναντήσουμε σε πολλά σημεία σε ένα SIP μήνυμα όπως στις επικεφαλίδες To , From ,Contact και στο Request-URI. Η μορφή ενός URI είναι παρόμοια με μια διεύθυνση e-mail.

2.4.1 SIP URI

Η γενική μορφή ενός Sip URI είναι η παρακάτω:

sip:user:password@host:port;uri-parameters?headers

- sip: δηλώνει τον τύπο της διεύθυνσης.
- user: το όνομα του χρήστη είναι ανάλογο με το ψευδώνυμο σε έναν λογαριασμό mail
- password: είναι προαιρετικό καθώς το Sip παρέχει άλλους τρόπους ασφάλειας καθώς η μεταφορά του κωδικού μέσω ενός απλού κειμένου όπως είναι το Sip URI έχει αποδειχτεί ριψοκίνδυνη.
- Host: Περιέχει είτε το domain όνομα του λογαριασμού είτε ολόκληρη την IPv4,IPv6 διεύθυνση.
- Port: Προσδιορίζει την πόρτα όπου πρέπει να σταλεί η αίτηση.
- Uri-parameters: Περιέχει παραμέτρους που επηρεάζουν την αίτηση και είναι σχετικές με το URI όπως transport, maddr, ttl,user, method και lr.
- Headers: Εισάγοντας το σύμβολο ? μπορούμε έπειτα να παραθέσουμε τις επικεφαλίδες του μηνύματος.

2.4.2 SIPS URI

Είναι παρόμοιο με το SIP URI μόνο που προϋποθέτει μια ασφαλή επικοινωνία μέσω TLS ανάμεσα μεταξύ του UAC και του domain που ανήκει το URI. Όλη η μετέπειτα επικοινωνία με τον χρήστη γίνεται με ασφαλή τρόπο ο οποίος καθορίζεται από το domain. Μια ακόμη διαφορά είναι στο σχήμα όπου από sip: γίνεται sips: .Τα SIP URI και SIPS URI περιγράφονται στο RFC 2396.

Παραδείγματα:

sip:alice@atlanta.com
sip:alice:secretword@atlanta.com;transport=tcp
sips:alice@atlanta.com?subject=project%20x&priority=urgent
sip:+1-212-555-1212:1234@gateway.com;user=phone
sips:1212@gateway.com
sip:alice@192.0.2.4
sip:atlanta.com;method=REGISTER?to=alice%40atlanta.com
sip:alice;day=tuesday@atlanta.com

2.4.3 TELEPHONE URI

Το telephone URI χρησιμοποιείται για να ενώσει 2 διαφορετικά δίκτυα: το PSTN με το VOIP.Ένας SIP Client μπορεί να χρησιμοποιήσει ένα telephone URI για να καλέσει έναν τηλεφωνικό αριθμό.

Παραδείγματα:

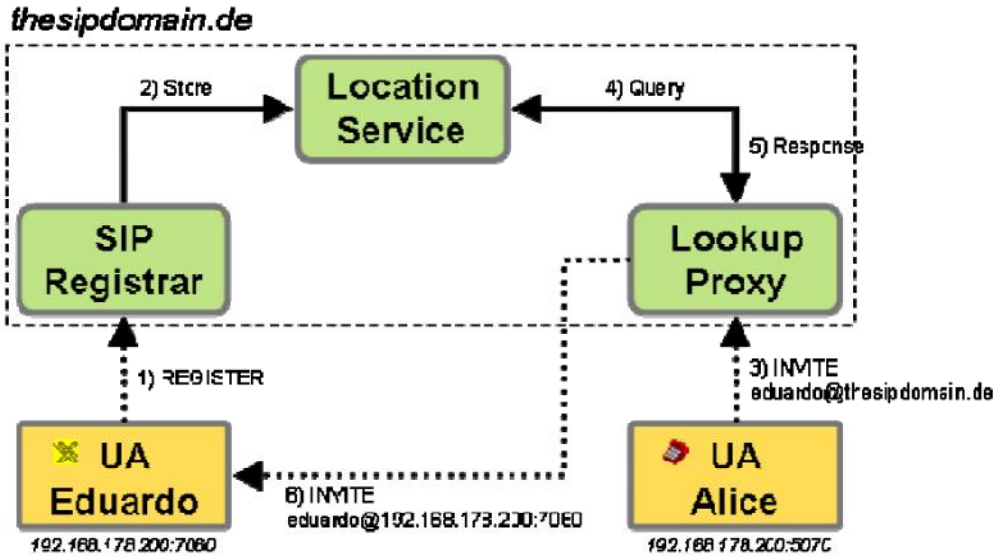
tel:+1-201-555-0123
tel:7042;phone-context=example.com
tel:863-1234;phone-context=+1-914-555

2.5 ΠΑΡΑΔΕΙΓΜΑΤΑ

2.5.1 REGISTRATION

Τα παρακάτω παραδείγματα δείχνει τα βήματα που πρέπει να ακολουθήσει ένας UA για να εγγραφεί με έναν Registrar Server. Το παράδειγμα 1 δείχνει μια επιτυχής εγγραφή με τον Registrar και στην συνέχεια κλήση μέσω ενός Lookup Proxy Server και το παράδειγμα 2 την δομή των πιθανών SIP αιτήσεων απαντήσεων.

ΠΑΡΑΔΕΙΓΜΑ 1



Σχήμα 2.2: **Εγγραφή μέσω** Registrar **και κλήση**.

ΠΑΡΑΔΕΙΓΜΑ 2

Ο UAC στέλνει την αίτηση:
REGISTER sip:192.168.2.2: 5060 SIP/2.0
Via: SIP/2.0/UDP 192.168.2.2:5080; rport;branch=z9hG4bK64657
Max-Forwards: 70
To: <sip:gentian@teipat.gr>
From: <sip: gentian@teipat.gr >;tag=z9hG4bK26754977
Call-ID: 241494949617@192.168.2.2
CSeq: 1 REGISTER
Contact: <sip:gentian@192.168.2.2:5080>
Expires: 1000
User-Agent: gentian_client_info
Content-Length: 0

-----End-of-message-----

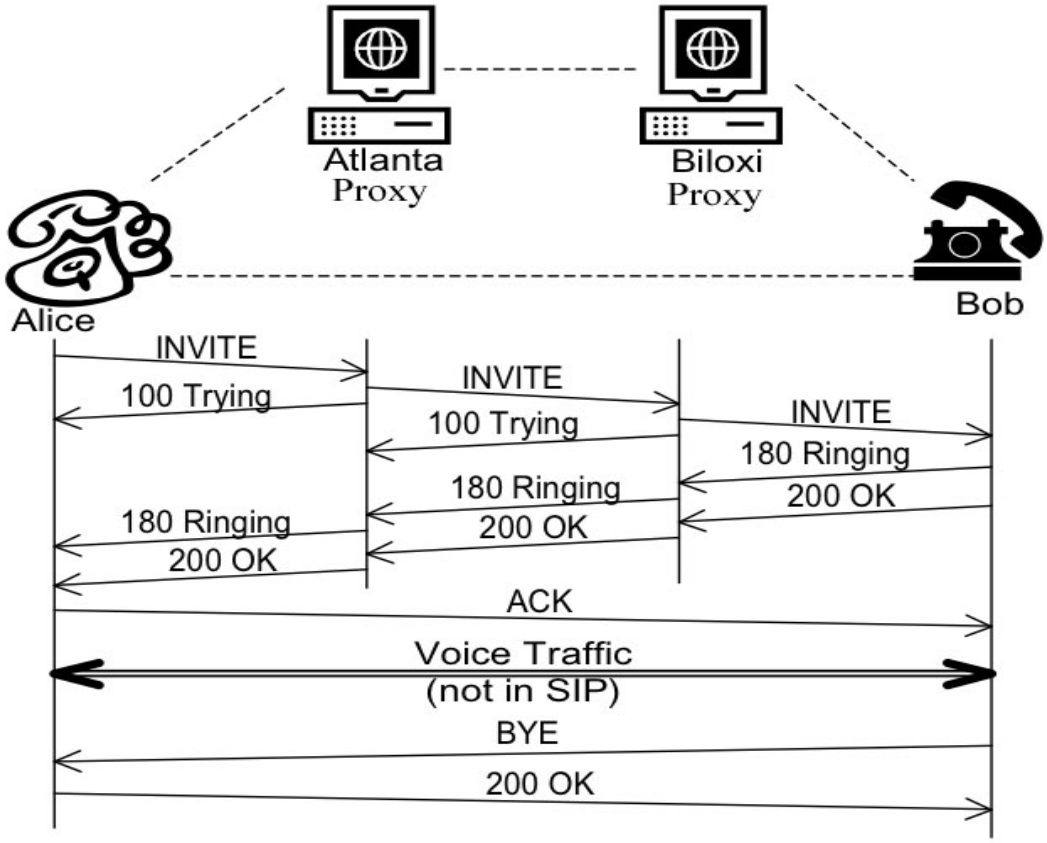
Και ο REGISTRAR απαντάει:
SIP/2.0 200 OK
Via: SIP/2.0/UDP 192.168.2.2:5080; branch=z9hG4bK64657;rport=5080
To: <sip: gentian@teipat.gr >

From: <sip: gentian@teipat.gr >;tag=z9hG4bK26754977
Call-ID: 241494949617@192.168.2.2
CSeq: 1 REGISTER
Server: gentian_server_info
Contact: <sip:gentian@192.168.2.2:5080>;expires=1000
Content-Length: 0

-----End-of-message-----

2.5.2 INVITE WITH PROXY SERVERS

Σε αυτό το παράδειγμα (Σχήμα 2.3)βλέπουμε τα SIP μηνύματα που πρέπει να ανταλλάξουν δύο χρήστες (Alice,Bob) για να εγκαταστήσουν μια σύνοδο μεταξύ τους με την βοήθεια δύο Proxy Servers (Atlanta Proxy,Biloxi Proxy). Παρότι στο παράδειγμα φαίνεται ότι οι Proxies δεν συμμετέχουν στην επικοινωνία, μόλις η Alice διαπιστώσει(μέσω του ACK μηνύματος) ότι ο Bob απάντησε και η επικοινωνία γίνεται άμεσα μεταξύ τους, αυτό δεν είναι η μόνη περίπτωση. Στην γενική περίπτωση ένας Proxy μπορεί να επιλέξει να παραμείνει στην μέση της επικοινωνίας για να παρέχει κάποιες υπηρεσίες διάσκεψης εν μέσω κλήσης(middle-call conferencing services).Σ` αυτήν την περίπτωση ο Proxy παρεμβάλλεται στην ροή των SIP μηνυμάτων αλλά η ροή των δεδομένων φωνής και βίντεο γίνεται πάλι απευθείας μεταξύ των δύο χρηστών.

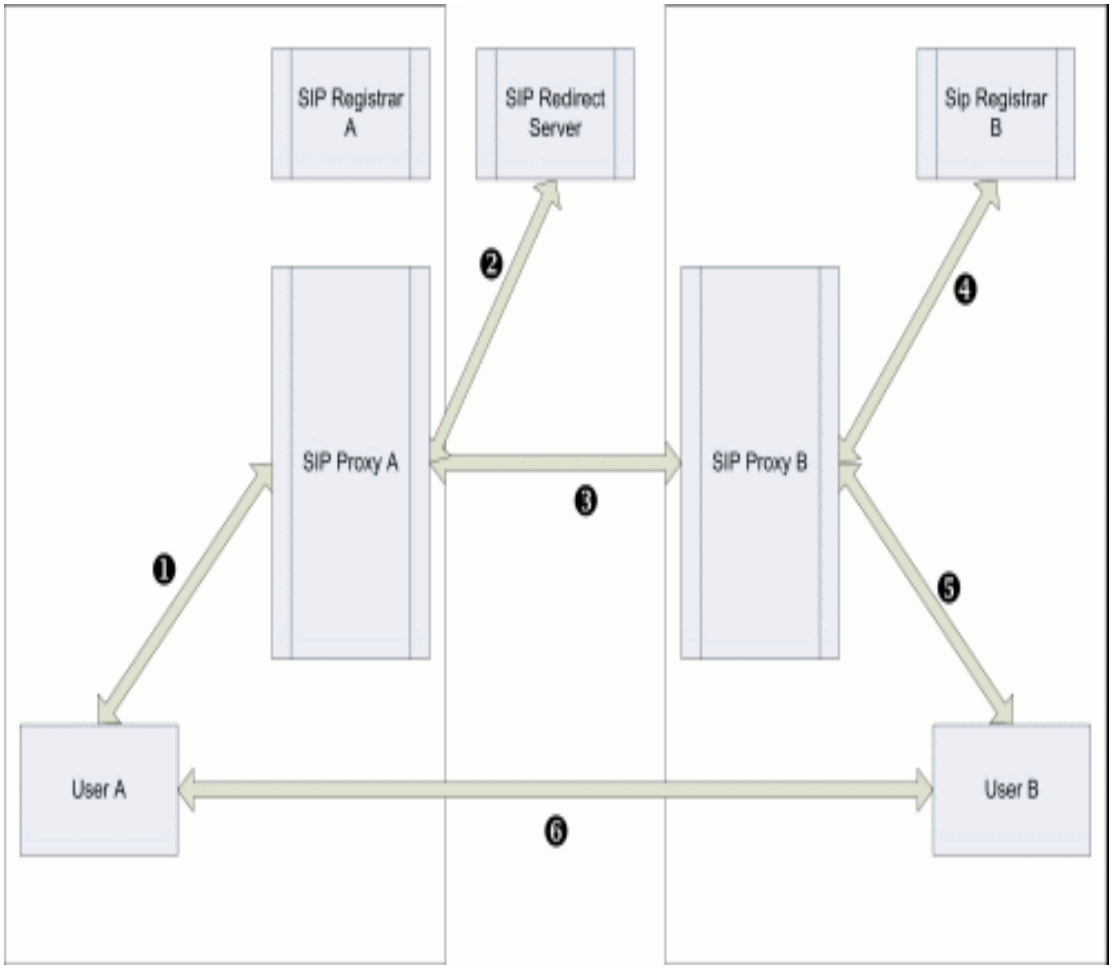


Σχήμα 2.3: **Κλήση με δύο** Proxy Servers.

2.5.3 INVITE WITH PROXY AND REDIRECT SERVERS

Το παρακάτω παράδειγμα(Σχήμα 2.4) δείχνει την εγκατάσταση μιας συνόδου μεταξύ δύο χρηστών μόνο που παρεμβάλλεται και ένας Redirect Server για την εύρεση του δεύτερου Proxy.

1. Ο χρήστης Α καλεί τον χρήστη Β. Ο χρήστης Β ανήκει σε διαφορετικό Domain από τον χρήστη Α και η αίτηση παραλαμβάνεται από τον Proxy Α.(Βέλος 1)
2. Ο Proxy Α αντιλαμβάνεται ότι ο χρήστης Β βρίσκεται σε διαφορετικό Domain από το δικό του και στέλνει μια αίτηση στον Redirect Server για να μάθει που θα βρει τον χρήστη Β. (Βέλος 2)
3. Ο Redirect Server απαντάει στον Proxy Α στέλνοντας την διεύθυνση του Proxy Β. (Βέλος 2)
4. Ο Proxy Α στέλνει την αρχική αίτηση του χρήστη Α στον Proxy Β. (Βέλος 3)
5. Ο Proxy Β αναζητά την θέση του χρήστη Β στέλνοντας μια αίτηση στον Registrar Β. (Βέλος 4)
6. Ο Registrar Β απαντά με την διεύθυνση του χρήστη Β. (Βέλος 4)
7. Ο Proxy Β στέλνει την αίτηση σον χρήστη Β. (Βέλος 5)
8. Ο χρήστης Β δέχεται την κλήση.
9. Η συσκευή του χρήστη Β στέλνει την απάντηση(ότι δέχθηκε την κλήση) πίσω στον Proxy Β. (Βέλος 5)
10. Ο Proxy Β στέλνει την απάντηση με την σειρά του στον Proxy Α. (Βέλος 3)
11. Ο Proxy Α στέλνει την απάντηση στην συσκευή του χρήστη Α. (Βέλος 1)
12. Το κανάλι επικοινωνίας έχει εγκατασταθεί και οι δύο χρήστες στέλνουν δεδομένα απευθείας. (Βέλος 6)



Σχήμα 2.4: **Κλήση με** Proxy **και** Redirect Servers.

ΚΕΦΑΛΑΙΟ 3

RTP/RTCP

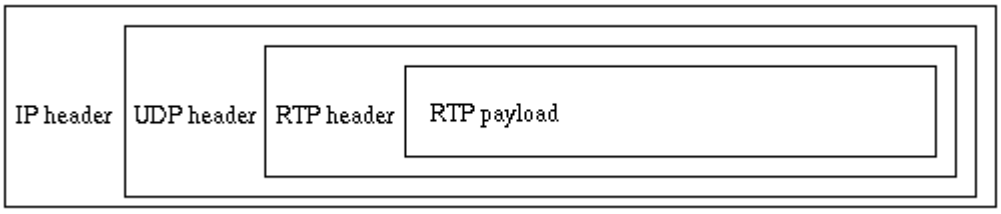
ΚΕΦΑΛΑΙΟ 3: RTP/RTCP

Το Real-Time Transport Protocol (πρωτόκολλο μεταφοράς πραγματικού χρόνου) ορίζει ένα τυπικό μορφότυπο πακέτου για την παράδοση ήχου και εικόνας με χαρακτηριστικά πραγματικού χρόνου μέσω του διαδικτύου. Δημιουργήθηκε από τον όμιλο Audio Video Transport Working του IETF και δημοσιεύτηκε για πρώτη φορά το 1996 ως RFC 1889. Έπειτα ανανεώθηκε το 2003 και ορίζεται στο RFC 3550.

Το RTP χρησιμοποιείται συνήθως σε:

- Simple multicast Audio Conference: συνδιάσκεψη μόνο φωνής σε
- Audio and Video Conference: Συνδιάσκεψη με ήχο και εικόνα.
- Mixers and Translators: Οι μεταφραστές απλώς μεταφράζουν μια μορφή ωφέλιμου φορτίου σε μια άλλη, ενώ οι μίκτες συνδυάζουν πολλαπλά ρεύματα σε ένα απλό ρεύμα διατηρώντας την αρχική τους μορφή. Μίκτες και μεταφραστές χρησιμοποιούνται συνήθως για την μετάδοση σε δίκτυα χαμηλών και υψηλών ταχυτήτων ταυτόχρονα(low-speed networks and high-speed networks).
- Layered Encodings: Ελέγχει τον ρυθμό μετάδοσης από την πλευρά του δέκτη και απελευθερώνει την πηγή συνδυάζοντας ένα σύστημα layered-encoding και ένα layered-transmission (rate-adaption στον δέκτη).

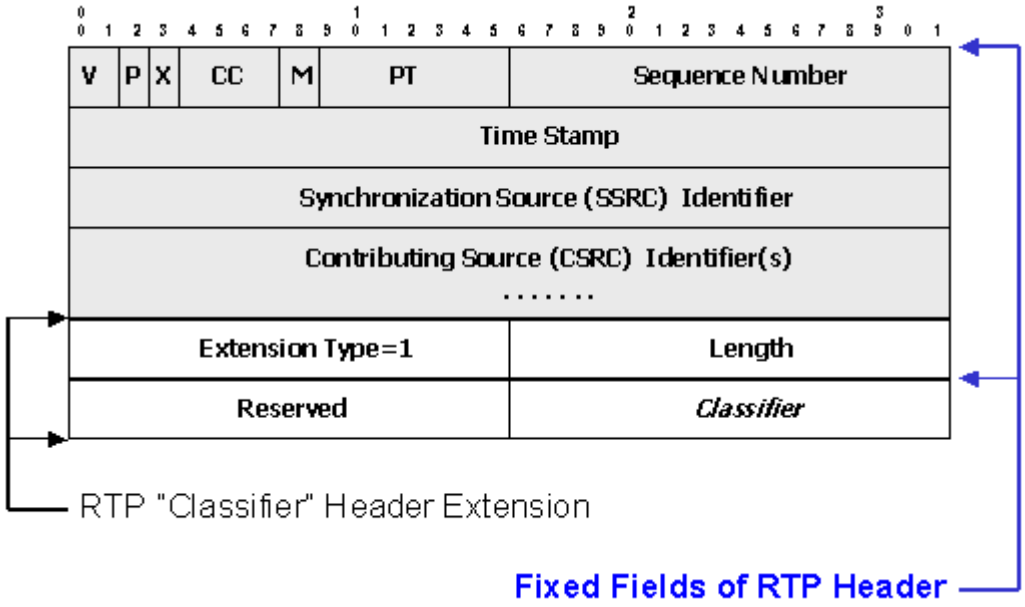
Το RTP είναι στενά συνδεδεμένο με το RTCP πρωτόκολλο(Real Time Control Protocol). Ενώ το RTP χρησιμοποιείται για την μεταφορά των δεδομένων το RTCP παρέχει πληροφορίες ποιότητας της συνόδου (QoS) καθώς και των μελών της συνόδου. Συνήθως τα RTP- RTCP δεσμεύουν τις θύρες μεταξύ 16384-32767. Το RTP δεσμεύει μια θύρα ζυγού αριθμού ενώ το RTCP την αμέσως επόμενη μονή.
Παρόλο που το κύριο πεδίο εφαρμογής για το οποίο είναι αρχικά σχεδιασμένο το RTP είναι η ικανοποίηση των αναγκών πολυμελούς τηλεδιάσκεψης πολυμέσων, εντούτοις δεν περιορίζεται στη συγκεκριμένη εφαρμογή. Εφαρμογές αποθήκευσης continuous δεδομένων, interactive distributed simulation, active badge, εφαρμογές ελέγχου και μετρήσεων και άλλες εφαρμογές πραγματικού χρόνου μπορούν να χρησιμοποιήσουν το RTP ικανοποιητικά.
Το RTP παρέχει υπηρεσίες μεταφοράς από άκρο σε άκρο, αλλά δεν παρέχει όλη την λειτουργικότητα που παρέχεται από ένα τυπικό πρωτόκολλο μεταφοράς. Για παράδειγμα, το RTP συνήθως λειτουργεί στην κορυφή του UDP για να χρησιμοποιεί τις υπηρεσίες πολύπλεξης και αθροίσματος ελέγχου του πρωτοκόλλου αυτού(Σχήμα 3.1). Μπορεί όμως να λειτουργεί και πάνω από IPX δίκτυα ή πάνω ATM δίκτυα. Το RTP δεν γνωρίζει την έννοια της σύνδεσης και γι αυτό μπορεί να λειτουργεί είτε πάνω από προσανατολισμένα κατά σύνδεση δίκτυα είτε πάνω από χωρίς σύνδεση πρωτόκολλα χαμηλού επιπέδου.



Σχήμα 3.1 RTP **δεδομένα σε ένα** IP **πακέτο**.

3.1 ΔΟΜΗ RTP ΠΑΚΕΤΟΥ

Κάθε πακέτο RTP περιέχει στην αρχή του μία επικεφαλίδα που αποτελείται από 12 υποχρεωτικά πεδία (octets) και 1 προαιρετικό(CSRC identifiers).Στο σχήμα 3.2 φαίνεται η δομή της RTP επικεφαλίδας και η περιγραφή όλων των πεδίων.



Σχήμα 3.2 **Επικεφαλίδα ενός** RTP **πακέτου**.

- Version-“V” (2 bits): Δείχνει την τρέχουσα έκδοση του πρωτοκόλλου. Αυτή τη στιγμή η τρέχουσα έκδοση είναι η 2.
- Padding-“P”(1 bit): Χρησιμοποιείται στην περίπτωση που η εφαρμογή απαιτεί η μεταδιδόμενη πληροφορία να είναι πολλαπλάσια ενός ακέραιου αριθμού bits. Η πληροφορία ενδέχεται να μην είναι πολλαπλάσιο αυτού του αριθμού, οπότε ο αριθμός 1 στο bit μας πληροφορεί πως υπάρχουν άχρηστα bits στο τέλος του πακέτου. Το τελευταίο byte του πακέτου σημειώνει τον ακριβή αριθμό από bits που είναι άχρηστα.
- Extension-“X” (1 bit): Όταν είναι ίσο με 1 τότε το σταθερό τμήμα της επικεφαλίδας ακολουθείται από την επέκταση της επικεφαλίδας, η οποία χρησιμοποιείται για πειραματικούς σκοπούς.
- Contributing Source (CSRC) Identifier-“CC”(4 bits): ο κωδικός του CSRC που ακολουθεί τη σταθερή επικεφαλίδα.
- Marker-“M”(1 bit): εξαρτάται από το είδος της πληροφορίας που μεταδίδει το πακέτο. Συνήθως σημαδεύει ένα όριο στη συνεχή ροή της πληροφορίας, π.χ. το τέλος ενός video frame ή την αρχή ομιλίας ενός συγκεκριμένου ομιλητή.
- Payload Type-“PT” (7 bits): πληροφορεί για τον τύπο της πληροφορίας που περιέχει το RTP πακέτο και που ακολουθεί την επικεφαλίδα.

- Sequence Number (16 bits): Κάθε πηγή πληροφορίας ξεκινά γεμίζοντας αυτό το πεδίο με έναν τυχαίο αριθμό, τον οποίο αυξάνει κατά ένα για κάθε πακέτο δεδομένων που αποστέλλεται. Χρησιμεύει στον παραλήπτη, ώστε να μπορεί σε συνδυασμό με τη χρονοσήμανση (timestamp) του πακέτου, να τοποθετεί τα λαμβανόμενα πακέτα στη σωστή σειρά, πριν τα επεξεργαστεί ή τα αναπαραγάγει. Για την τοποθέτηση των πακέτων στη σωστή σειρά είναι απαραίτητα και τα δύο πεδία, καθώς μερικά πακέτα (πχ αυτά που απαρτίζουν το ίδιο video frame) ανήκουν στην ίδια χρονική στιγμή.
- Timestamp (32 bits): Αντιστοιχεί στη χρονική στιγμή της δημιουργίας του πρώτου byte στην πληροφορία του τρέχοντος πακέτου. Το πεδίο παίρνει τιμή από το τοπικό ρολόι του αποστολέα.
- Synchronization Source Identifier-"SSRS"(32 bits): Ένας τυχαία παραγόμενος αριθμός ο οποίος μοναδικά περιγράφει μία πηγή πληροφορίας μέσα σε μία σύνοδο.
- Contributing Source Identifier-"CSRS"(0-15 items,32 bits each): Σηματοδοτεί την πηγή που συμμετέχει στο τμήμα της πληροφορίας που ακολουθεί στο πακέτο. Το πεδίο αυτό χρησιμοποιείται όταν τα δεδομένα τα οποία λαμβάνονται προέρχονται από ένα μείκτη και προσδιορίζει ποίοι από τους συμμετέχοντες έχουν συνεισφέρει για την πληροφορία την οποία περιέχει το πακέτο που λαμβάνουμε.

Το πεδίο "Payload Type" περιέχει έναν κωδικό που αντιστοιχεί στο είδος της πληροφορίας που μεταφέρει το πακέτο μαζί με τη μέθοδο κωδικοποίησης ή και συμπίεσης που ενδεχομένως έχει γίνει στην πληροφορία. Σε σταθερές συνθήκες δικτύου μία πηγή χρησιμοποιεί μόνο μία κωδικοποίηση πληροφορίας, η οποία όμως μπορεί να αλλάξει με την αλλαγή συνθηκών και συμπεριφοράς του δικτύου. Στους πίνακες 3.1 και 3.2 παρουσιάζονται οι τύποι πληροφορίας για ήχο και βίντεο αντίστοιχα του RFC 3551 "RTP Profile for Audio and Video Conferences with Minimal Control":

PT	name	encoding	media type	clock rate (Hz)	channels
0	PCMU	A		8,000	1
1	reserved	A			
2	reserved	A			
3	GSM	A		8,000	1
4	G723	A		8,000	1
5	DVI4	A		8,000	1
6	DVI4	A		16,000	1
7	LPC	A		8,000	1
8	PCMA	A		8,000	1
9	G722	A		8,000	1
10	L16	A		44,100	2
11	L16	A		44,100	1
12	QCELP	A		8,000	1
13	CN	A		8,000	1
14	MPA	A		90,000	(see text)
15	G728	A		8,000	1
16	DVI4	A		11,025	1
17	DVI4	A		22,050	1
18	G729	A		8,000	1
19	reserved	A			

20	unassigned	A		
21	unassigned	A		
22	unassigned	A		
23	unassigned	A		
dyn	G726-40	A	8,000	1
dyn	G726-32	A	8,000	1
dyn	G726-24	A	8,000	1
dyn	G726-16	A	8,000	1
dyn	G729D	A	8,000	1
dyn	G729E	A	8,000	1
dyn	GSM-EFR	A	8,000	1
dyn	L8	A	var.	var.
dyn	RED	A		(see text)
dyn	VDVI	A	var.	1

Πίνακας 3.1: Payload Types **για ήχο**.

	PT name	encoding	media type	clock rate (Hz)
24	unassigned	V		
25	CelB	V	90,000	
26	JPEG	V	90,000	
27	unassigned	V		
28	nv	V	90,000	
29	unassigned	V		
30	unassigned	V		
31	H261	V	90,000	
32	MPV	V	90,000	
33	MP2T	AV	90,000	
34	H263	V	90,000	
35-71	unassigned	?		
72-76	reserved	N/A	N/A	
77-95	unassigned	?		
96-127	dynamic	?		
dyn	H263-1998	V	90,000	

Πίνακας 3.2: Payload Types **για βίντεο**.

3.2 ΔΟΜΗ RTCP ΠΑΚΕΤΟΥ

Το πρωτόκολλο RTP χρησιμοποιείται μόνο για τη μεταφορά των δεδομένων πραγματικού χρόνου. Αυτό το ίδιο δεν αποτελεί μέσο για την παρακολούθηση και τον έλεγχο της εφαρμογής πραγματικού χρόνου. Το τελευταίο είναι στόχος του πρωτοκόλλου RTCP. Το RTCP είναι ένα πρωτόκολλο ελέγχου που σχεδιάστηκε για να συνεργάζεται με το RTP. Υπάρχουν οι παρακάτω 5 τύποι πακέτων για το πρωτόκολλο RTCP:

- RR: receiver report. Περιέχουν πληροφορία για την ποιότητα των δεδομένων στα σημεία της παραλαβής τους, καθώς επίσης και στατιστικά στοιχεία.
- SR: sender report. Δημιουργούνται από τους αποστολείς και περιέχουν πληροφορία για τα δεδομένα που στέλνονται.
- SDES: source description items. Περιέχουν πληροφορία για τις πηγές (sources) των δεδομένων.
- BYE: δηλώνει τέλος συμμετοχής.
- APP: application specific functions. Χρησιμοποιείται από τις εφαρμογές για την υποστήριξη ιδιαίτερων λειτουργιών οι οποίες δεν περιλαμβάνονται στον ορισμό του RTP/RTCP.

Μέσω των παραπάνω πακέτων ελέγχου, το RTCP παρέχει τις παρακάτω υπηρεσίες:

- QoS monitoring και congestion control. Είναι μια από τις βασικές λειτουργίες του RTCP. Το RTCP παρέχει πληροφορία (feedback) στις εφαρμογές για την ποιότητα της μετάδοσης των δεδομένων. Το RTCP χρησιμοποιεί μετάδοση multicast και είναι εύκολο με αυτόν τον τρόπο όλα τα μέλη μιας τηλεδιάσκεψης να αποστέλλουν και να λαμβάνουν στοιχεία σχετικά με την ποιότητα της εφαρμογής. Για παράδειγμα οι παραλήπτες δεδομένων πραγματικού χρόνου μπορούν από τα RTCP πακέτα του αποστολέα να συμπεράνουν το ρυθμό μετάδοσης δεδομένων και να εκτιμήσουν την τελική ποιότητα μιας εφαρμογής τηλεδιάσκεψης. Παρόμοια ο αποστολέας δεδομένων μπορεί από τα RTCP πακέτα των παραληπτών να λάβει γνώση της ποιότητας της εφαρμογής τηλεδιάσκεψης και να εκτιμήσει το αν προβλήματα παρουσιάζονται μόνο σε μια στενή ομάδα χρηστών ή αν τα προβλήματα αυτά είναι καθολικά. Ανάλογα με την εκτιμούμενη κατάσταση είναι δυνατόν να ληφθούν μέτρα για τη διόρθωσή τους. Ένα τέτοιο ενδεχόμενο είναι να μειωθεί η ποιότητα των συμπιεσμένων δεδομένων προκειμένου να επιτευχθεί χαμηλότερος ρυθμός μετάδοσης δεδομένων ώστε να μειωθούν οι απώλειές τους.
- Source identification. Στα πακέτα δεδομένων του RTP, οι πηγές (sources) αναγνωρίζονται μέσω 32 bit identifiers οι οποίοι παράγονται τυχαία. Οι identifiers αυτοί δεν είναι βολικοί για τους ανθρώπους. Τα RTCP SDES (source description) πακέτα περιέχουν textual πληροφορία που καλείται canonical names (CNAME), στη θέση των identifiers. Αυτό το CNAME χρησιμοποιείται για να παρακολουθούνται τα άτομα που συμμετέχουν σε μια RTP σύνοδο. Το RTCP παρέχει την δυνατότητα στον αποστολέα δεδομένων να εσωκλείσει πληροφορία για την ταυτότητά του σε μορφή κειμένου μέσα στα RTCP πακέτα. Είναι ευκολότερο τότε ιδιαίτερα στους παραλήπτες που μετέχουν ταυτόχρονα σε πολλές τηλεδιασκέψεις να συσχετίζουν μια ροή δεδομένων με συγκεκριμένη τηλεδιάσκεψη. Επίσης, οι παραλήπτες χρησιμοποιούν το CNAME για να συνδέουν πολλαπλές ροές δεδομένων από ένα συγκεκριμένο

άτομο που συμμετέχει στη σύνοδο σε ένα σύνολο συνδεδεμένων RTP συνόδων (π.χ. για το συγχρονισμό ήχου και εικόνας).

- Inter-media synchronization. Τα RTCP sender reports περιέχουν πληροφορίες για τον πραγματικό χρόνο και το αντίστοιχο RTP timestamp. Η πληροφορία αυτή μπορεί να χρησιμοποιηθεί για εσωτερικό συγχρονισμό διαφορετικών streams (π.χ. εικόνα και ήχος σε ένα video stream). Όπως αναφέραμε και προηγουμένως, κατά την μετάδοση multimedia δεδομένων με την χρήση RTP/RTCP, κάθε media μεταδίδεται σε ξεχωριστό stream.
- Control information scaling. Τα RTCP πακέτα στέλνονται περιοδικά ανάμεσα σε αυτούς που συμμετέχουν στην σύνοδο. Καθώς ο αριθμός των συμμετεχόντων αυξάνεται, γίνεται απαραίτητη η αποκατάσταση κάποιας ισορροπίας ανάμεσα στην πληροφορία ελέγχου που ανταλλάσσεται και στο φόρτο του δικτύου. Το πρωτόκολλο RTCP χρησιμοποιείται για την εκτίμηση του πλήθους των μελών μιας τηλεδιάσκεψης, καθώς κάθε μέλος αποστέλλει τακτικά μηνύματα RTCP. Καθώς όμως ο αριθμός των μελών μιας συνόδου αυξάνεται, μεγαλώνει επίσης και ο αριθμός των RTCP πακέτων που κυκλοφορούν στο δίκτυο. Για να αποτραπεί η κατανάλωση όλων των πόρων του δικτύου από τον έλεγχο κυκλοφορίας και για να επιτραπεί στο RTP να εξυπηρετεί έναν μεγάλο αριθμό ατόμων που συμμετέχουν σε μια σύνοδο, ο έλεγχος κυκλοφορίας περιορίζεται στο πέντε τοις εκατό το πολύ της συνολικής κυκλοφορίας συνόδου. Αυτό το όριο επιβάλλεται ρυθμίζοντας το ρυθμό με τον οποίο τα RTCP πακέτα μεταδίδονται σαν μια συνάρτηση των ατόμων που συμμετέχουν. Αφού ο κάθε συμμετέχων στέλνει πακέτα ελέγχου σε όλους τους άλλους, ο καθένας μπορεί να παρακολουθεί τον συνολικό αριθμό των συμμετεχόντων και να χρησιμοποιεί τον αριθμό αυτό για να υπολογίζει το ρυθμό με τον οποίο πρέπει να στέλνει RTCP πακέτα.

Η μετάδοση δεδομένων RTCP αποτελείται από μία δέσμη διαφορετικών τύπων πακέτων, τα οποία ενσωματώνονται σε ένα UDP datagram ή σε διαφορετικό τύπο πακέτου αν χρησιμοποιείται διαφορετικό πρωτόκολλο. Τα είδη των RTCP πακέτων είναι τα ακόλουθα:

- Sender Report (SR) και Receiver Report (RR). Οι παραλήπτες πληροφορίας σε ένα RTP session επιστρέφουν στον εκάστοτε αποστολέα δεδομένα που αφορούν την ποιότητα μετάδοσης. Ο αποστολέας είναι σε θέση από τις αναφορές αυτές να διαπιστώσει το είδος ενός προβλήματος, το αν περιορίζεται σε μία στενή γεωγραφική περιοχή ή εξαπλώνεται σε πολύ μεγαλύτερη. Έχουν μάλιστα δημιουργηθεί συστήματα τα οποία παρακολουθούν μόνο τα RTCP πακέτα, και όχι τα RTP, από τα οποία εξάγουν συμπεράσματα σχετικά με την απόδοση του multicast IP στα δίκτυα που παρακολουθούν. Αν ένα μέλος μιας συνόδου είναι μόνο παραλήπτης πληροφορίας αποστέλλει "Receiver Report", ενώ αν είναι και αποστολέας πληροφορίας αποστέλλει "Sender Report".
- Source Description (SDES). Είναι ο τύπος του πακέτου που χρησιμοποιείται για να δίνουν τα μέλη μιας συνόδου πληροφορίας σχετικές με τον εαυτό τους, για παράδειγμα όνομα, διεύθυνση ηλεκτρονικού ταχυδρομείου το όνομα της εφαρμογής που χρησιμοποιείται στη σύνοδο καθώς και άλλα στοιχεία. Στον πίνακα 3.3 καταγράφονται οι τύποι πληροφορίας που περιέχονται σε ένα SDES/RTCP πακέτο:

Value	Name	Description
0	END	End of SDES list.
1	CNAME	Canonical name: unique among all participants within one RTP session
2	NAME	Real user name of the source
3	EMAIL	E-mail address
4	PHONE	Telephone number
5	LOC	Geographic Location
6	TOOL	Name of application generating the stream
7	NOTE	Transient message describing the current state of the source
8	PRIV	Private experimental or application-specific extensions

Πίνακας 3.3: **Τα είδη πληροφορίας ενός** SDES/RTCP **πακέτου**.

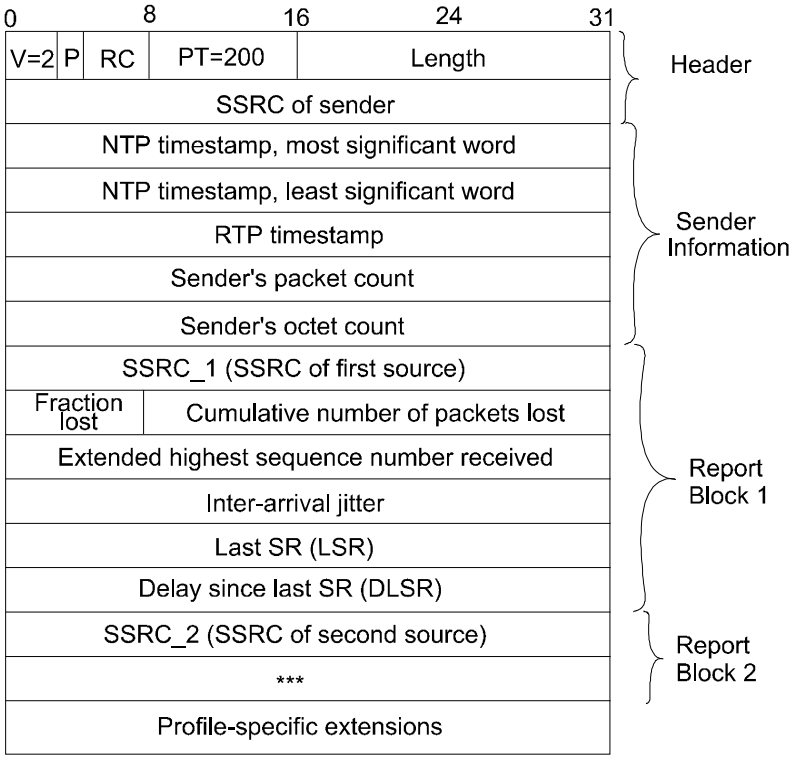
- Goodbye (BYE). Ο τύπος αυτός σηματοδοτεί την αποχώρηση από τη σύνοδο ενός ή περισσοτέρων μελών. Για την επισήμανση των μελών που αποχωρούν το πακέτο περιέχει τη λίστα με τους αντίστοιχους κωδικούς των πηγών. Συνηθίζεται τα μέλη να εσωκλείσουν στο πακέτο και τους λόγους για τους οποίους αποχωρούν από τη σύνοδο. Όταν ένας μίκτης λαμβάνει τέτοιο πακέτο αφήνει τους κωδικούς πηγών πληροφορίας αναλλοίωτους.
- Application specific (APP). Ο ειδικός αυτός τύπος πακέτου χρησιμοποιείται για πειραματικούς σκοπούς στην ανάπτυξη νέων εφαρμογών και χαρακτηριστικών χωρίς να απαιτείται να γίνουν επίσημα αποδεκτοί νέοι τύποι μέσων επικοινωνίας. Μετά από τις αναγκαίες δοκιμές και αφού ο νέος τύπος μετάδοσης πληροφορίας ή τα νέα χαρακτηριστικά γίνουν ευρέως αποδεκτά γίνεται αίτηση στον οργανισμό IANA για την επίσημη κατοχύρωση του συγκεκριμένου τύπου πακέτου.

Παρακάτω παρουσιάζονται αναλυτικότερα τα παραπάνω πακέτα:

Sender / Receiver reports (Αναφορές Αποστολέα / Παραλήπτη)

Οι αποστολείς σε μια περιοδική σύσκεψη μεταφέρουν πακέτα αναφοράς αποστολέα για να ενημερώσουν τα άλλα μέρη για το τι έπρεπε να λάβουν. Το πακέτο αναφοράς αποστολέα (sender report) αποτελείται από 3 τμήματα, πιθανώς ακολουθούμενα από ένα τέταρτο ιδιαιτέρου προφίλ προέκτασης τμήμα αν έχει οριστεί.

Η μόνη διαφορά μεταξύ της αναφοράς αποστολέα και δέκτη, επί πλέον του κώδικα τύπου πακέτου, είναι ότι η αναφορά αποστολέα περιλαμβάνει ένα 20-byte τμήμα πληροφορίας αποστολέα για χρήση από ενεργούς αποστολείς. Στο σχήμα 3.3 φαίνεται η δομή και των δύο πακέτων. Οι διαφορές μεταξύ των δύο πακέτων περιγράφονται στην συνέχεια.



Σχήμα 3.3: **Δομή πακέτων** Sender / Receiver reports.

Το πρώτο τμήμα (header), έχει μέγεθος 8 octets και περιλαμβάνει τα παρακάτω πεδία:

- Version (έκδοση V). Προσδιορίζει την έκδοση του RTP, που είναι η ίδια με τα RTCP πακέτα καθώς και τα RTP πακέτα δεδομένων. Επί του παρόντος αυτό είναι 2.
- Padding (P). Αν το padding bit είναι set, το RTCP πακέτο περιέχει μερικά επιπρόσθετα padding octets στο τέλος που δεν είναι μέρος της πληροφορίας ελέγχου.
- Απαρίθμηση αναφοράς λήψης (Reception report count RC). Ο αριθμός των block αναφοράς λήψης περιέχονται σε αυτό το πακέτο. Μια τιμή μηδέν είναι έγκυρη.
- Τύπος πακέτου (Packet type PT). Περιέχει τη σταθερά 200 για να αναγνωρίσει αυτό ως ένα RTCP SR πακέτο ή την σταθερά 201 για να αναγνωρίσει αυτό ως ένα RTCP RR πακέτο.
- Μήκος (length). Το μήκος του RTCP πακέτου σε 32-bit λέξεις, περιλαμβάνοντας το πρώτο τμήμα (header) και κάθε padding.
- SSRC. Ο αναγνωριστής συγχρονισμού πηγής για τη δημιουργία αυτού του SR (ή RR) πακέτου.

Το δεύτερο τμήμα, πληροφορία αποστολέα, έχει μέγεθος 20 octets και είναι παρόν σε κάθε πακέτο αναφοράς αποστολέα. Αυτό συνοψίζει τη μεταφορά δεδομένων από τον αποστολέα. Τα πεδία έχουν το ακόλουθο μήνυμα:

- NTP timestamp. Δείχνει το χρόνο του wall-clock όταν η αναφορά έχει σταλεί έτσι που να μπορεί να χρησιμοποιηθεί σε συνδυασμό με τα timestamps που επιστρέφονται σε αναφορές λήψης από άλλους δέκτες ώστε να μετρούν του round-trip χρόνους μετάδοσης ανάμεσα σε εκείνους και τους δέκτες.

- RTP timestamp. Ανταποκρίνεται στον ίδιο χρόνο με το NTP timestamp, αλλά στις ίδιες μονάδες και με την ίδια τυχαία αντιστάθμιση όπως τα RTP timestamps σε πακέτα δεδομένων.
- Απαρίθμηση πακέτου δέκτη. Ο συνολικός αριθμός RTP πακέτων δεδομένων που έχουν μεταφερθεί από τον αποστολέα από την έναρξη της μεταφοράς μέχρι τη στιγμή που έχει δημιουργηθεί το SR πακέτο. Η απαρίθμηση επαναφέρεται σε αρχικές συνθήκες αν ο αποστολέας αλλάξει το πεδίο SSRC.
- Απαρίθμηση octet αποστολέα. Ο συνολικός αριθμός των ωφέλιμων octets (που δεν περιλαμβάνουν header or padding) που μεταφέρονται σε RTP πακέτα δεδομένων από τον αποστολέα από την έναρξη της μεταφοράς μέχρι τη στιγμή που έχει δημιουργηθεί το SR πακέτο. Η απαρίθμηση επαναφέρεται σε αρχικές συνθήκες αν ο αποστολέας αλλάξει το πεδίο SSRC.

Το τρίτο τμήμα περιέχει μηδέν ή περισσότερα block αναφοράς λήψης στηριζόμενα στον αριθμό άλλων πηγών του αποστολέα ως την τελευταία αναφορά. Κάθε block αναφοράς λήψης μεταβιβάζει στατιστικά στη λήψη RTP πακέτων από μια απλή πηγή συγχρονισμού. Αυτά τα στατιστικά είναι:

- SSRC_n (αναγνωριστής πηγής). Ο SSRC αναγνωριστής πηγής στην οποία αναφέρεται η πληροφορία στο block αναφοράς λήψης.
- Χαμένο κλάσμα (Fraction lost). Το κλάσμα του RTP πακέτου δεδομένων από την πηγή SSRC_n χαμένα από το προηγούμενο SR ή RR πακέτο που έχει σταλεί. Αυτό το κλάσμα ορίζεται ως ο αριθμός χαμένων πακέτων δια του αριθμού των πακέτων αναμένονται.
- Αθροιστικός αριθμός χαμένων πακέτων. Ο συνολικός αριθμός RTP πακέτων δεδομένων από την πηγή SSRC_n που έχουν χαθεί από την έναρξη της λήψης.
- Αριθμός εκτεταμένης υψηλότερης ακολουθίας που έχει ληφθεί (Extended highest sequence number received). Το χαμηλό 16 bits περιέχει τον αριθμός υψηλότερης ακολουθίας που έχει ληφθεί σε ένα RTP πακέτο δεδομένων από πηγή SSRC_n, και τα πιο σημαντικά 16 bits παρατείνουν τον αριθμό ακολουθίας με αντιστοιχία απαρίθμησης από κύκλους αριθμού ακολουθίας. Διαφορετικοί δέκτες στο ίδιο τμήμα θα δημιουργήσουν διαφορετικές προεκτάσεις στον αριθμό ακολουθίας αν οι χρόνοι έναρξής τους διαφέρουν σημαντικά.
- Inter-arrival jitter. Ένας υπολογισμός της στατιστικής διακύμανσης του inter-arrival χρόνου ενός RTP πακέτου δεδομένων, μετρημένο σε μονάδες timestamp και καθορισμένο σαν ένας μη προσημασμένος ακέραιος.
- Τελευταίο SR timestamp. Το μέσο 32 bits εκτός 64 στο NTP timestamp που έχουν ληφθεί ως κομμάτι του πιο πρόσφατου RTCP αναφοράς αποστολέα (SR) πακέτου από πηγή SSRC_n. Αν κανένα SR δεν έχει ληφθεί ακόμα, το πεδίο είναι στο μηδέν.
- Καθυστέρηση από το τελευταίο SR. Η καθυστέρηση καθορισμένη σε μονάδες από 1/65536 δευτερόλεπτα, μεταξύ λαμβάνοντας το τελευταίο SR πακέτο από την πηγή SSRC_n και στέλνοντας το block αναφοράς λήψης. Αν κανένα SR δεν έχει ληφθεί ακόμα, το πεδίο είναι στο μηδέν.

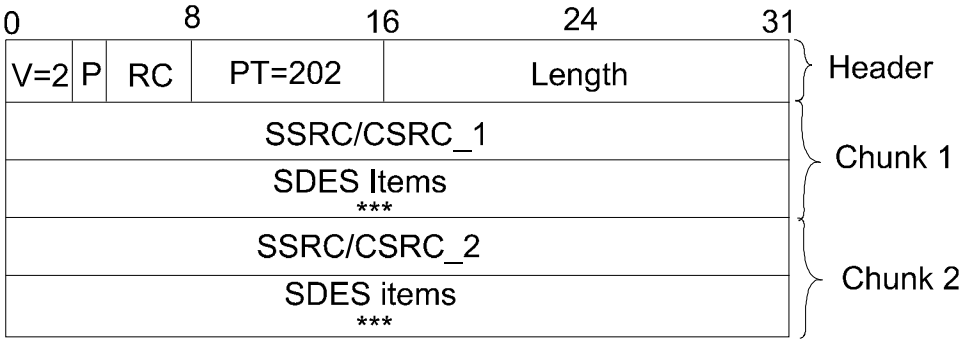
Source description (Περιγραφείς πηγής)

Ο τρίτος τύπος RTCP πακέτου, η περιγραφή πακέτου ή SDES πακέτου είναι μια τριών επιπέδων δομή συγκροτημένη από ένα header και μηδέν ή περισσότερα blocks όπως φαίνεται και στην σχήμα 3.4. Το SDES πακέτο περιέχει τα παρακάτω πεδία:

- Έκδοση (Version (V)), περίβλημα (padding (P)), μήκος (length).). Όπως περιγράφεται για το πακέτο Αναφοράς Αποστολέα.
- Packet type (PT). Περιέχει τη σταθερά 202 να αναγνωρίσει αυτό σαν ένα RTCP SDES πακέτο.
- Source count (SC). Ο αριθμός των SSRC/CSRC κομματιών που περιέχονται σε αυτό το SDES πακέτο. Η τιμή μηδέν είναι έγκυρη, αλλά χωρίς νόημα. Κάθε κομμάτι αποτελείται από έναν SSRC/CSRC προσδιοριστή ακολουθούμενο από μια λίστα μηδενικών ή περισσότερων στοιχείων, η οποία μεταφέρει πληροφορία για το SSRC/CSRC και ξεκινά από ένα 32-bit όριο. Κάθε στοιχείο αποτελείται από ένα 8-bit πεδίο εγγραφής, από μια οκταδική μέτρηση που περιγράφει το μήκος του κειμένου (σ΄ αυτό δεν περιέχεται η δυαδικο-οκταδική κεφαλίδα) και το κείμενο. Ο πίνακας 3.4 περιλαμβάνει τα τρέχοντα στοιχεία που χρησιμοποιούνται στο SDES.

CNAME	**Περιγραφή**
NAME	Όνομα χρήστη
EMAIL	Ηλεκτρονική διεύθυνση
PHONE	Αριθμός Τηλεφώνου
LOC	Γεωγραφική τοποθεσία χρήστη
TOOL	Όνομα εφαρμογής ή εργαλείου
NOTE	Σημείωση / κατάσταση
PRIV	Ιδιωτικές προεκτάσεις

Πίνακας 3.4: **Οι τιμές του** CNAME

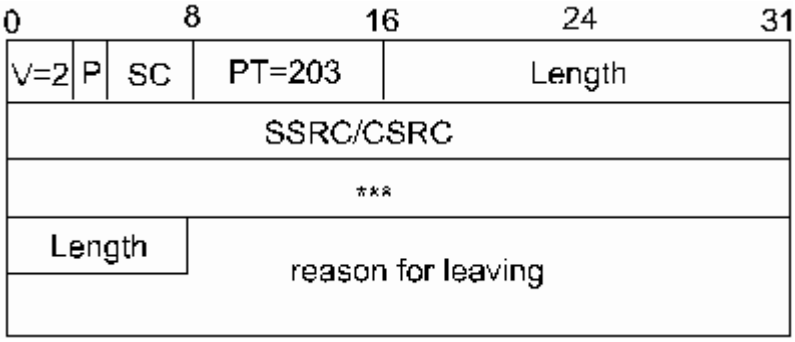


Σχήμα 3.4: Source description

BYE (Πακέτο αποχωρισμού)
Μια πηγή χρησιμοποιεί ένα Bye πακέτο για να ανακοινώσει ότι αφήνει μια σύνοδο, παρόλο που οι άλλοι συμμετέχοντες θα επισημάνουν τελικά την απουσία της πηγής χωρίς αυτό το πακέτο, αυτό κάνει τα πράγματα πιο γρήγορα, γεγονός που μπορεί να οδηγήσει σε πιο αποδοτική χρήση του bandwidth που είναι δυνατόν να χρησιμοποιηθεί. Το BYE πακέτο περιλαμβάνει τα παρακάτω πεδία(Σχήμα 3.5):

- Έκδοση (Version (V)), περίβλημα (padding (P)), μήκος (length). Όπως περιγράφεται για το πακέτο Αναφοράς Πηγής.
- Τύπος Πακέτου (Packet type (PT)). Περιέχει την σταθερά 203 για να την καθορίσει σαν ένα RTCP Bye πακέτο.

- Μέτρηση Πηγής (Source count (SC)). Ο αριθμός των SSRC/CSRC προσδιοριστών που εμπεριέχονται σε αυτό το Bye πακέτο.
- Λόγος αποχώρησης (reason for leaving).

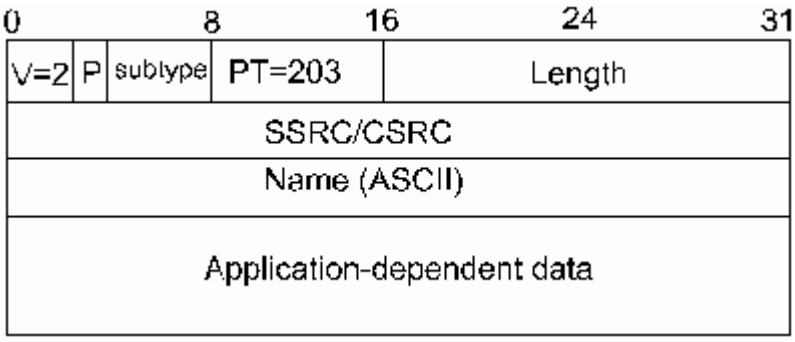


Σχήμα 3.5: **Δομή BYE πακέτου**.

APP Πακέτο

Το App πακέτο(σχήμα 3.6) επινοήθηκε για πειραματική χρήση καθώς νέες εφαρμογές και νέα χαρακτηριστικά αναπτύσσονταν, χωρίς να απαιτούν καταχώριση τιμής του τύπου πακέτου. Σε περίπτωση που μια ιδιαίτερα ειδική εφαρμογή αποδεικνυόταν χρήσιμη, το πιο πιθανόν θα ήταν να μετατραπεί σε ένα νέο RTCP πακέτο με τη τον δικό του επίσημο τύπο πακέτου :

- Έκδοση (Version (V)), περίβλημα (padding (P)), μήκος (length).). Όπως περιγράφεται για το πακέτο Αναφοράς Αποστολέα.
- Υποτύπος (Subtype). Ίσως χρησιμοποιείται σαν υποτύπος για να επιτρέψει ένα σετ από App πακέτα να προσδιορίζονται κάτω από μοναδικό όνομα ή για κάθε δεδομένο εξαρτημένο από την εφαρμογή.
- Τύπος Πακέτου (Packet type (PT)). Περιέχει την σταθερά 204 για να προσδιορίσει αυτό σαν ένα RTCP App πακέτο.
- Όνομα (Name). Ένα όνομα διαλεγμένο από το πρόσωπο που προσδιορίζει το σετ των App πακέτων να είναι μοναδικό με σεβασμό στα άλλα App πακέτα που ίσως η εφαρμογή αυτή λαμβάνει.
- Δεδομένα εξαρτημένα από την Εφαρμογή (Application-dependent data). Δεδομένα εξαρτώμενα από την εφαρμογή ίσως να εμφανίζονται ίσως και να μην εμφανίζονται σε ένα App πακέτο. Αυτά μεταφράζονται από την εφαρμογή και όχι από το RTP. Πρέπει να είναι πολλαπλάσιο μήκος των 32 bits.



Σχήμα 3.6: **Δομή** APP **Πακέτο**.

ΚΕΦΑΛΑΙΟ 4

SDP

ΚΕΦΑΛΑΙΟ 4: SDP

Το πρωτόκολλο περιγραφής Συνόδου SDP (Session Description Protocol) είναι ένα μορφότυπο(format) για να περιγράψουμε τις παραμέτρους εκκίνησης των μέσων ροής

(media streams) με ένα ASCII αλφαριθμητικό. Δημοσιεύτηκε από την IETF ως RFC 4566 τον Απρίλιο του 1998 και ανανεώθηκε στην πιο πρόσφατή του έκδοση τον Ιούλιο του 2006.

Σκοπός του SDP είναι να πιστοποιεί την ύπαρξη ενός session και να παρέχει πληροφορίες για να γίνει εφικτή η σύνδεση και η συμμετοχή σε ένα session. Δεν αντικαθιστά ένα πρωτόκολλο μεταφοράς (transport protocol) αλλά προορίζεται να χρησιμοποιεί διαφορετικά πρωτόκολλα σηματοδοσίας και μεταφοράς όπως τα Session Announcement Protocol(SAP), Session Initiation Protocol(SIP), Real Time Streaming Protocol(RTSP), e-mail χρησιμοποιώντας MIME extensions, Hypertext Transport Protocol(http).

Το SDP περιέχει πληροφορίες όπως:

- Το όνομα και το σκοπό του session.
- Χρονολογίες που το session είναι ενεργό.
- Τα πολυμέσα(media) που εμπεριέχονται στο session
- Πληροφορίες για την λήψη αυτών των media(π.χ. διευθύνσεις, πόρτες, διαμορφώσεις κ.α.).

Το SDP πρέπει να παρέχει επαρκείς πληροφορίες για να μπορέσουν οι εφαρμογές να συμμετέχουν σε ένα session (με την πιθανή εξαίρεση τα κλειδιά αποκρυπτογράφησης – encryption keys) και να ανακοινώνει τους πόρους που χρειάζεται κάποιος για να συμμετέχει στο session(συνήθως όταν χρησιμοποιείται με ένα Multicast Session Announcement Protocol).

Για τις media πληροφορίες το SDP περιέχει:

- το είδος των Media (βίντεο, ήχος κτλ),
- το πρωτόκολλο μεταφοράς (RTP/UTP/IP, H.320, κτλ),
- τη διαμόρφωση των media (H.261 video, MPEG video, κτλ),

Επιπλέον παρέχει πληροφορίες για τις διευθύνσεις και πόρτες όπου για IP Multicast Sessions σημαίνει:

- την ομάδα των Multicast διευθύνσεων(Multicast group address) για τα media
- και την πόρτα(transport port) των media .

και για IP Unicast Sessions σημαίνει:

- την Remote Address(απομακρυσμένη διεύθυνση) των media
- και την Remote Port(απομακρυσμένη πόρτα) των media.

Για πληροφορίες χρόνου το SDP μπορεί να περιέχει:

- μια λίστα από χρονικές στιγμές αρχής και τέλους του session.

- επαναλαμβανόμενες περιόδους για κάθε όριο, όπως για παράδειγμα “κάθε Δευτέρα στις 10π.μ για 3 ώρες”.

Το SDP μπορεί να περιέχει πρόσθετους δείκτες με τη μορφή των Universal Resources Identifiers (URIs) για περισσότερες πληροφορίες.

4.1 ΑΡΧΙΤΕΚΤΟΝΙΚΗ SDP

Οι περιγραφές του SDP είναι κυρίως σε μορφή κειμένου γραμματοσειράς ISO 10646 και κωδικοποίηση UTF-8. Μια SDP περιγραφή ενός session αποτελείται από μια σειρά γραμμών κειμένου που έχουν την μορφή:

- <type>=<value>.

Όπου <type> είναι πάντα ένας χαρακτήρας case significant(με διαφορά πεζών /κεφαλαίων) και <value> που είναι είτε ένας αριθμός πεδίων χωρισμένα από ένα κενό χαρακτήρα είτε ένα αλφαριθμητικό και είναι case significant .

Η περιγραφή ενός session αποτελείται από τρία μέρη:

- την περιγραφή του session (session description)
- την περιγραφή των χρόνων(time description)
- και την περιγραφή των πολυμέσων(media description)

Μια ανακοίνωση περιέχει ένα τμήμα σε επίπεδο session και ακολουθούν μηδέν, ένα ή περισσότερα τμήματα σε επίπεδο media. Στο κομμάτι του session-level η πρώτη γραμμή ξεκινάει με “v=”. Στην media περιγραφή η πρώτη γραμμή αρχίζει με “m=”. Σε κάθε περιγραφή μερικές γραμμές απαιτούνται και άλλες όχι, αλλά όλες εμφανίζονται με τη σειρά που φαίνεται παρακάτω. Οι προαιρετικές εμφανίζονται με ‘’*’’:

Session description
- v= (protocol version)
- o= (originator and session identifier)
- s= (session name)
- i=* (session information)
- u=* (URI of description)
- e=* (email address)
- p=* (phone number)
- c=* (connection information -- not required if included in all media)
- b=* (zero or more bandwidth information lines)
- One or more time descriptions ("t=" and "r=" lines; see below)
- z=* (time zone adjustments)
- k=* (encryption key)
- a=* (zero or more session attribute lines)
- Zero or more media descriptions

Time description

t= (time the session is active)
r=* (zero or more repeat times)

Media description, if present
m= (media name and transport address)
i=* (media title)
c=* (connection information -- optional if included at session level)
b=* (zero or more bandwidth information lines)
k=* (encryption key)
a=* (zero or more media attribute lines)

Το παρακάτω είναι ένα παράδειγμα μιας SDP περιγραφής:

```
v=0
o=mhandley 2890844526 2890842807 IN IP4 126.16.64.4
s=SDP Seminar
i=A Seminar on the session description protocol
u=http://www.cs.ucl.ac.uk/staff/M.Handley/sdp.03.ps
e=mjh@isi.edu (Mark Handley)
c=IN IP4 224.2.17.12/127
t=2873397496 2873404696
a=recvonly
m=audio 49170 RTP/AVP 0
m=video 51372 RTP/AVP 3
m=application 32416 udp wb
a=orient:portrait
```

Οι παράμετροι έχουν την εξής σημασία:

- "v": Δίνει την έκδοση του SDP.
- "o": <username> <session id> <version> <network type> <address type> <address>: Όπου username, είναι το όνομα του χρήστη στη μηχανή που δημιούργησε το session. Το session id είναι ένα νούμερο που είναι μοναδικό και καθορίζει το session. Το Version καθορίζει την έκδοση της ανακοίνωσης. Το network type είναι ένα κείμενο που καθορίζει το είδος του δικτύου. Το "IN" στο συγκεκριμένο παράδειγμα σημαίνει "Internet". Το address type είναι ένα κείμενο που δίνει το είδος της διεύθυνσης που ακολουθεί. Στο παράδειγμα είναι "IP4". Το address δίνει τη διεύθυνση της μηχανής που δημιούργησε το session. Αν το domain name είναι διαθέσιμο, δίνεται. Διαφορετικά, όπως και στο παράδειγμα, δίνεται η IP διεύθυνση.
- "s": Δίνει το όνομα του session.
- "i": Δίνει μια περιγραφή του session.
- "u": Δίνει ένα URI όπου ο ενδιαφερόμενος μπορεί να μάθει περισσότερες πληροφορίες για το session.
- "e": Δίνει την e-mail διεύθυνση του υπεύθυνου του session.
- "c": <network type> <address type> <connection address>. Το network type είναι ένα κείμενο που δίνει το είδος του δικτύου. Στο παράδειγμα είναι "IN" δηλαδή Internet. Το address type είναι και αυτό ένα κείμενο που καθορίζει το είδος της

διεύθυνσης που ακολουθεί. Στο παράδειγμα είναι IP4. Τέλος το πεδίο connection address δίνει την IP multicast διεύθυνση που χρησιμοποιείται. Επίσης δίνει και το TTL με το οποίο στέλνονται τα πακέτα.

- "t": <start time> <stop time>. Αυτό το πεδίο καθορίζει πότε αρχίζει και πότε τελειώνει το session.

4.2 PAYLOAD TYPES FOR STANDARD AUDIO/VIDEO

Στον παρακάτω πίνακα 4.1 παραθέτουμε τον πίνακα με τα Payload Types (PT)για ήχο και εικόνα τα οποία χρησιμοποιούνται για την περιγραφή των πολυμέσων (media description) στο SDP.

PT	encoding name	audio/video (A/V)	clock rate (Hz)	channels (audio)
0	PCMU	A	8000	1
1	1016	A	8000	1
2	G721	A	8000	1
3	GSM	A	8000	1
4	unassigned	A	8000	1
5	DVI4	A	8000	1
6	DVI4	A	16000	1
7	LPC	A	8000	1
8	PCMA	A	8000	1
9	G722	A	16000	1
10	L16	A	44100	2
11	L16	A	44100	1
12	unassigned	A		
13	unassigned	A		
14	MPA	A	90000	(see text)
15	G728	A	8000	1
16–23	unassigned	A		
24	unassigned	V		
25	CelB	V	90000	
26	JPEG	V	90000	
27	unassigned	V		
28	nv	V	90000	
29	unassigned	V		
30	unassigned	V		
31	H261	V	90000	
32	MPV	V	90000	
33	MP2T	AV	90000	
34–71	unassigned	?		
72–76	reserved	N/A	N/A	N/A
77–95	unassigned	?		
96–127	dynamic	?		

Πίνακας 4.1: Payload Types.

Υπάρχει επιπλέον η δυνατότητα να ορίσουμε εμείς ένα νέο payload type ως εγγράφοντας το στην IANA και δίνοντας του μια στατική τιμή PT στο εύρος τιμών που δεν είναι ανατεθειμένες κάποιες τιμές(unasigned στον πίνακα).Τα PT με ετικέτα "unassigned" υπάρχουν γι` αυτόν τον σκοπό.

Μπορούμε επίσης να ορίσουμε δυναμικά PT μέσω ενός Conference Control Protocol π.χ. ένας κατάλογος συνόδων(session directory) μπορεί να ορίζει τον ότι για ένα προκαθορισμένο session το PT 96 να αντιστοιχεί σε κωδικοποίηση PCMU, με ρυθμό 8.000Hz και 2 κανάλια.

Τα PT που έχουν την ετικέτα "reserved" δεν χρησιμοποιούνται για να έχουμε αξιόπιστη διαφοροποίηση των RTCP και RTP πακέτων.

ΚΕΦΑΛΑΙΟ 5

JMF

ΚΕΦΑΛΑΙΟ 5: JMF

Το Java Media Framework (JMF) αποτελεί μια βιβλιοθήκη της Java που επιτρέπει σε Java εφαρμογές και μίνι-εφαρμογές(applets) να ενσωματώσουν ήχο και εικόνα. Δημιουργήθηκε

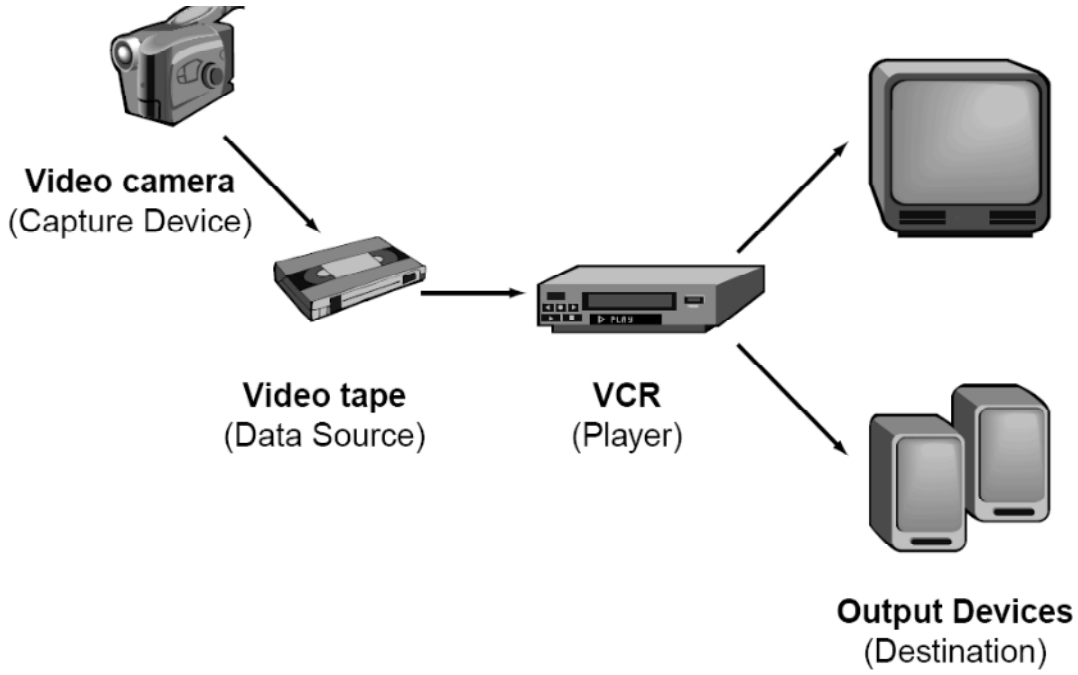
για πρώτη φορά το 1997(JMF 1.0) από τις εταιρίες Sun Microsystems ,Silicon Graphics και Intel . Σήμερα έχει ανανεωθεί στην έκδοση JMF 2.0 από τις Sun Microsystems και IBM.

5.1 ΑΡΧΙΤΕΚΤΟΝΙΚΗ

Αποτελείται από τέσσερα διαφορετικά αρχεία βιβλιοθηκών Java(JAR Files):

- JMStudio - A simple player GUI
- JMFRegistry - A GUI for managing the JMF "registry," which manages preferences, plug-ins, etc.
- JMFCustomizer - Used for creating a JAR file that contains only the classes needed by a specific JMF application, which allows developers to ship a smaller application.
- JMFInit

Το βασικό μοντέλο του JMF θυμίζει σε μεγάλο βαθμό το μοντέλο ενός κοινού Video Player (VCR) για εγγραφή, επεξεργασία και παρουσίαση time-based media όπως φαίνεται από την εικόνα 5.1 και το σχήμα 5.1.



Εικόνα 5.1: **Μοντέλο** VCR.

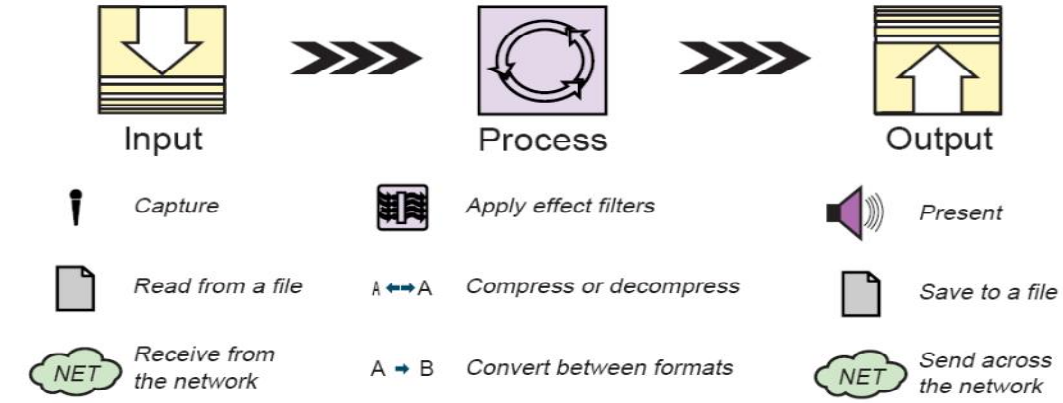
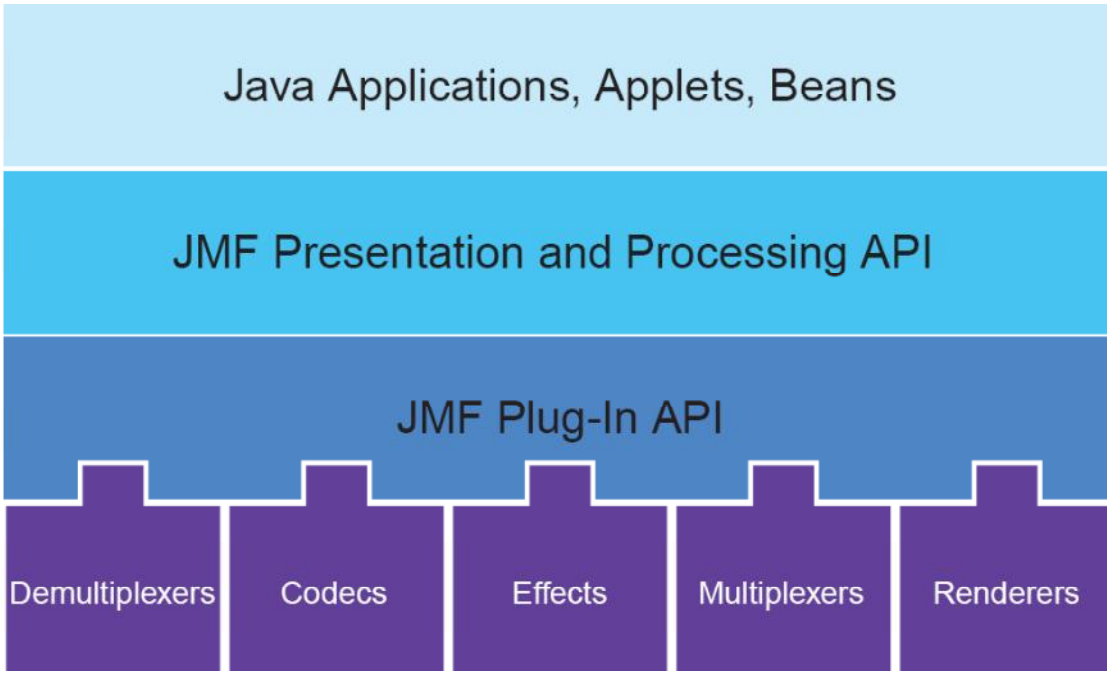


Figure 1-1: Media processing model.

Σχήμα 5.1: Media proccesing model of JMF.

Στο σχήμα 5.2 φαίνεται το ιεραρχικό μοντέλο στο οποίο βασίζεται η JMF .



Σχήμα 5.2: **Ιεραρχικό μοντέλο** JMF.

Το JMF βασίζεται σε πέντε βασικά μοντέλα (αντικείμενα):

- Time Model:όπου ένα Time Object χρησιμοποιείται για να αναπαραστήσει οποιοδήποτε σημείο του χρόνου χρησιμοποιώντας τρία βασικά σημεία τα:
    - Time-base Start-time :Δηλώνει πότε ξεκινά η παρουσίαση.
    - Media Start-time:την θέση από την οποία ξεκινάει η παρουσίαση.
    - Playback Rate:Τον ρυθμό αναπαράστασης.

  Και υπολογίζοντας τον χρόνο σύμφωνα με τον παρακάτω τύπο:

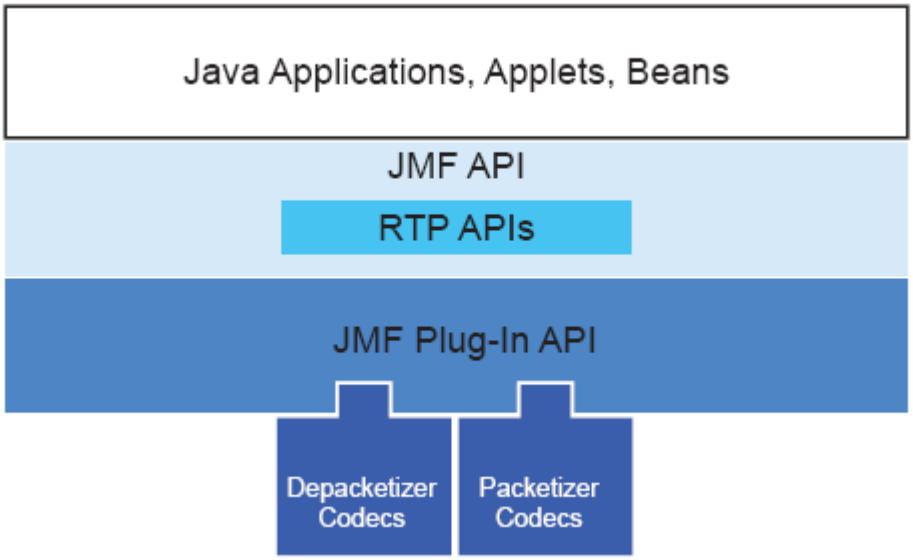
  MediaTime = MediaStartTime + Rate(TimeBaseTime - TimeBaseStartTime)

- Managers:Αποτελεί το Interface που ορίζει την συμπεριφορά και την αλληλεπίδραση των αντικειμένων που χρησιμοποιούνται για να καταγράψουν, να επεξεργαστούν και να παρουσιάσουν τα πολυμέσα. Χωρίζεται σε :
    - Manager:χρησιμοποιείται για την κατασκευή Players, Processors, Datasinks, DataSources.
    - PackageManager:Συντηρεί τα πακέτα που περιέχουν τα Players, Processors, Datasinks, DataSources.Για να επεκτείνουμε την βιβλιοθήκη πρέπει να εγγράψουμε τα πακέτα μας με τον PackageManager.
    - CaptureDeviceManager:Κατέχει εγγραφές από συσκευές σύλληψης(Capture Devices) που είναι διαθέσιμες.
    - PlugInManager:Περιέχει τα διαθέσιμα plug-in όπως (άπο)κωδικοποιητές , εφέ, κωδικοποιήσεις κ.α.
- Event Model:Βοηθάει για την ενημέρωση της κατάστασης του συστήματος πολυμέσων.
- Data Model:Βοηθάει στην διαχείριση των δεδομένων των πολυμέσων. Αποτελείται από τα:
    - DataSources: που χωρίζονται σε Push και PullDataSources
    - SpecialtyDataSources: που αποτελούνται από Cloneable και MergingDataSources
    - Format: που χωρίζονται σε Audio και VideoFormat.
- Controls:χρησιμοποιείται για να θέτουμε τα χαρακτηριστικά των αντικειμένων π.χ. τα PortControl και MonitorControl χρησιμοποιούνται για να δώσουν έλεγχο στον χρήστη στην διαδικασία της σύλληψης εικόνας ή ήχου(π.χ. έλεγχος στην κάμερα).

5.2 JMF/RTP

Με την βιβλιοθήκη JMF είναι δυνατόν αντί να αποθηκεύσουμε εικόνα και ήχο σε ένα αρχείο να μεταδώσουμε ή λάβουμε μέσω ενός δικτύου(Internet ή Intranet). Η JMF χρησιμοποιεί το πρωτόκολλο RTP για την αποστολή και λήψη streaming media.

Στο σχήμα 5.3 φαίνεται πώς ενσωματώνεται το RTP API μέσα στο ιεραρχικό μοντέλο της JMF.



Σχήμα 5.3: **Μοντέλο** JMF/RTP.

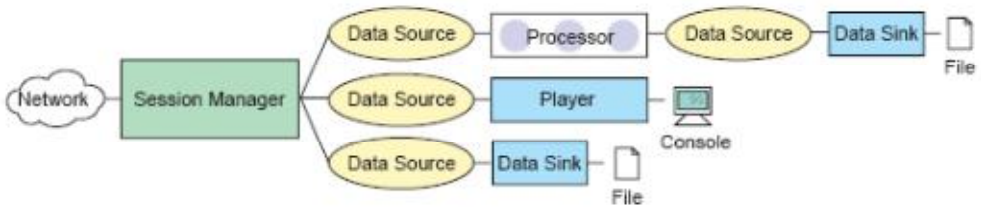
Τα πακέτα και οι κλάσεις του RTP API ακολουθούν την ίδια δομή της JMF έτσι έχουμε τα παρακάτω μοντέλα:

- SessionManager:Δημιουργεί και διαχειρίζεται το RTP session και υποδιαιρείται σε τρεις λογικές οντότητες τις SessionStatistics, SessionParticipants και SessionStreams οι οποίες διαχειρίζονται αντίστοιχα τα στατιστικά, τους συμμετέχοντες και τα streams της συνόδου.
- RTPEvents: Παρέχει μηχανισμούς που αναφέρουν την κατάσταση και τα λάθη του RTP Session. Υποδιαιρείται σε τέσσερα είδη συμβάντων: SessionListener,SendStreamListener,ReceiveStreamListener, RemoteListener.
- RTP Data: Προσδιορίζει τα RTP δεδομένα .Ορίζει τους DataHandlers οι οποίοι χρησιμοποιούνται είτε για το κανάλι δεδομένων είτε για το κανάλι ελέγχου (data channel,control channel) του RTP Session και ορίζει τα RTP Data Format (H261_RTP,ULAW_RTP κ.α.)
- RTPControls: Παρέχει μόνο το αντικείμενο RTPControl βοηθάει για να προσθέτεις μια αντιστοίχηση μεταξύ ενός dynamic payload και ενός format.Επίσης παρέχει μεθόδους για ανάκτηση των στατιστικών της συνόδου και του εκάστοτε payload format.

Με το JMF έχουμε την δυνατότητα να αναπαράγουμε, να αποθηκεύουμε (ή και τα δύο) RTP δεδομένα που λαμβάνουμε από το δίκτυο(RTP streams) όπως φαίνεται και στην σχήμα 5.4. Υπάρχουν δύο τρόποι για να μπορέσουμε να λάβουμε RTP streams:

1. Να χρησιμοποιήσουμε έναν MediaLocator ο οποίος θα περιέχει τις παραμέτρους του RTP session και να δημιουργήσουμε έναν Player καλώντας την μέθοδο Manager.createPlayer(MediaLocator).
2. Να δημιουργήσουμε έναν Player για κάποιο συγκεκριμένο Receive Stream παίρνοντας το DataSource του stream και καλώντας την μέθοδο Manager.createPlayer(DataSource).

Χρησιμοποιώντας τον πρώτο τρόπο (με τον MediaLocator) έχουμε την δυνατότητα να αναπαραστήσουμε μόνο το πρώτο RTP streams που θα ανιχνευτεί στο session. Αν χρειαζόμαστε να αναπαράγουμε πολλαπλά RTP streams (multiple RTP streams) σε ένα session πρέπει να διαλέξουμε τον δεύτερο τρόπο χρησιμοποιώντας την κλάση SessionManager και κατασκευάζοντας ένα Player για κάθε νέο ReceiveStream. Σε περίπτωση που δεν χρειαζόμαστε πολλαπλά RTP streams ο πρώτος τρόπος είναι προτιμότερος λόγω της απλότητας και της ευκολίας υλοποίησής του.

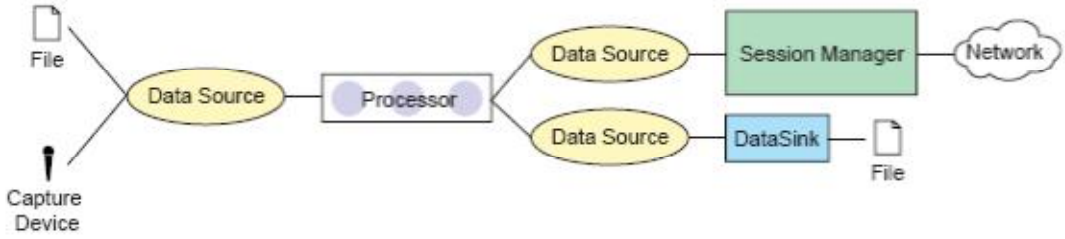


Σχήμα 5.4: RECEIVE.

Στην περίπτωση που θέλουμε να αποστείλουμε δεδομένα κάνουμε την αντίθετη εργασία από την παραπάνω όπως φαίνεται από το σχήμα 5.5. Όπως και στην λήψη έτσι και εδώ έχουμε δύο τρόπους να αποστείλουμε δεδομένα:

1. Χρησιμοποιώντας έναν MediaLocator ο οποίος περιέχει τις παραμέτρους του RTP session και δημιουργώντας ένα RTP DataSink με την βοήθεια της μεθόδου Manager.CreateDataSink(MediaLocator).
2. Να δημιουργήσουμε Send Streams χρησιμοποιώντας απευθείας την κλάση SessionManager για να στείλουμε δεδομένα και να ελέγξουμε την αποστολή.

Ο πρώτος τρόπος είναι πιο απλός και εύκολος στην υλοποίηση όμως είμαστε περιορισμένοι στο να στείλουμε μόνο το πρώτο Stream. Με τον δεύτερο τρόπο έχουμε την δυνατότητα να στέλνουμε πολλαπλά RTP streams σε ένα session και να έχουμε καλύτερο έλεγχο στο session.



Σχήμα 5.5: TRANSMIT.

Εκτός από την αναπαραγωγή των πολυμέσων που λαμβάνουμε ή στέλνουμε μέσω της JMF, έχουμε την δυνατότητα να αποθηκεύσουμε τα δεδομένα που λαμβάνουμε σε ένα αρχείο ή να διαβάσουμε δεδομένα από ένα αποθηκευμένο αρχείο και να τα στείλουμε στο δίκτυο. Οι παραπάνω επιλογές έχουν ιδιαίτερη αξία σε VoIP εφαρμογές καθώς ίσως να χρειαστεί να αποθηκεύσουμε μια συνομιλία ή να ακούγεται ένα ηχογραφημένο μήνυμα όταν μας καλούνε.

ΚΕΦΑΛΑΙΟ 6

ΥΛΟΠΟΙΗΣΗ

ΚΕΦΑΛΑΙΟ 6: ΥΛΟΠΟΙΗΣΗ

Σκοπός της εφαρμογής VA Messenger(Video and Audio Messenger) είναι η διαχείριση συνόδων μεταξύ δύο μερών μέσω του πρωτοκόλλου σηματοδοσίας SIP και πραγματοποίηση βίντεο κλήσης. Στο κεφάλαιο αυτό περιγράφονται τα βασικά στοιχεία της εφαρμογής και οι αποφάσεις που οδήγησαν σε αυτήν την υλοποίηση.

6.1 ΑΝΑΛΥΣΗ ΣΧΕΔΙΑΣΗΣ

Στις επόμενες ενότητες αναφέρονται τα βασικά στοιχεία λογισμικού και υλισμικού τα οποία συνθέτουν την εφαρμογή VA Messenger.

6.1.1 ΓΕΝΙΚΑ

Η εφαρμογή VA Messenger παρέχει δύο βασικές λειτουργίες, αυτή της εγγραφής και της κλήσης. Για να γίνει αυτό εφικτό χρειάστηκε να υλοποιηθεί ένας SIP User Agent ο οποίος είναι υπεύθυνος για την εγγραφή ενός χρήστη στο σύστημα και την βίντεο κλήση σε κάποιον απομακρυσμένο χρήστη. Για την εγγραφή των χρηστών χρειάστηκε η υλοποίηση ενός SIP Registrar Server ο οποίος είναι υπεύθυνος για την εξυπηρέτηση των εγγραφών των χρηστών καθώς και των SIP Proxy Servers στην εύρεση των θέσεων των χρηστών στο δίκτυο. Για την καταγραφή των θέσεων από τον Registrar χρειάστηκε η υλοποίηση ενός Location Service το οποίο αποτελεί την βάση δεδομένων με τις εγγραφές των χρηστών(binds).

6.1.2 ΒΙΒΛΙΟΘΗΚΕΣ

Ως βασική βιβλιοθήκη που παρέχει την υλοποίηση του SIP πρωτοκόλλου επιλέχθηκε μια ελεύθερη βιβλιοθήκη η MjSip (κάτω από τους όρους της GNU GPL license (General Public Licence) η οποία αποτελεί μια προσπάθεια των τμημάτων Dpt. of Information Engineering στο University of Parma και DIE - University of Roma “Tor Vergata”. Επίσης χρησιμοποιήθηκαν οι βιβλιοθήκες:

- JMF API: Για την υλοποίηση των λειτουργιών του Media Session.
- mysql-connector-java-5.1.6:Για την επικοινωνία με την βάση.

6.1.3 ΓΙΑΤΙ MYSQL

Για την συντήρηση των εγγραφών στο Location Service επιλέχθηκε η λύση της βάσης δεδομένων με κύριο μέλημα την επεκτασιμότητα και την ταχύτητα με την αύξηση του όγκου δεδομένων. Η MySQL αποτελεί ένα σύστημα διαχείρισης Σχεσιακών Βάσεων Δεδομένων (RDBMS) το οποίο συνοδεύεται από έναν δωρεάν MySQL Community Server(free software under the GNU General Public License (GPL)). Επίσης η ύπαρξη βιβλιοθήκης για επικοινωνία με Java εφαρμογές καθώς και το πλήρες documentation ήταν δύο παράγοντες που ενθάρρυναν την επιλογή της.

6.1.4 ΑΠΑΙΤΗΣΕΙΣ ΣΥΣΤΗΜΑΤΟΣ

Για την εγκατάσταση του User Agent(UAC Client) χρειάζονται:

ΥΛΙΚΟ:

- Κάμερα συμβατή με την βιβλιοθήκη JMF(συμβατοί τύποι κατασκευαστών καθώς και κωδικοποιήσεις υπάρχουν στο [23]).
- Συσκευή μικροφώνου.
- Ηχεία.
- 8 Mb RAM.
- 16 Mb ελεύθερα σον σκληρό δίσκο.

ΛΟΓΙΣΜΙΚΟ:

- Windows XP (or Later),Unix,Linux
- Jvm 6 or Later
- Java Libraries
    - AbsoluteLayout.jar
    - Jmf.jar
    - Sip.jar
    - Swing-layout-1.0.3.jar
    - Toplink-essentials.jar
    - Toplink-essentials-agent.jar
    - Ua.jar

Για την εγκατάσταση του Registrar Server(Registrar_sql) και Location Service χρειάζονται:

ΥΛΙΚΟ:

- 8 Mb RAM.
- 16 Mb ελεύθερα σον σκληρό δίσκο.(Αρχικά και αυξάνει ανάλογα με το μέγεθος της βάσης του LocationService)

ΛΟΓΙΣΜΙΚΟ:

- Windows XP (or Later),Unix,Linux
- Jvm 6 or Later
- Java Libraries
    - AbsoluteLayout.jar
    - Jmf.jar
    - Sip.jar
    - Swing-layout-1.0.3.jar
    - Toplink-essentials.jar
    - Toplink-essentials-agent.jar
    - Ua.jar
    - Mysql-connector-java-5.1.jar
- MySQL Server Version 5.0 or later

6.2 CLIENT

Αποτελεί την τηλεφωνική εφαρμογή με την οποία αλληλεπιδρά ο χρήστης μέσω παραθυρικού περιβάλλοντος για την διεξαγωγή και την διαχείριση κλήσεων. Υλοποιεί έναν SIP USER AGENT καθώς και τις απαραίτητες λειτουργίες για την σύλληψη, αναπαραγωγή και λήψη/μετάδοση του ήχου και της εικόνας. Επίσης παρέχει εργαλεία για την παρακολούθηση των Sip μηνυμάτων ,την διαχείριση των λογαριασμών του χρήστη καθώς και ελέγχου των συσκευών σύλληψης και απεικόνισης.

6.2.1 CONFIGURATION

Η παραμετροποίηση της εφαρμογής γίνεται μέσω δύο αρχείων των uaconfig.txt και SipStack_cl.txt καθώς και ένα αρχείο contacts.txt το οποίο περιέχει πληροφορίες καταλόγου(τις SIP και IP διευθύνσεις φίλων του χρήστη). Ένα παράδειγμα contacts.txt αρχείου φαίνεται στην σχήμα 6.1.

```
sofi=192.168.10.2:5080
maria=maria@ceid.teipat.gr
dimitris=192.168.10.2:5090
gentian=gentian@pc11.inet.com
popi=192.156.12.45:5050
&popi=C:\music\House 2008\CD 1\101_flat_mode_-_wonder_why.mp3
&bill=C:\billdiplom\Voip\UA_Files\ring.wav
```

Σχήμα 6.1: **Παράδειγμα αρχείου** contacts.txt

Το αρχείο uaconfig.txt περιέχει πληροφορίες για:

- Τους λογαριασμούς των χρηστών
- Τον χρόνο που θα επανεγγράφεται ο χρήστης στον Registrar
- Τις παραμέτρους του RTP Session.

Ένα παράδειγμα του uaconfig.txt αρχείου φαίνεται στοην σχήμα 6.2.

```
from_url=spiros@teipat.gr
from_url=maria@teipat.gr
from_url=gentian@upatras.gr
from_url=ok
contact_url=gentian
username=g.gentian
realm=gsgshshs
renew_time=15
contacts_file=UAconfig/contacts.txt
remote_address=192.168.2.3;
local_audio_port=22222
local_video_port=44444
remote_audio_port=22266
remote_video_port=44466
audio_codec=LINEAR
audio_encoding=G723_RTP
video_codec=RGB
video_encoding=H263_RTP
default_tone=C:\gentiandiplom\Voip\UA_Files\ring.WAV
```

Σχήμα 6.2: **Παράδειγμα ενός** uaconfig.txt **αρχείου**.

Το αρχείο SipStack_cl.txt περιέχει τις απαραίτητες πληροφορίες για το SIP πρωτόκολλο όπως τις θύρες, τα υποστηριζόμενα πρωτόκολλα(udp,tcp),τον μέγιστο αριθμό προώθησης των SIP μηνυμάτων κ.α.

Ένα παράδειγμα του αρχείου SipStack_cl.txt φαίνεται παρακάτω(Σχήμα 6.3).

```
default_port=5080
default_transport_protocols=udp
default_nmax_connections=45
use_rport=yes
force_rport=no
max_forwards=100
retransmission_timeout=2000
max_retransmission_timeout=4000
transaction_timeout=10000
```

```
clearing_timeout=5000
single_timer=yes
early_dialog=yes
default_expires=1000
ua_info=bill_graph_user_agent
server_info=bill_server
debug_level=5
log_path= ./log
max_logsize=4096
og_rotations=4
log_rotation_time=7 DAYS
host_port=5060
host_ifaddr=192.168.10.10
transport_protocols=udp
nmax_connections=77
outbound_proxy=proxy.wonderland.net:5022
ua_info=gentian_client_info
server_info=gentian_server_info
```

Σχήμα 6.3: **Παράδειγμα ενός** SipStack_cl.txt **αρχείου**.

6.2.2 ΕΚΚΙΝΗΣΗ ΕΦΑΡΜΟΓΗΣ

Για την εκκίνηση της εφαρμογής σε περιβάλλον Windows απλά τρέχουμε το αρχείο client.bat.Για την σωστή εκκίνηση του client πρέπει να έχουμε εγκαταστήσει τον JRE στο C:\Program Files\Java\jre6 αλλιώς θα χρειαστεί να επεξεργαστούμε την πρώτη γραμμή του .bat και να θέσουμε το μονοπάτι που είναι αποθηκευμένο το JRE όπως παρακάτω:

Από **à** set path=%path%;C:\Program Files\Java\jre6\bin

Σε **à** set path=%path%;<<yours jre bin path>>

Επίσης μπορούμε να τροποποιήσουμε κατευθείαν την μεταβλητή περιβάλοντος path μέσω Windows εκτελώντας τα παρακάτω βήματα:

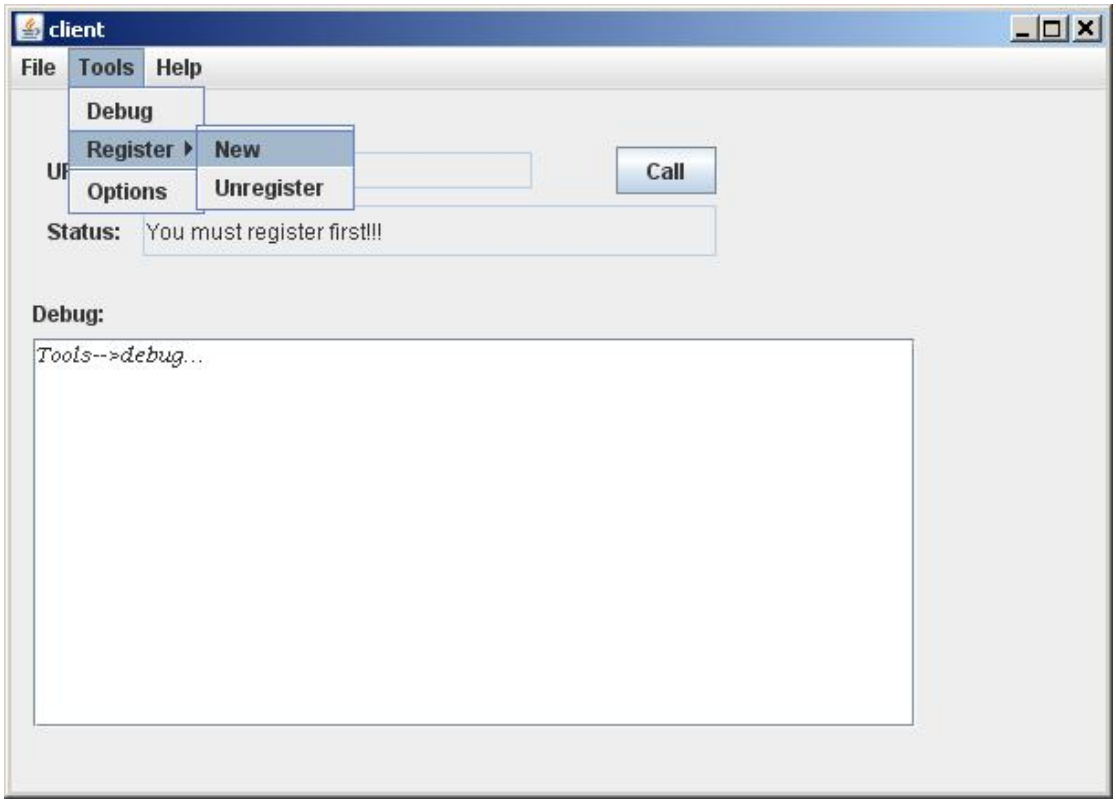
1. δεξί κλικ **à** ο υπολογιστής μου
2. αριστερό κλικ **à** ιδιότητες
3. αριστερό κλικ **à** Για προχωρημένους
4. αριστερό κλικ **à** μεταβλητές περιβάλλοντος
5. επιλέγουμε την μεταβλητή συστήματος Path
6. αριστερό κλικ **à** επεξεργασία
7. και εισάγουμε στο τέλος ένα «;» και έπειτα το πλήρες μονοπάτι του jre\bin
8. διπλό κλικ στο αρχείο client.bat.

Όλο το project έχει δημιουργηθεί με την βοήθεια του περιβάλλοντος NetBeans IDE 6.1 .Οπότε αν ενδιαφέρεστε για επέκταση της εφαρμογής ή για έλεγχο της εφαρμογής με χρήση στατιστικών εργαλείων αρκεί να κάνεται εισαγωγή όλο τον φάκελο Voip ως project στο περιβάλλον NetBeans και να κάνετε αλλαγές στο πακέτο client.

6.2.3 ΛΕΙΤΟΥΡΓΙΕΣ

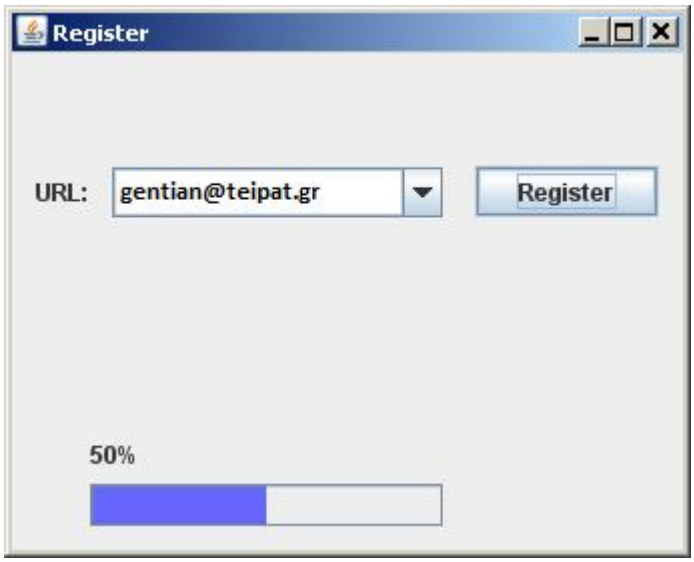
Παρακάτω αριθμούνται οι λειτουργίες της εφαρμογής του πελάτη.

1.**ΕΓΓΡΑΦΗ ΧΡΗΣΤΗ ΜΕ ΤΟΝ** REGISTRAR.



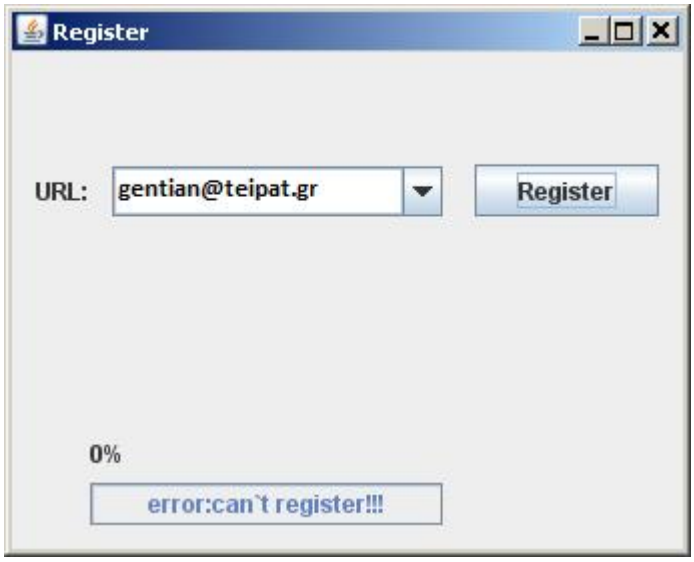
Εικόνα 6.1: **ΕΓΓΡΑΦΗ -Βήμα** 1.

Εικόνα 6.2: **ΕΓΓΡΑΦΗ -Βήμα** 2.



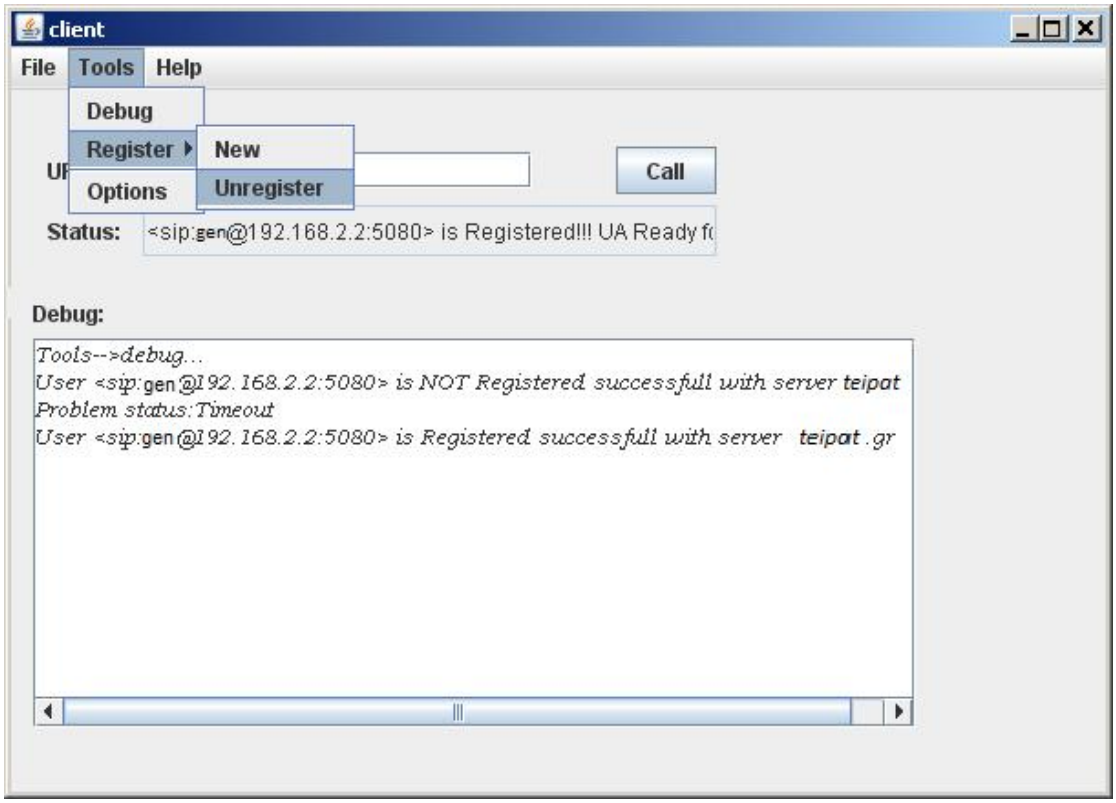
Εικόνα 6.3: **ΕΓΓΡΑΦΗ -Βήμα** 3.

Εικόνα 6.4: **ΕΓΓΡΑΦΗ - Επιτυχής Εγγραφή.**

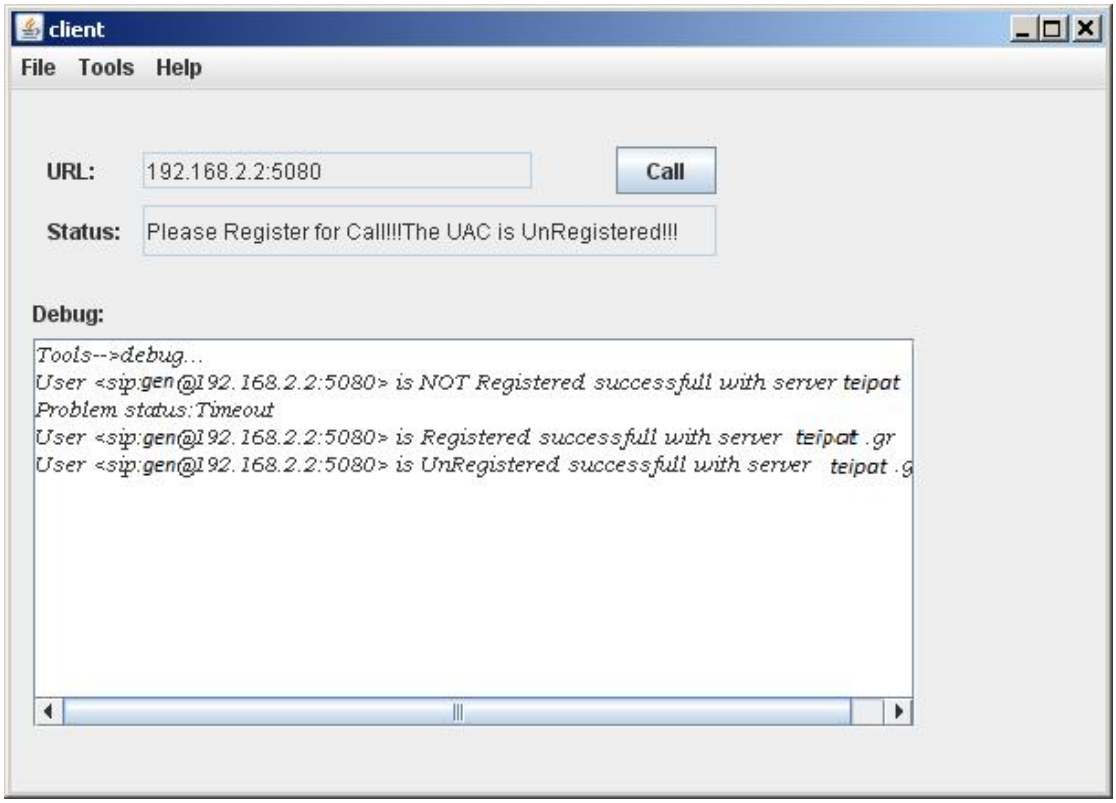


Εικόνα 6.5: **ΕΓΓΡΑΦΗ -Ανεπιτυχής Εγγραφή.**

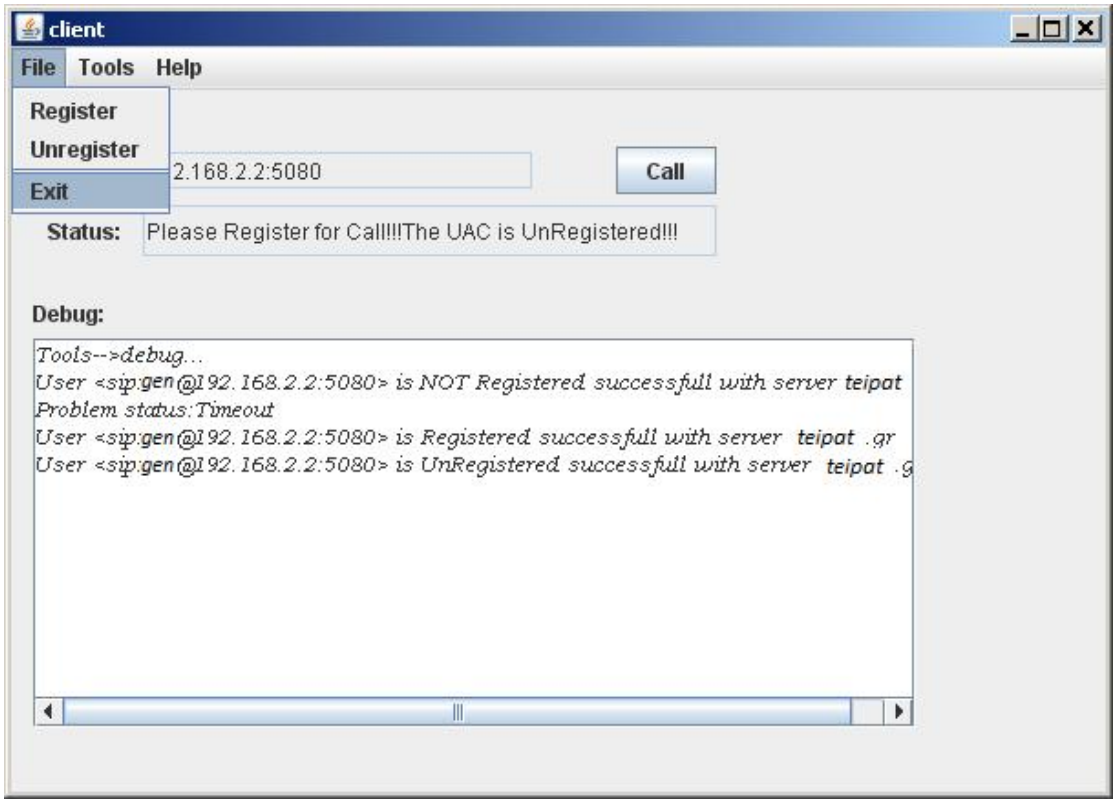
2. **ΑΠΕΓΓΡΑΦΗ ΑΠΟ ΤΟΝ** REGISTRAR



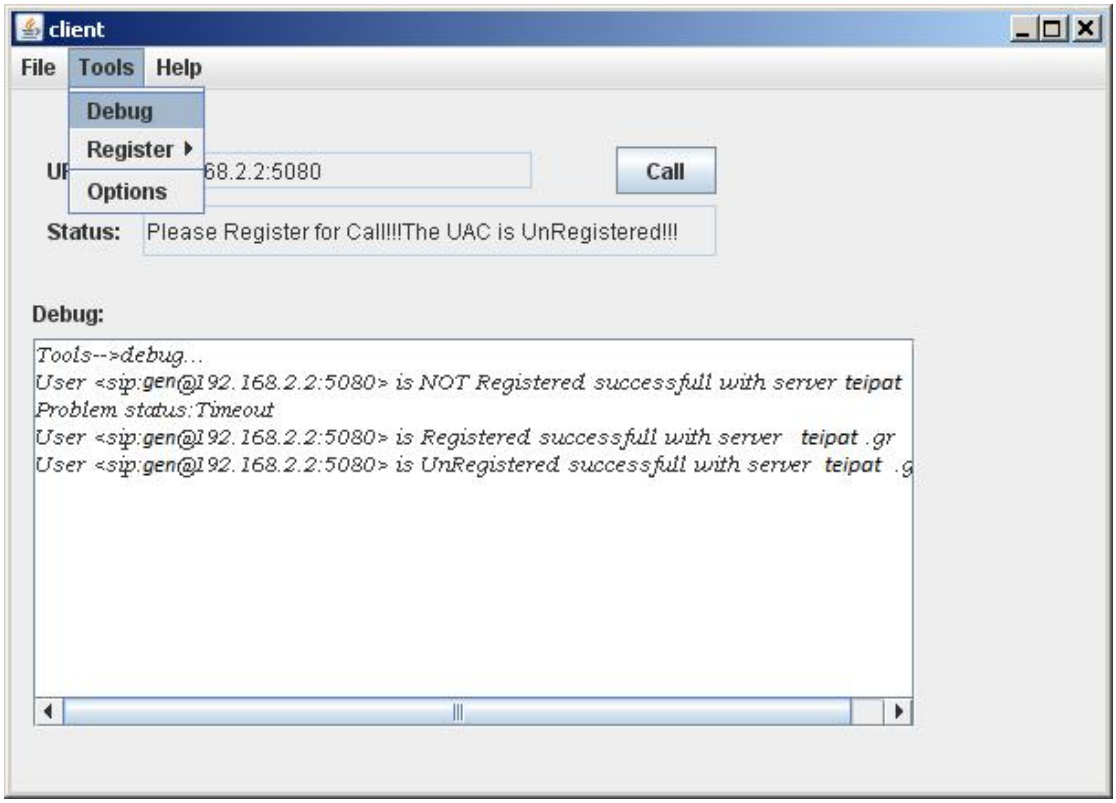
Εικόνα 6.6: **ΑΠΕΓΓΡΑΦΗ -Βήμα** 1.



Εικόνα 6.7: **ΑΠΕΓΓΡΑΦΗ -Βήμα** 2.

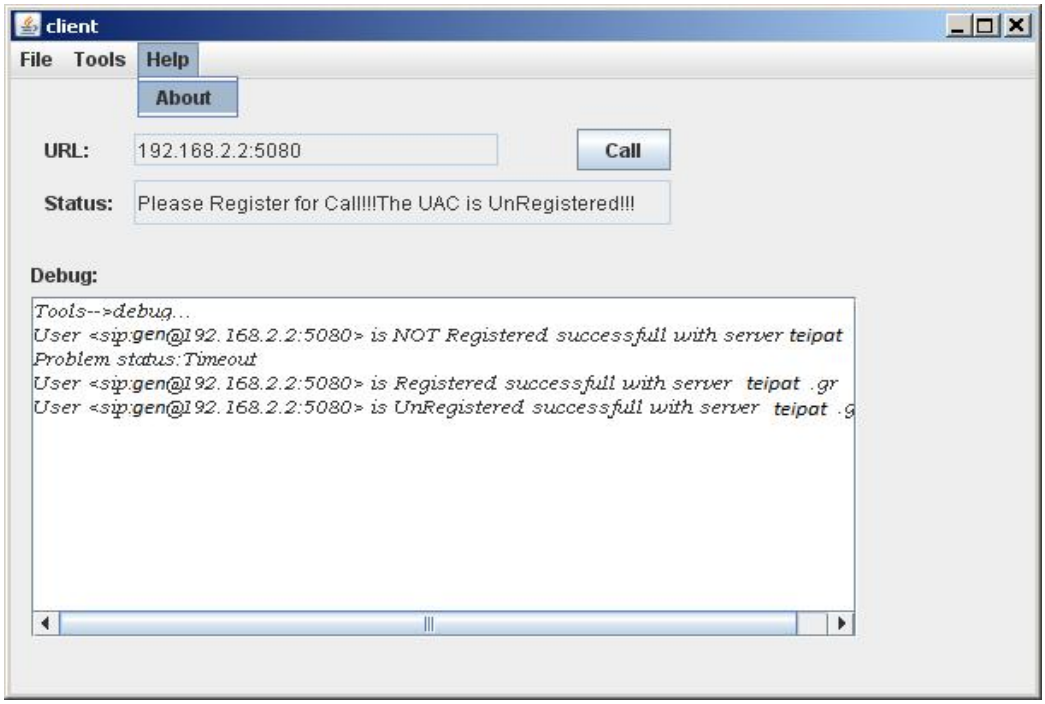


Εικόνα 6.8: **Έξοδος**.

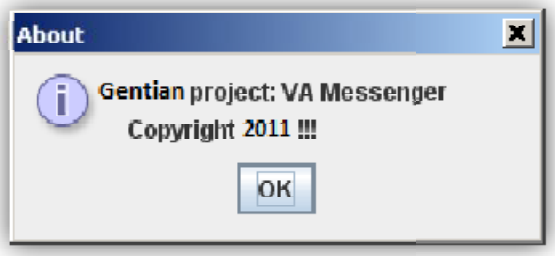


Εικόνα 6.9: **Προβολή λεπτομερειών**-Debug.

5. **ΠΡΟΒΟΛΗ ΠΛΗΡΟΦΟΡΙΩΝ** CLIENT

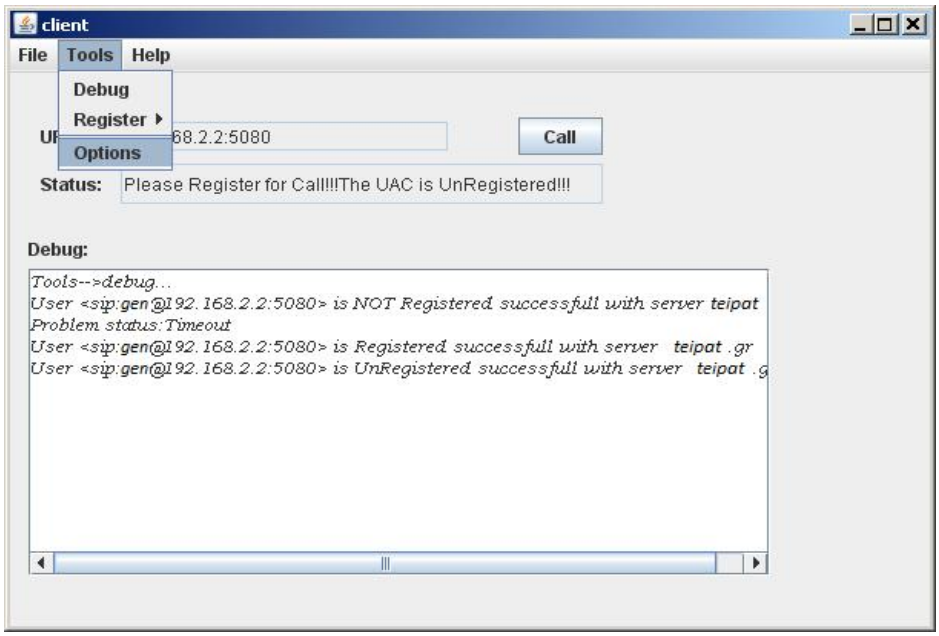


Εικόνα 6.10: **Προβολή πληροφοριών** Client.

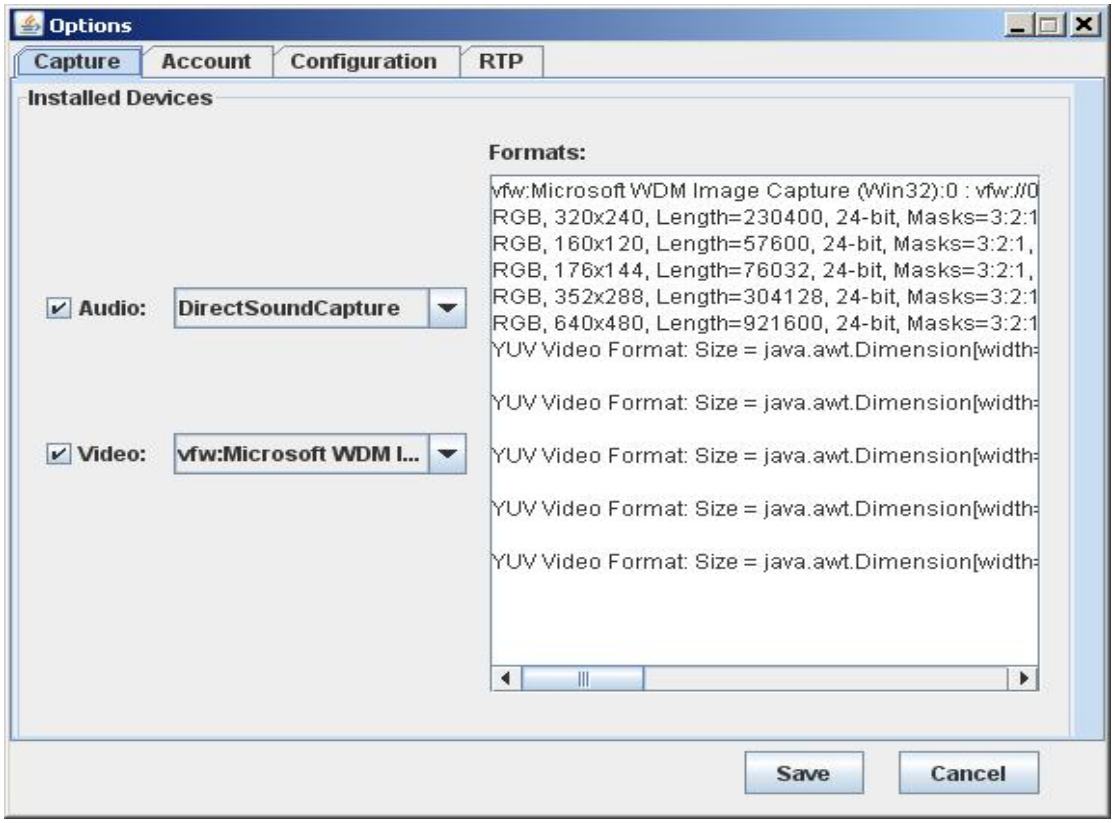


Εικόνα 6.11: **Προβολή.**

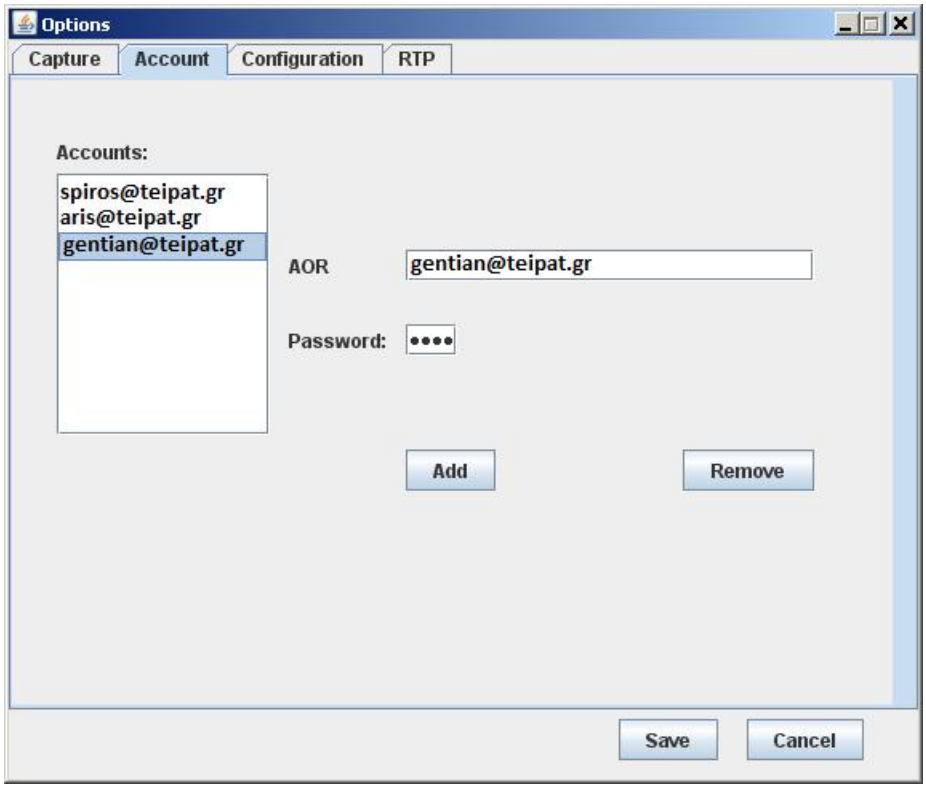
6. **ΕΙΣΟΔΟΣ ΣΤΟ ΜΕΝΟΥ ΠΑΡΑΜΕΤΡΟΠΟΙΗΣΗΣ ΤΟΥ** CLIENT



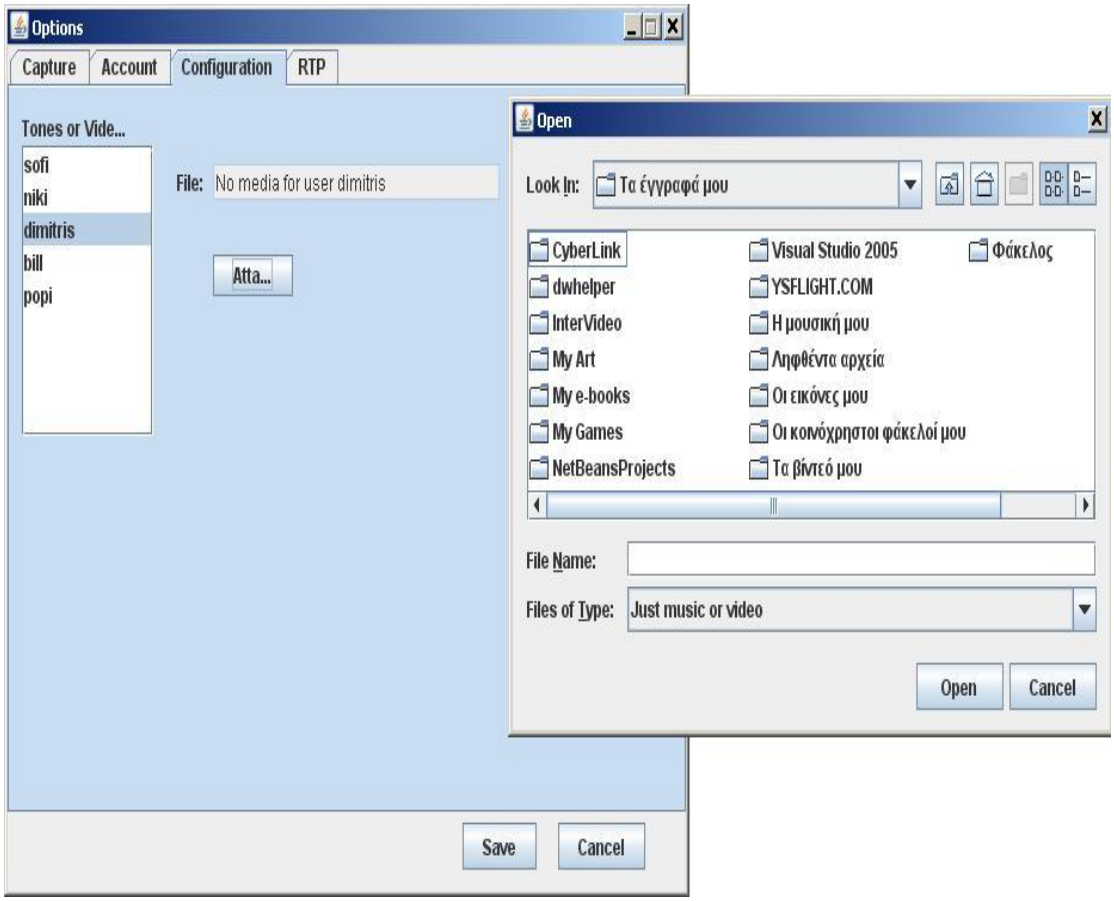
Εικόνα 6.12: **Είσοδος στις Επιλογές.**



Εικόνα 6.13: **Επιλογές πολυμεσικών συσκευών**.

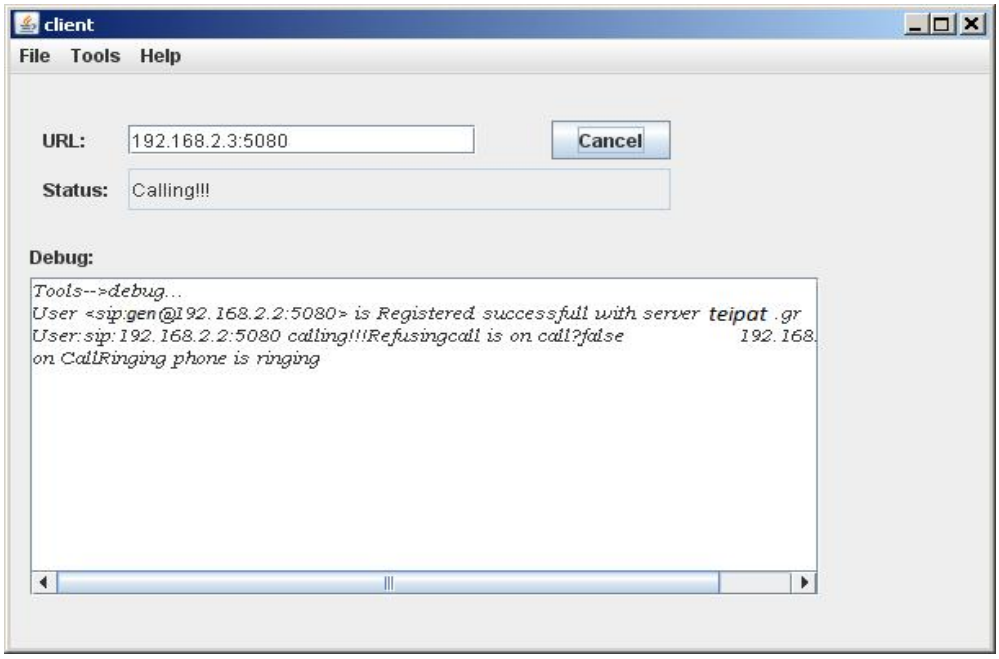


Εικόνα 6.14: **Διαχείρηση λογαριασμών**.

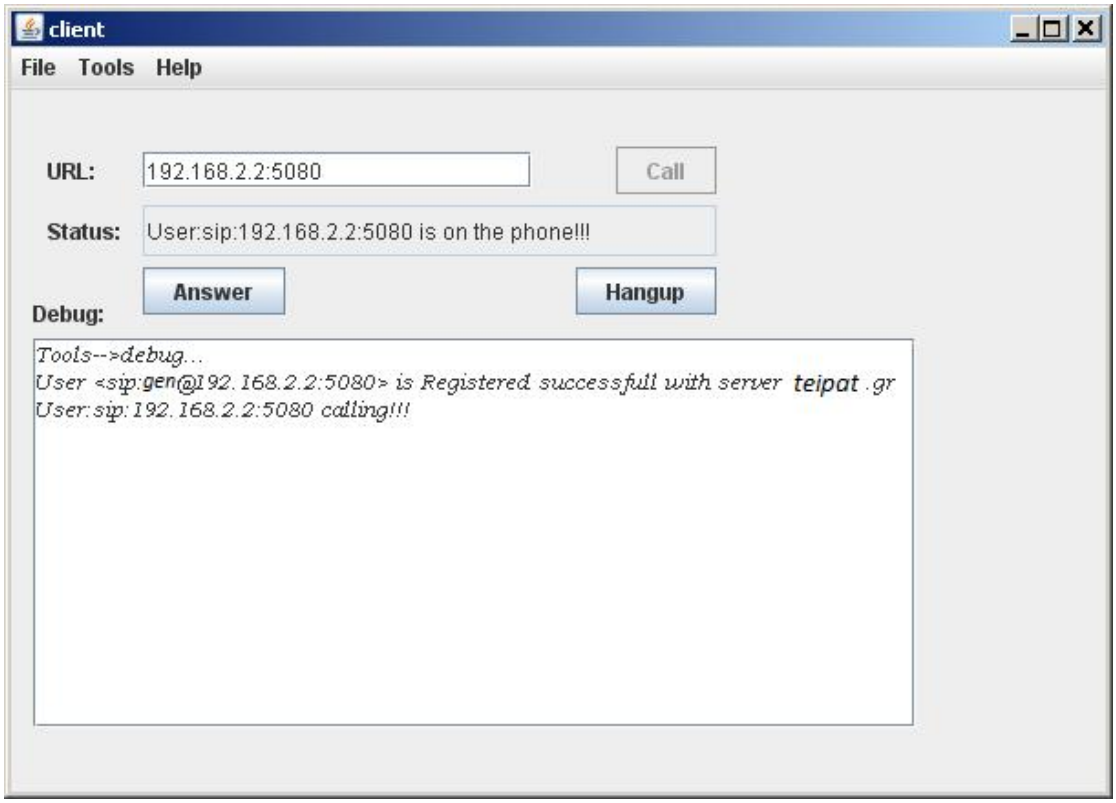


Εικόνα 6.15: **ΔΙΑΧΕΙΡΙΣΗ ΕΠΑΦΩΝ(ΣΥΝΔΕΣΗ ΗΧΟΥ ΚΛΗΣΗΣ ΑΝΑ ΕΠΑΦΗ)**

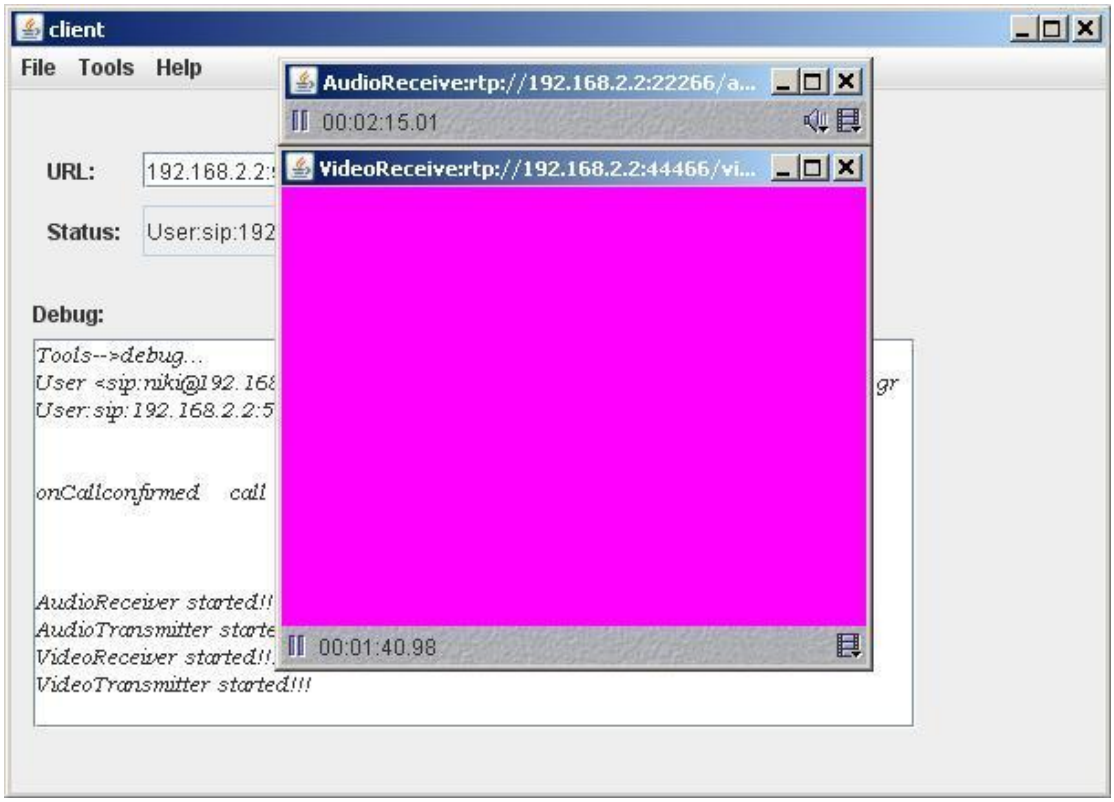
7. **ΚΛΗΣΗ**



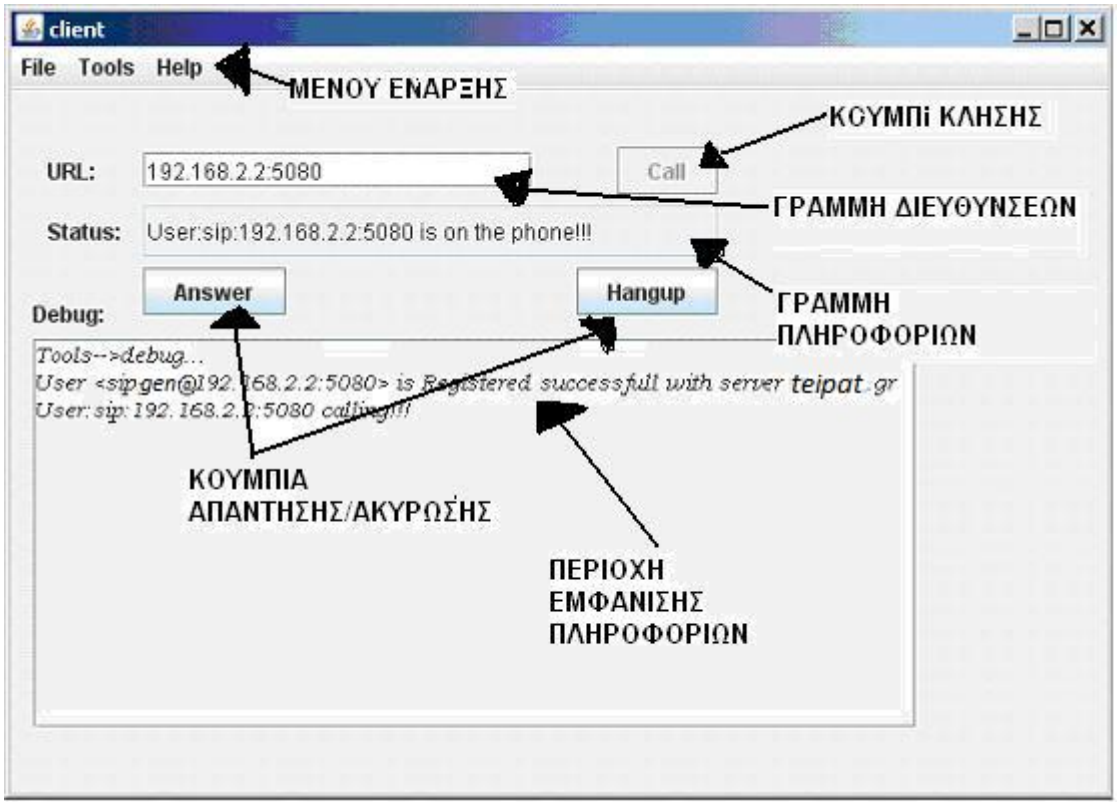
Εικόνα 6.16: **Κλήση Βήμα** 1.



Εικόνα 6.17: **Κλήση Βήμα** 2.



Εικόνα 6.18: **Κλήση Βήμα** 3.



Εικόνα 6.19: **ΓΕΝΙΚΗ ΠΕΡΙΓΡΑΦΗ ΑΡΧΙΚΟΥ ΠΑΡΑΘΥΡΟΥ**.

6.3 REGISTRAR SERVER

Αποτελεί την υλοποίηση ενός Registrar Server παρέχοντας όλες τις βασικές λειτουργίες ενός Registrar όπως περιγράφονται στο RFC 3261.Ώς εξυπηρετητής θέσεως είναι υπεύθυνος να γνωρίζει την θέση των χρηστών που είναι συνδεδεμένοι στην επικράτεια την οποία καλύπτει.

6.3.1 CONFIGURATION

Η παραμετροποίηση του Registrar γίνεται μέσω του αρχείου SipStack_serv.txt. Στο Σχήμα 6.4 φαίνεται ένα παράδειγμα του αρχείου SipStack_serv.txt καθώς και η επεξήγηση της κάθε τιμής.

```
#sets the default port for sip
default_port=5060
#set an ip address or set AUTO for auto configuration for sip address
release=192.168.2.2
#set the default transport protocols that can used from sip(only udp,tcp)
```

```
default_transport_protocols=udp
#set the number of max connections that the application can hundle
default_nmax_connections=45
#set the maximum number of the message hops
max_forwards=100
#Starting retransmission timeout (milliseconds); called T1 in RFC2361; they suggest
T1=500ms
retransmission_timeout=2000
# Maximum retransmission timeout (milliseconds); called T2 in RFC2361; they suggest
T2=4sec
max_retransmission_timeout=4000
# Transaction timeout (milliseconds); RFC2361 suggests 64*T1=32000ms
transaction_timeout=10000
# Clearing timeout (milliseconds); T4 in RFC2361; they suggest T4=5sec
clearing_timeout=5000
# Whether 1xx responses create an "early dialog" for methods that create dialog.
early_dialog=yes
#set the default expires haeder of sip messages
default_expires=1800
#server info(only for Registrar)
server_info=bill_server
#Log level. Only logs with a level less or equal to this are written.
debug_level=5
#set the log path that the log files are written
log_path= ./log
# The domain name(s) that the server administers.
# It lists the domain names for which the Location Service maintains user bindings.
domain_names=upatras.gr
```

Σχήμα 6.4:**Παράδειγμα αρχείου** SipStack_serv.txt

6.3.2 ΕΚΚΙΝΗΣΗ ΕΦΑΡΜΟΓΗΣ

Για την εκκίνηση της εφαρμογής του Registrar υπάρχει ένα αντίστοιχο αρχείο όπως στον client ,το Registrar.bat. Τα βήματα που περιγράφηκαν στο κεφάλαιο 6.2.2 για την εγκατάσταση ή παραμετροποίηση του jre ισχύουν και για την εκκίνηση του Registrar.

Επιπλέον χρειάζεται να έχουμε εκκινήσει την υπηρεσία του MySQL Server για την επιτυχή σύνδεση του Registrar με την βάση δεδομένων.

6.3.3 ΛΕΙΤΟΥΡΓΙΕΣ

Η εφαρμογή του Registrar δεν υποστηρίζει παραθυρικό περιβάλλον διαθέτει όμως τα αρχεία “realm”_events.log(σχήμα 6.5) και “realm”_messages.log(σχήμα 6.6) στον φάκελο log όπου παρέχουν πληροφορίες σχετικά με τις καταστάσεις και τα μηνύματα που στέλνει /λαμβάνει ο Registrar αντίστοιχα. Η μεταβλητή realm αλλάζει αναλόγως των πραγματικών δικτυακών χαρακτηριστικών που τρέχει η εφαρμογή.Μερικά παραδείγματα παρουσιάζονται παρακάτω:

1. Είναι η τιμή της μεταβλητής host_ifaddr συνοδευόμενη από την τιμή default_port στο αρχείο SipStack_serv.txt από όπου καθορίζεται η διεύθυνση και η θύρα όπου χρησιμοποιεί η εφαρμογή.
2. Σε περίπτωση που η μεταβλητή host_ifaddr στο αρχείο SipStack_serv.txt έχει την τιμή AUTO τότε η διεύθυνση υπολογίζεται αυτόματα ανάλογα με τις διαθέσιμες συνδέσεις του συστήματος όπου τρέχει ο Registrarκαι έτσι το όνομα του αρχείου αλλάζει ως:

   <διεύθυνση>>.$default_port_(events/messages).log

SipProvider-5060: Date: 22:50:57.453 Thu 05 Mar 2009
SipProvider-5060: SipStack: mjsip stack 1.6
SipProvider-5060: new SipProvider(): 5060/udp
SipProvider-5060: udp is up
SipProvider-5060: adding SipProviderListener: ANY

22:51:00.109 Thu 05 Mar 2009, 192.168.2.2:5090/udp (351 bytes): REGISTER sip:192.168.2.2:5060 SIP/2.0, received

SipProvider-5060: received new SIP message
SipProvider-5060: DEBUG: transaction-id: 978883221412@192.168.2.2-1-REGISTER-192.168.2.2:5090-z9hG4bK88318
SipProvider-5060: DEBUG: dialog-id: 978883221412@192.168.2.2-null-z9hG4bK71229152
SipProvider-5060: message passed to uas: ANY
Registrar: Register request received RegisterRequestFromUA() fired!!!Via: SIP/2.0/UDP 192.168.2.2:5090;branch=z9hG4bK88318;rport=5090

TransactionServer#0: id: 978883221412@192.168.2.2-1-REGISTER-192.168.2.2:5090-z9hG4bK88318
TransactionServer#0: created
TransactionServer#0: changed transaction state: T_Trying
SipProvider-5060: adding SipProviderListener: 978883221412@192.168.2.2-1-REGISTER-192.168.2.2:5090-z9hG4bK88318
SipProvider-5060: Sending message through conn null
SipProvider-5060: no active connection found matching null
SipProvider-5060: using transport udp
SipProvider-5060: Resolving host address '192.168.2.2'
SipProvider-5060: Sending message to udp:192.168.2.2:5090

22:51:00.265 Thu 05 Mar 2009, 192.168.2.2:5090/udp (309 bytes): SIP/2.0 200 OK, sent

TransactionServer#0: changed transaction state: T_Completed
SipProvider-5060: removing SipProviderListener: 978883221412@192.168.2.2-1-REGISTER-192.168.2.2:5090-z9hG4bK88318
TransactionServer#0: changed transaction state: T_Terminated

22:51:06.234 Thu 05 Mar 2009, 192.168.2.2:5080/udp (351 bytes): REGISTER sip:192.168.2.2:5060 SIP/2.0, received

SipProvider-5060: received new SIP message
SipProvider-5060: DEBUG: transaction-id: 293512324943@192.168.2.2-1-REGISTER-192.168.2.2:5080-z9hG4bK16334
SipProvider-5060: DEBUG: dialog-id: 293512324943@192.168.2.2-null-z9hG4bK95275680
SipProvider-5060: message passed to uas: ANY
Registrar: Register request received RegisterRequestFromUA() fired!!!Via: SIP/2.0/UDP 192.168.2.2:5080;branch=z9hG4bK16334;rport=5080

TransactionServer#1: id: 293512324943@192.168.2.2-1-REGISTER-192.168.2.2:5080-z9hG4bK16334
TransactionServer#1: created
TransactionServer#1: changed transaction state: T_Trying
SipProvider-5060: adding SipProviderListener: 293512324943@192.168.2.2-1-REGISTER-192.168.2.2:5080-z9hG4bK16334
SipProvider-5060: Sending message through conn null
SipProvider-5060: no active connection found matching null
SipProvider-5060: using transport udp
SipProvider-5060: Resolving host address '192.168.2.2'
SipProvider-5060: Sending message to udp:192.168.2.2:5080

22:51:06.265 Thu 05 Mar 2009, 192.168.2.2:5080/udp (309 bytes): SIP/2.0 200 OK, sent

Σχήμα 6.5: **παράδειγμα αρχείου** 192.168.2.2.5060_events.log

22:51:00.109 Thu 05 Mar 2009, 192.168.2.2:5090/udp (351 bytes): received
REGISTER sip:192.168.2.2:5060 SIP/2.0
Via: SIP/2.0/UDP 192.168.2.2:5090;rport;branch=z9hG4bK88318
Max-Forwards: 70
To: <sip:aris@teipat.gr>
From: <sip:aris@teipat.gr>;tag=z9hG4bK71229152
Call-ID: 978883221412@192.168.2.2
CSeq: 1 REGISTER
Contact: <sip:aris@192.168.2.2:5090>
Expires: 1000

User-Agent: bill_client_info
Content-Length: 0

-----End-of-message-----

22:51:00.265 Thu 05 Mar 2009, 192.168.2.2:5090/udp (309 bytes): sent
SIP/2.0 200 OK
Via: SIP/2.0/UDP 192.168.2.2:5090;branch=z9hG4bK88318;rport=5090
To: <sip:aris@teipat.gr>
From: <sip:aris@teipat.gr>;tag=z9hG4bK71229152
Call-ID: 978883221412@192.168.2.2
CSeq: 1 REGISTER
Server: gentian_server_info
Contact: <sip:aris@192.168.2.2:5090>;expires=1000
Content-Length: 0

-----End-of-message-----

22:51:06.234 Thu 05 Mar 2009, 192.168.2.2:5080/udp (351 bytes): received
REGISTER sip:192.168.2.2:5060 SIP/2.0
Via: SIP/2.0/UDP 192.168.2.2:5080;rport;branch=z9hG4bK16334
Max-Forwards: 70
To: <sip: gentian@teipat.gr >
From: <sip: gentian@teipat.gr >;tag=z9hG4bK95275680
Call-ID: 293512324943@192.168.2.2
CSeq: 1 REGISTER
Contact: <sip:gentian@192.168.2.2:5080>
Expires: 1000
User-Agent: gentian_client_info
Content-Length: 0

-----End-of-message-----

22:51:06.265 Thu 05 Mar 2009, 192.168.2.2:5080/udp (309 bytes): sent
SIP/2.0 200 OK
Via: SIP/2.0/UDP 192.168.2.2:5080;branch=z9hG4bK16334;rport=5080
To: <sip: gentian@teipat.gr >
From: <sip: gentian@teipat.gr >;tag=z9hG4bK95275680
Call-ID: 293512324943@192.168.2.2
CSeq: 1 REGISTER
Server: gentian_server_info
Contact: <sip:gentian@192.168.2.2:5080>;expires=1000
Content-Length: 0

-----End-of-message-----

Σχήμα 6.6: **παράδειγμα αρχείου** 192.168.2.2.5060_messages.log

6.4 LOCATION SERVICE

Αποτελεί την βάση δεδομένων όπου αποθηκεύονται οι θέσεις των χρηστών που είναι συνδεδεμένοι(binds).Έπιτρέπει την σύνδεση πολλών χρηστών ταυτόχρονα ,είτε τοπικά ,είτε απομακρυσμένα και χρήση διαφορετικών λογαριασμών σύνδεσης με διαφορετικά δικαιώματα.Αποτελεί μια σχεσιακή βάση δεδομένων όπου χρησιμοποιείται η InnoDB μηχανή αποθήκευσης του MySQL Server καθώς και η SQL γλώσσα για την διαχείρηση της βάσης.

Στα επόμενα κεφάλαια παρουσιάζονται τα βήματα της εγκατάστασης της βάσης δεδομένων σε έναν MySQL Server καθώς και το UML διάγραμμα περιγραφής της βάσης δεδομένων.

6.4.1 ΕΓΚΑΤΑΣΤΑΣΗ

Η βάση δεδομένων έχει εξαχθεί στο αρχείο LocatoinService.sql. Για να μπορέσουμε να την εισάγουμε στον τοπικό μας Server θα πρέπει πρώτα να δημιουργήσουμε μια βάση δεδομένων (database) με το όνομα locationservice και στην συνέχεια έναν λογαριασμό που να μπορεί να υλοποιεί όλες τις πράξεις πάνω στην βάση δεδομένων μας με τα χαρακτηριστικά:

- Username **à** Registrar
- Password **à** sip

Για την δημιουργία του λογαριασμού εκτελούμε τις παρκάτω εντολές:

1. create user 'Registrar' identified by 'sip' (αυτή η εντολή δημιουργεί τον χρήστη).
2. grant all on locationservice.* to 'Registrar' (ενεργοποιούμε όλα τα δικαιώματα για την βάση δεδομένων locationservice).

Εάν ο Registrar Server βρίσκεται στον ίδιο υπολογιστή με τον MySQL Server είναι προτιμότερο να ενεργοποιήσουμε τον λογαριασμό μόνο για τοπική σύνδεση με τον MySQL Server αλλάζοντας τις παραπάνω εντολές ως εξής:

1. create user 'Registrar'@'localhost' identified by 'sip' (αυτή η εντολή δημιουργεί τον χρήστη).
2. grant all on locationservice.* to 'Registrar'@'localhost' (ενεργοποιούμε όλα τα δικαιώματα για την βάση δεδομένων locationservice).

Μπορούμε να εισάγουμε την βάση δεδομένων μας από το αρχείο LocatoinService.sql με δύο τρόπους:

1. Χρησιμοποιώντας κάποιο ολοκληρωμένο εργαλείο της Mysql.
2. Χρησιμοποιώντας το command promt των Windows.

Για την δεύτερη περίπτωση πρέπει όπως ειπώθηκε πιο πάνω να κατασκευάσουμε πρώτα μια βάση δεδομένων με το όνομα locationservice.Αυτό γίνεται με την παρακάτω εντολή στο command line client του Mysql server:

- Create database locationservice;

Για έλεγχο μπορούμε να εκτελέσουμε την εντολή:

- show databases;

Άν υπάρχει η βάση δεδομένων με το όνομα locationservice στη λίστα τότε η βάση έχει δημιουργηθεί.

Στην συνέχεια ανοίγουμε το command promt των Windows(start **à** run **à** cmd) και εκτελούμε την παρακάτω εντολή για να εισάγουμε την βάση δεδομένων στον Mysql Server:

- mysql –u $username -p$password locationservice < $path\locationservice.sql

όπου:

1. $username **à** το ψευδώνυμο του διαχειριστή που έχεται ορίσει στον στον MySQL Server.
2. $password **à** ο κωδικός του διαχειριστή που έχεται ορίσει στον στον MySQL Server(πρέπει ανάμεσα στον κωδικό και την παράμετρο –p να μην υπάρχει κενό).
3. $path **à** η διαδρομή του δίσκου στο αρχείο locationservice.sql .

Όταν ολοκληρωθεί η παραπάνω διαδικασία θα έχει εγκατασταθεί το μοντέλο της βάσης δεδομένων του LocationService μαζί με κάποια αρχικά στιγμιότυπα.

Η παραπάνω διαδικασία μπορεί να εκτελεσθεί και απομακρυσμένα εάν στις παραπάνω εντολές προσθέσουμε την παράμετρο –h με την διεύθυνση του υπολογιστή που είναι εγκατεστημένος ο Mysql server.(Περισσότερες πληροφορίες στο [3])

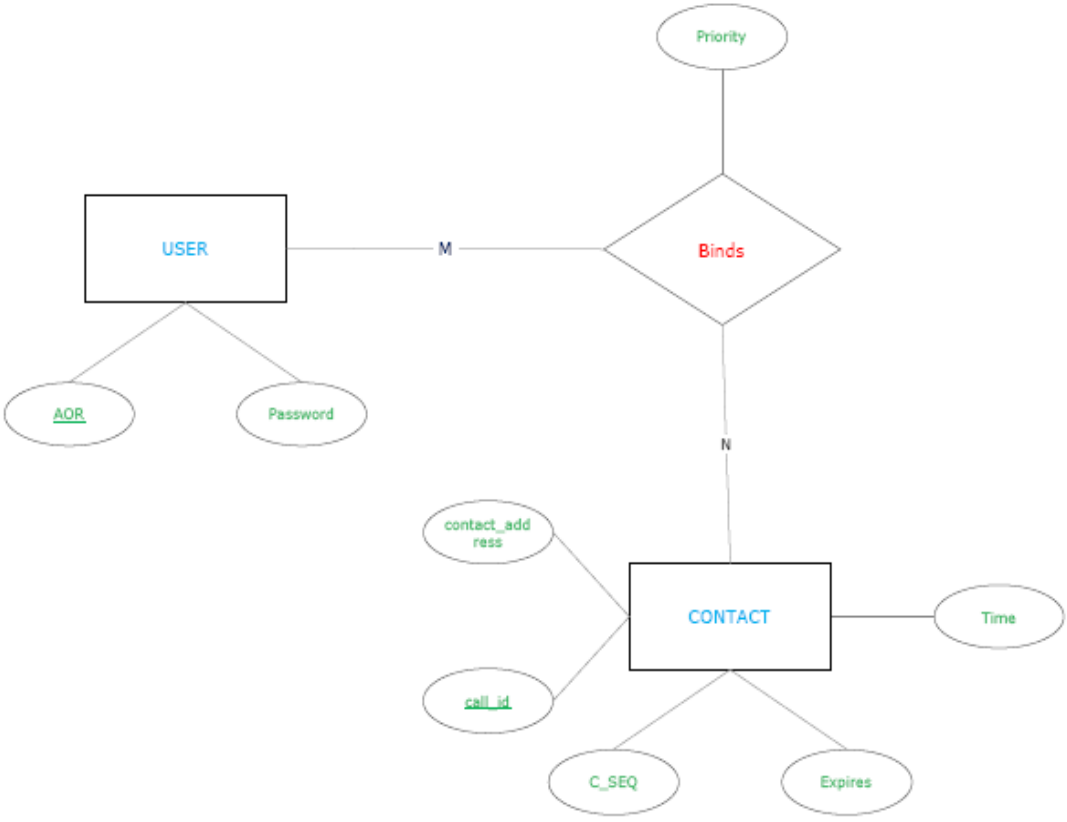
6.4.2 ΕΚΚΙΝΗΣΗ

Για την εκκίνηση της εφαρμογής χρειάζεται μόνο να εκκινήσουμε τον Mysql server. Στα Windows αυτό μπορεί να γίνει χειροκίνητα ως εξής:

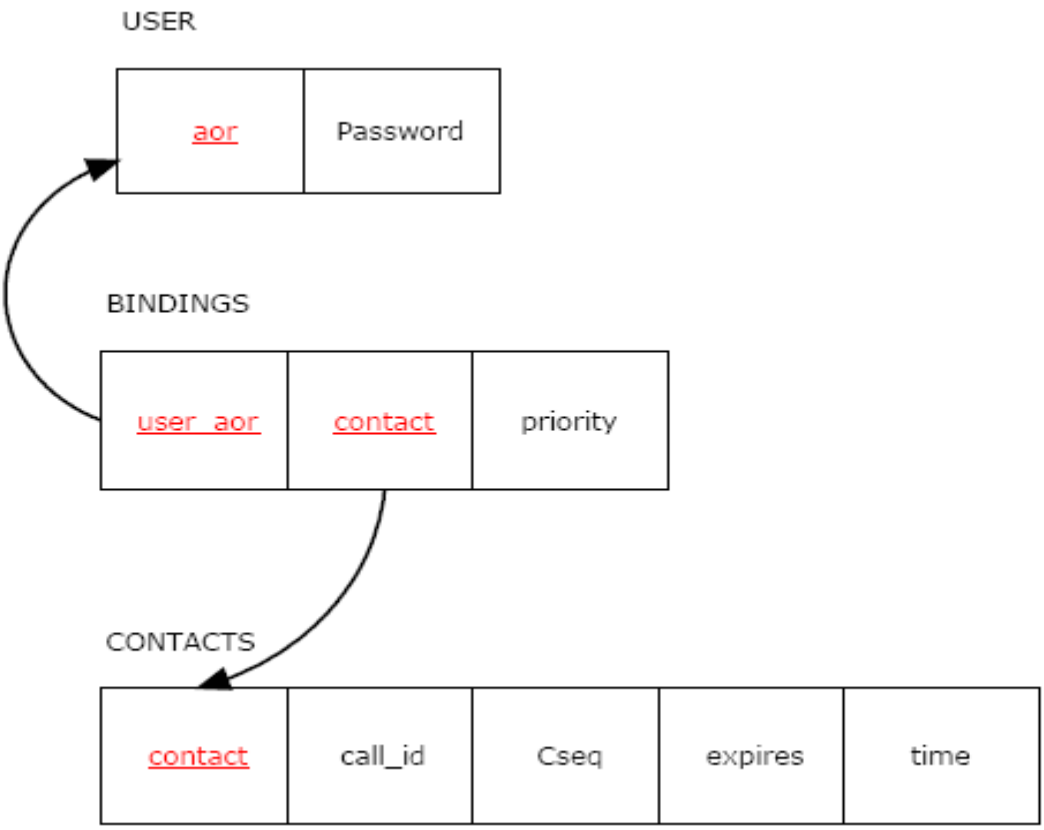
1. Πρέπει να ανοίξουμε το μενού με τις υπηρεσίες των Windows(Start **à** Run **à** services.msc)
2. Έπειτα επιλέγουμε την υπηρεσία MySQL για επεξεργασία και την εκκινούμε.

6.4.3 ΠΕΡΙΓΡΑΦΗ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ

Η βάση δεδομένων LocationService αποτελεί μια σχεσιακή βάση δεδομένων και αποτελείται από τρείς πίνακες.Στο σχήμα 6.7 φαίνεται το ER διάγραμμα (διάγραμμα Οντοτήτων-Συσχετίσεων) και στο σχήμα 6.8 το αντίστοιχο σχήμα της βάσης δεδομένων.



Σχήμα 6.7: **Διάγραμμα** ER **της βάσης δεδομένων**.



Σχήμα 6.8: **Διάγραμμα σχήματος για το σχήμα της σχεσιακής βάσης δεδομένων** LocationService.

ΚΕΦΑΛΑΙΟ 7

ΠΕΡΙΓΡΑΦΗ ΤΩΝ ΚΛΑΣΕΩΝ

ΚΕΦΑΛΑΙΟ 7: ΠΕΡΙΓΡΑΦΗ ΤΩΝ ΚΛΑΣΕΩΝ

Στο παρόν κεφάλαιο γίνεται μια σύντομη περιγραφή των πακέτων και των κλάσεων για την καλύτερη κατανόηση της δόμησης της εφαρμογής.Επίσης δίδεται και μια σχηματική αναπαράσταση με χρήση UML διαγραμμάτων.

7.1 PACKAGE CLIENT

Το πακέτο Client περιέχει όλες τις κλάσεις όπου συνθέτουν την εφαρμογή του sip client. Παρακάτω δίδεται μια περιγραφή των κλάσεων που συνθέτουν το πακέτο Client και στο σχήμα 7.1 φαίνεται το διάγραμμα κλάσεων.

Package Client

class AudioTransmit implements javax.media.ControllerListener: Η κλάση AudioTransmit επιλέγει την συσκευή σύλληψης ήχου και αποστέλει τα δεδομένα του ήχου μέσω του πρωτοκόλλου rtp στον συνομιλητή.Χρησιμοποιείται από την κλάση GUIClient για την αποστολή των δεδομένων ήχου στην σύνοδο πολυμέσων.

class ChooserFilter extends javax.swing.filechooser.FileFilter: Αποτελεί μια προέκταση της κλάσης FileFileter για το φιλτράρισμα των αρχείων πολυμέσων που είναι συμβατά με την βιβλιοθήκη JMF για την επιλογή ,μέσω του παραθύρου FileChooser ,των πολυμέσων που θα αντιστοιχίζονται σαν ήχος / βίντεο κλήσης ανά χρήστη στην κλάση OptionFrame.

class ClientPlayer implements javax.media.ControllerListener: Υλοποιεί το μέσο αναπαραγωγής των πολυμέσων του client.

class GUIClient extends javax.swing.JFrame implements GuiInterfaceListener,CallListener: Είναι η κύρια κλάση του client από την οποία αρχικοποιείται όλος ο client ,υλοποιείται το αρχικό παράθυρο και υλοποιεί την διεπαφή του χρήστη.

interface GuiInterfaceListener: Αποτελεί την διεπαφή για την επικοινωνία των επιπέδων που ορίζουν οι κλάσεις GUIClient και RegisterFrame.

class JAudioReceive extends javax.swing.JFrame implements javax.media.ControllerListener, java.awt.event.WindowListener: Λαμβάνει και αναπαράγει τα δεδομένα του ήχου που στέλνονται από τον συνομιλητή μέσω της rtp συνόδου.

class JVideoReceive extends javax.swing.JFrame implements javax.media.ControllerListener, java.awt.event.WindowListener: Ότι και η κλάση JaudioReceive για τα δεδομένα εικόνας που στέλνονται από τον συνομιλητή.

class OptionFrame extends javax.swing.Jframe: Υλοποιεί ένα παραθυρικό περιβάλλον για μετατροπή των παραμέτρων του client από τον χρήστη.Δίδει την δινατότητα για επεξεργασία των συσκευών σύλληψης ήχου και εικόνας ,επεξεργασία των λογαριασμών του χρήστη και των επαφών του.

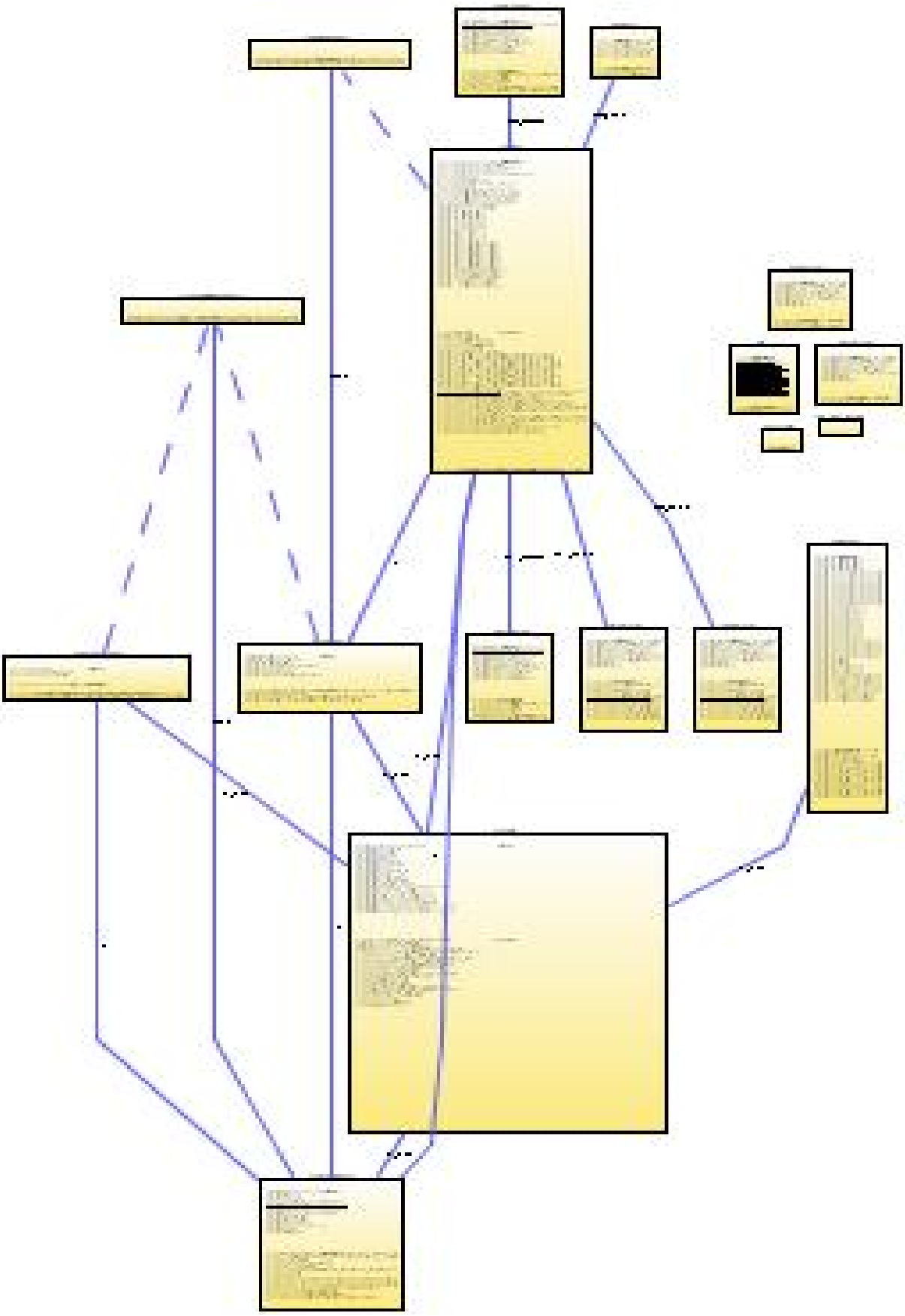
class RegisterFrame extends javax.swing.JFrame implements UserRegisterAgentListener: Υλοποιεί το παραθυρικό περιβάλλον για τον έλεγχο της διαδικασίας εγγραφής με τον Registrar του χρήστη.Ενημερώνει την κλάση GUIClient για την πορεία της εγγραφής.

class UaConfig extends Configure: Είναι η κλάση η οποία χρησιμοποιείται για την ανάγνωση-εγγραφή των αρχείων παραμετροποίησης(uaconfig.txt,contacts.txt).Όλες οι παράμετροι περνιούνται με την βοήθεια συναρτήσεων καθώς η εγγραφή και ανάγνωση των αρχείων γίνεται αποκλειστικά από την συγκεκριμένη κλάση για την αποφυγή συγκρούσεων.

class UserRegisterAgent implements Runnable,TransactionClientListener: Υλοποιεί την λειτουργία της εγγραφής με τον Registrar σε χαμήλο επίπεδο(επίπεδο Transactions) και ενημερώνει την κλάση RegisterFrame για τα στάδια της εγγραφής.

interface UserRegisterAgentListener: Αποτελεί την διεπαφή για την επικοινωνία των επιπέδων που ορίζουν οι κλάσεις RegisterFrame και UserRegisterAgent.

class VideoTransmit implements ControllerListener: Η κλάση AudioTransmit επιλέγει την συσκευή σύλληψης εικόνας και αποστέλει τα δεδομένα της εικόνας μέσω του πρωτοκόλλου rtp στον συνομιλητή.Χρησιμοποιείται από την κλάση GUIClient για την αποστολή των δεδομένων βίντεο στην σύνοδο πολυμέσων.



Σχήμα 7.1: **Διάγραμμα κλάσεων του** package Client.

7.2 PACKAGE SERVER

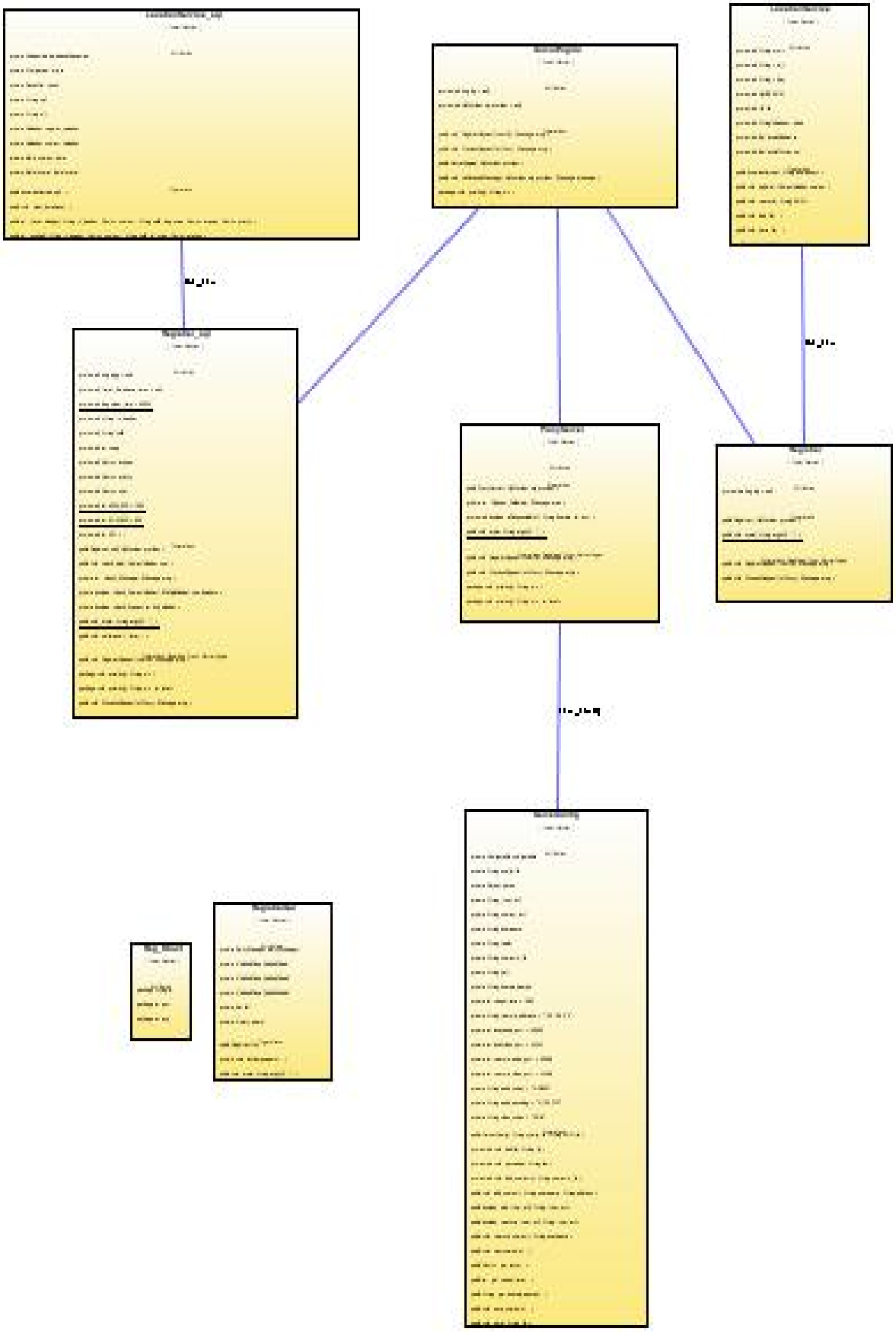
Το πακέτο Server περιέχει όλες τις κλάσεις όπου συνθέτουν την εφαρμογή του sip Registrar. Παρακάτω δίδεται μια περιγραφή των κλάσεων που συνθέτουν το πακέτο Server και στο σχήμα7.2 φαίνεται το διάγραμμα κλάσεων.

Package Server:

class LocationService_sql: Καλύπτει την επικοινωνία του Registrar με την βάση δεδομένων(LocationService). Ειδικότερα παρέχει τις sql πράξεις εισαγωγής / διαγραφής / επιλογής στην κλάση Registrar_sql για την δημιουργία / διαγραφή / επεξεργασία των εγγραφών(binds).

class Registrar_sql extends ServerEngine implements TimerListener: Επεκτείνει την κλάση ServerEngine και υλοποιεί τις βασικές λειτουργίες ενός Registrar Server όπως ορίζεται στο RFC 3261.

class ServerEngine implements SipProviderListener: Η κλάση ServerEngine αποτελεί μια αφηρημένη κλάση(abstract) καλύπτοντας το επίπεδο μεταφοράς του εξυπηρετητεί με την αποστολή και λήψη των SIP μηνυμάτων και την προώθηση των μηνυμάτων στην κλάση Registrar_sql(έλεγχος αν πρόκειται για Register μηνύματα).



Σχήμα 7.2: **Διάγραμμα κλάσεων του** package Server.

ΚΕΦΑΛΑΙΟ 8

ΠΑΡΑΔΕΙΓΜΑΤΑ ΕΚΤΕΛΕΣΗΣ

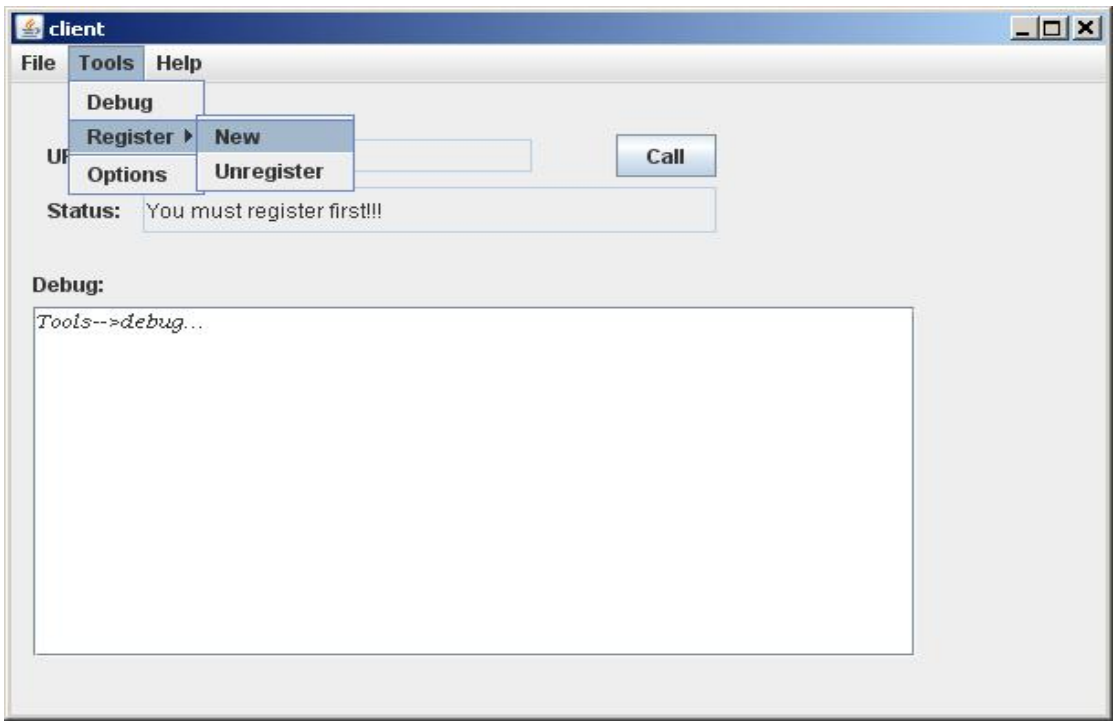
ΚΕΦΑΛΑΙΟ 8: ΠΑΡΑΔΕΙΓΜΑΤΑ ΕΚΤΕΛΕΣΗΣ

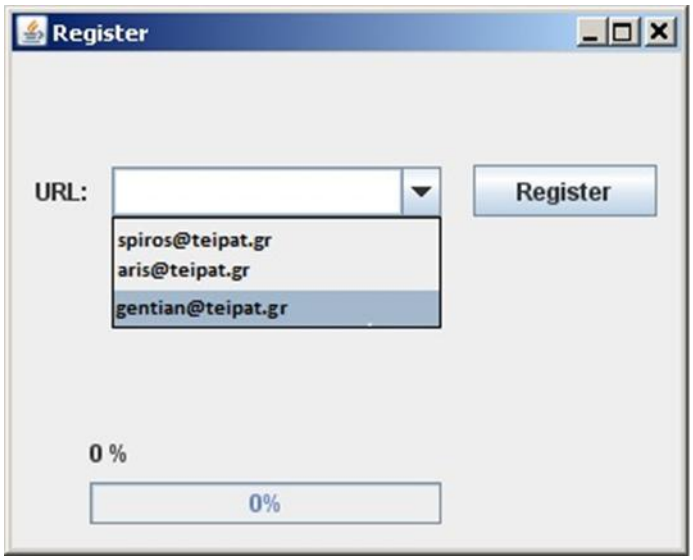
Στο κεφάλαιο αυτό δίνονται δύο βασικά παραδείγματα εκτέλεσης των δύο εφαρμογών Registrar και Client.

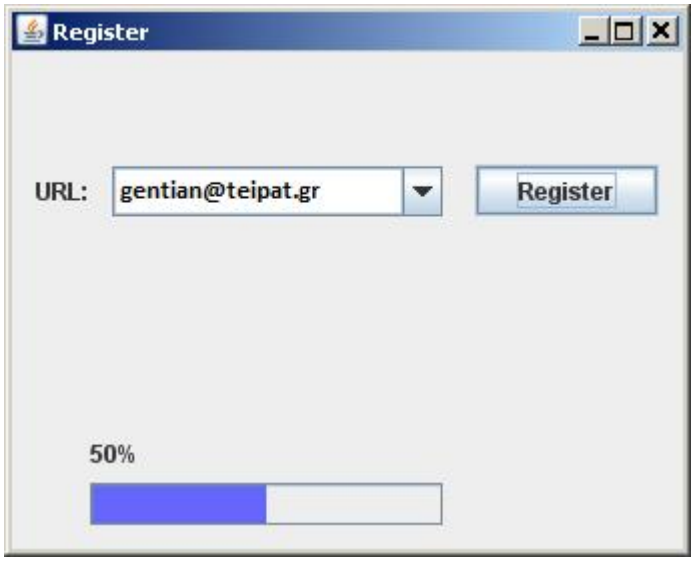
8.1 ΕΓΓΡΑΦΗ(REGISTER)

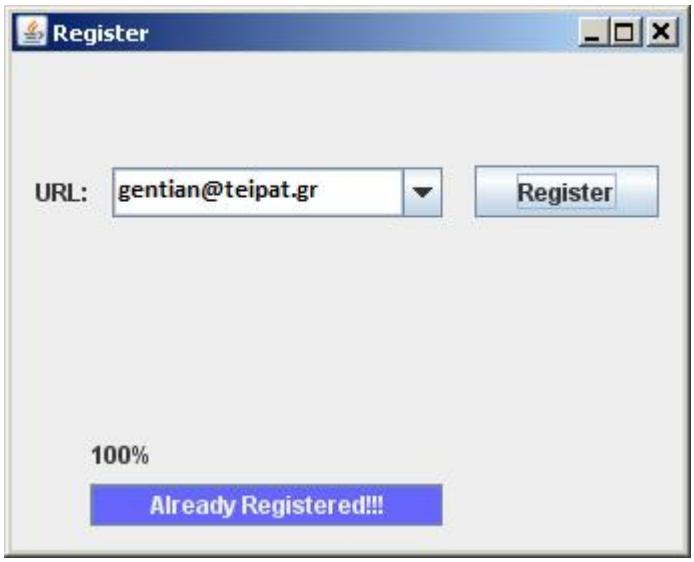
Από την πλευρά του Client βλέπουμε το γραφικό περιβάλλον καθώς και το παράθυρο Command Promt των Windows.Παρακάτω φαίνονται εικόνες οι οποίες περιγράφουν βήμα προς βήμα την εκτέλεση μιας εγγραφής .

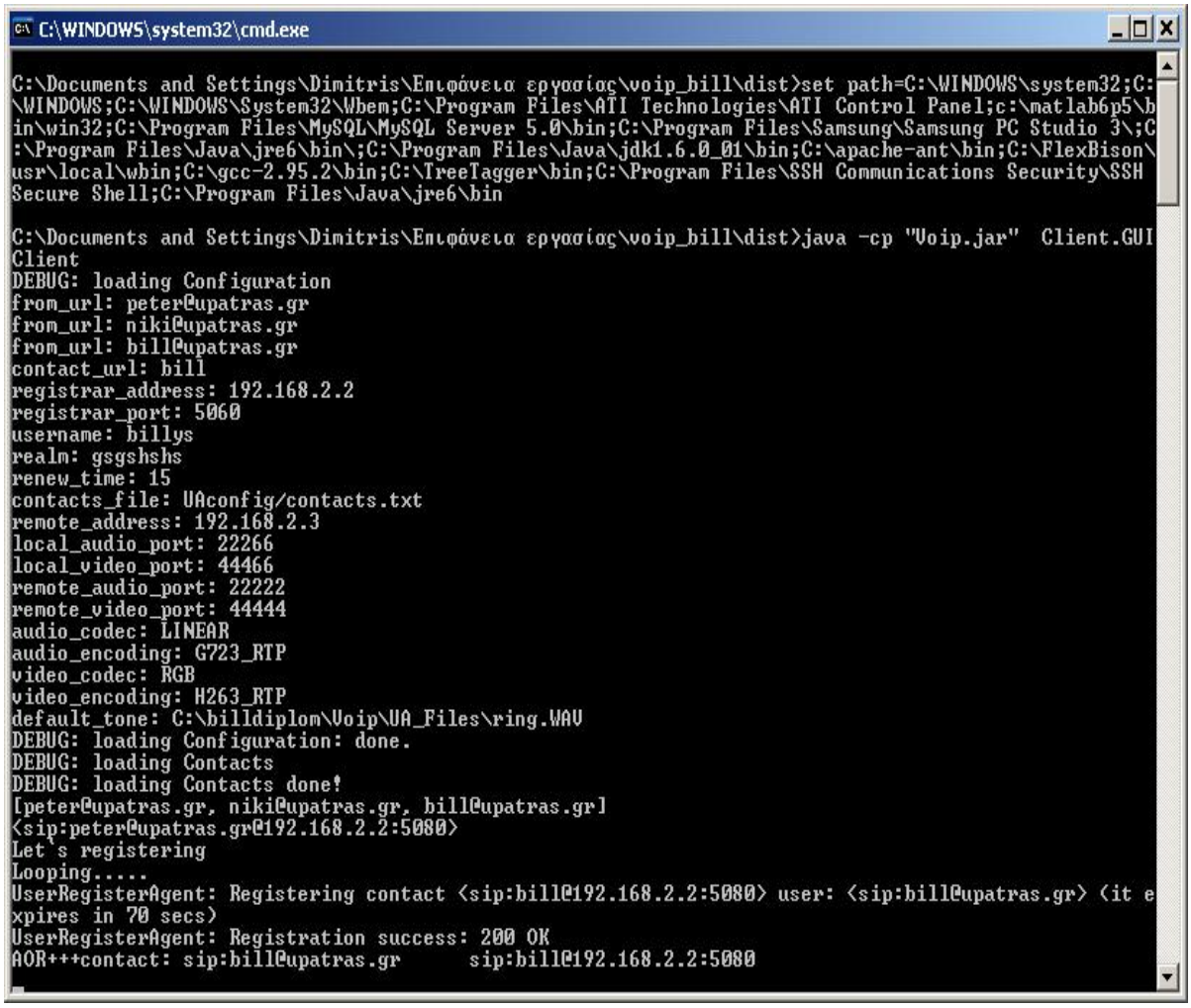
CLIENT:






Register
URL:
gentian@teipat.gr
Register
50%


Register
URL:
gentian@teipat.gr
Register
100%
Already Registered!!!


C:\WINDOWS\system32\cmd.exe
C:\Documents and Settings\Dimitris\Επιφάνεια εργασίας\voip_bill\dist>set path=C:\WINDOWS\system32;C:\WINDOWS;C:\WINDOWS\System32\Wbem;C:\Program Files\ATI Technologies\ATI Control Panel;c:\matlab6p5\bin\win32;C:\Program Files\MySQL\MySQL Server 5.0\bin;C:\Program Files\Samsung\Samsung PC Studio 3\;C:\Program Files\Java\jre6\bin\;C:\Program Files\Java\jdk1.6.0_01\bin;C:\apache-ant\bin;C:\FlexBison\usr\local\wbin;C:\gcc-2.95.2\bin;C:\TreeTagger\bin;C:\Program Files\SSH Communications Security\SSH Secure Shell;C:\Program Files\Java\jre6\bin
C:\Documents and Settings\Dimitris\Επιφάνεια εργασίας\voip_bill\dist>java -cp "Voip.jar" Client.GUIClient
DEBUG: loading Configuration
from_url: peter@upatras.gr
from_url: niki@upatras.gr
from_url: bill@upatras.gr
contact_url: bill
registrar_address: 192.168.2.2
registrar_port: 5060
username: billys
realm: gsgshshs
renew_time: 15
contacts_file: UAconfig/contacts.txt
remote_address: 192.168.2.3
local_audio_port: 22266
local_video_port: 44466
remote_audio_port: 22222
remote_video_port: 44444
audio_codec: LINEAR
audio_encoding: G723_RTP
video_codec: RGB
video_encoding: H263_RTP
default_tone: C:\billdiplom\Voip\UA_Files\ring.WAV
DEBUG: loading Configuration: done.
DEBUG: loading Contacts
DEBUG: loading Contacts done!
[peter@upatras.gr, niki@upatras.gr, bill@upatras.gr]
<sip:peter@upatras.gr@192.168.2.2:5080>
Let's registering
Looping.....
UserRegisterAgent: Registering contact <sip:bill@192.168.2.2:5080> user: <sip:bill@upatras.gr> (it expires in 70 secs)
UserRegisterAgent: Registration success: 200 OK
AOR+++contact: sip:bill@upatras.gr sip:bill@192.168.2.2:5080


Register
URL:
gentian@teipat.gr
Register
0%
error:can't register!!!

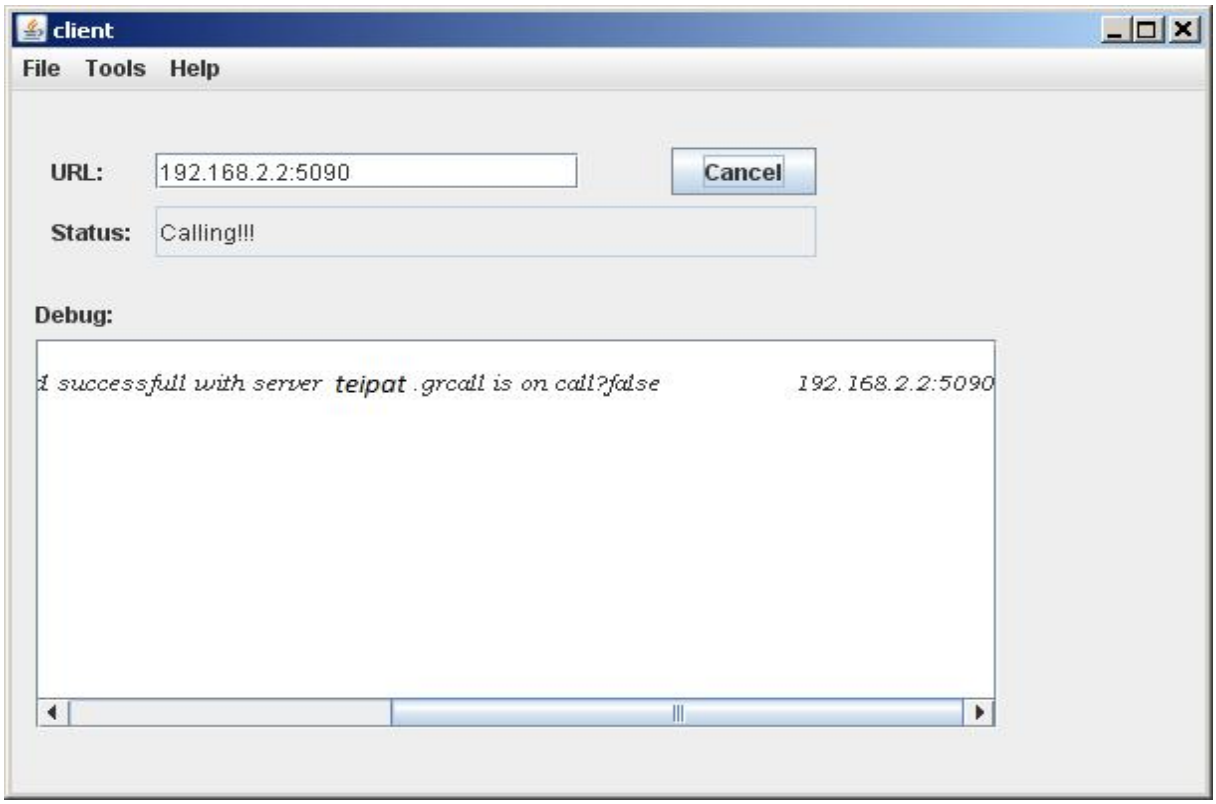
```
C:\WINDOWS\system32\cmd.exe

C:\Documents and Settings\Dimitris\Επιφάνεια εργασίας\voip_bill\dist>set path=C:\WINDOWS\system32;C:
\WINDOWS;C:\WINDOWS\System32\Wbem;C:\Program Files\ATI Technologies\ATI Control Panel;c:\matlab6p5\b
in\win32;C:\Program Files\MySQL\MySQL Server 5.0\bin;C:\Program Files\Samsung\Samsung PC Studio 3\;C
:\Program Files\Java\jre6\bin\;C:\Program Files\Java\jdk1.6.0_01\bin;C:\apache-ant\bin;C:\FlexBison\
usr\local\wbin;C:\gcc-2.95.2\bin;C:\TreeTagger\bin;C:\Program Files\SSH Communications Security\SSH
Secure Shell;C:\Program Files\Java\jre6\bin

C:\Documents and Settings\Dimitris\Επιφάνεια εργασίας\voip_bill\dist>java -cp "Voip.jar"  Client.GUI
Client
DEBUG: loading Configuration
from_url: peter@upatras.gr
from_url: niki@upatras.gr
from_url: bill@upatras.gr
contact_url: bill
registrar_address: 192.168.2.20
registrar_port: 5060
username: billys
realm: gsgshshs
renew_time: 15
contacts_file: UAconfig/contacts.txt
remote_address: 192.168.2.3
local_audio_port: 22266
local_video_port: 44466
remote_audio_port: 22222
remote_video_port: 44444
audio_codec: LINEAR
audio_encoding: G723_RTP
video_codec: RGB
video_encoding: H263_RTP
default_tone: C:\billdiplom\Voip\UA_Files\ring.WAV
DEBUG: loading Configuration: done.
DEBUG: loading Contacts
DEBUG: loading Contacts done!
[peter@upatras.gr, niki@upatras.gr, bill@upatras.gr]
<sip:peter@upatras.gr@192.168.2.2:5080>
Let's registering
Looping.....
UserRegisterAgent: Registering contact <sip:bill@192.168.2.2:5080> user: <sip:bill@upatras.gr> (it e
xpires in 70 secs)
UserRegisterAgent: Registration failure: No response from server.
```

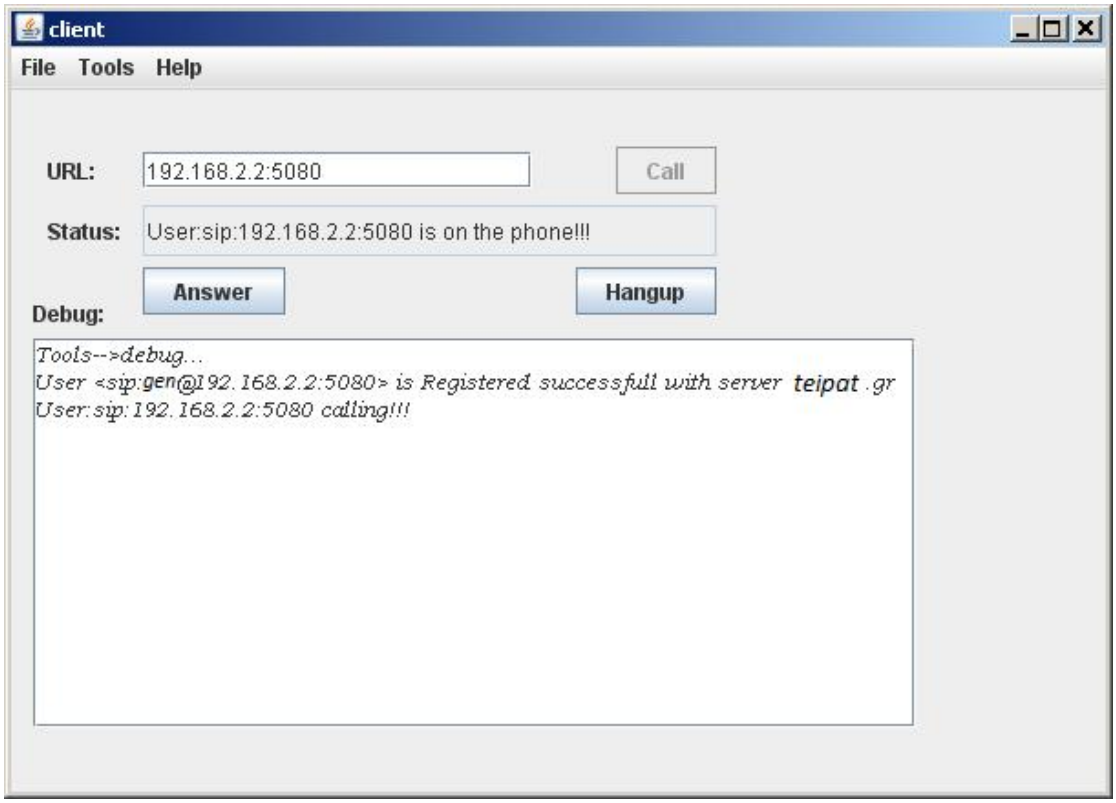
SERVER:

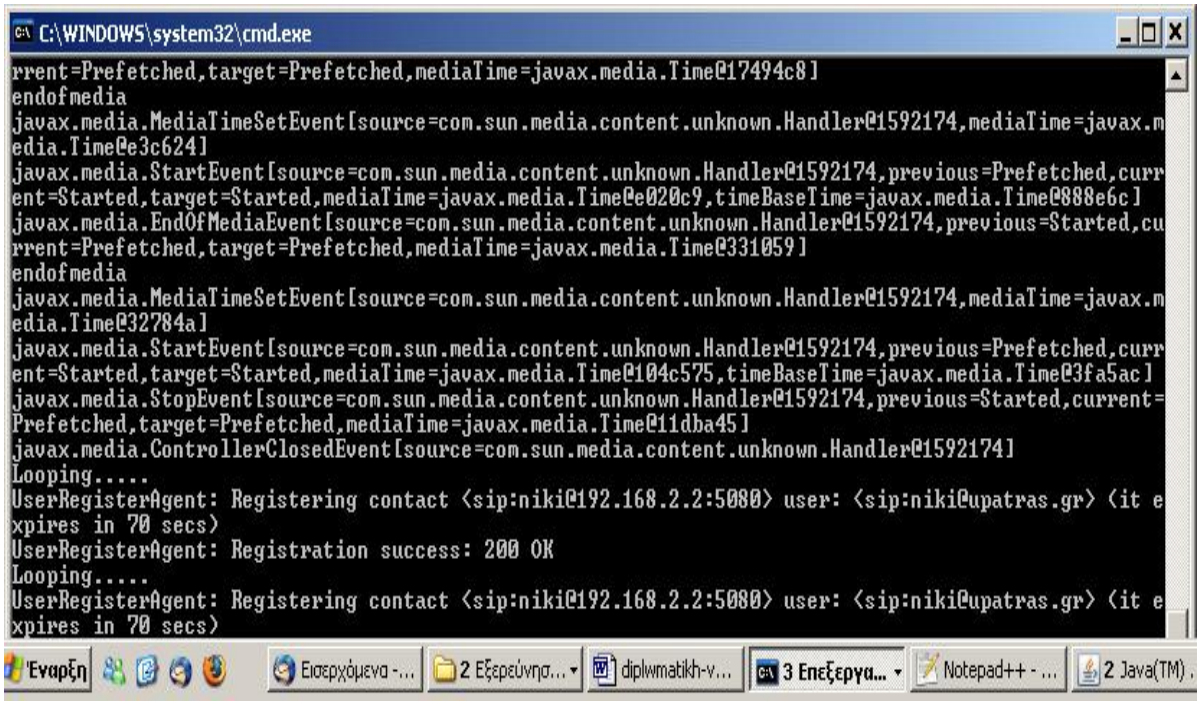
```
C:\WINDOWS\system32\cmd.exe

C:\Documents and Settings\Dimitris\Επιφάνεια εργασίας\voip_bill\dist>set path=C:\WINDOWS\system32;C:
\WINDOWS;C:\WINDOWS\System32\Wbem;C:\Program Files\ATI Technologies\ATI Control Panel;c:\matlab6p5\b
in\win32;C:\Program Files\MySQL\MySQL Server 5.0\bin;C:\Program Files\Samsung\Samsung PC Studio 3\;C
:\Program Files\Java\jre6\bin\;C:\Program Files\Java\jdk1.6.0_01\bin;C:\apache-ant\bin;C:\FlexBison\
usr\local\wbin;C:\gcc-2.95.2\bin;C:\TreeTagger\bin;C:\Program Files\SSH Communications Security\SSH
Secure Shell;C:\Program Files\Java\jre6\bin

C:\Documents and Settings\Dimitris\Επιφάνεια εργασίας\voip_bill\dist>java -cp "Voip.jar"  Server.Reg
istrar_sql
5060bill_server1800
2009-04-07 23:35:14
1
[Contact: <sip:bill@192.168.2.2:5080>
]
Hello********************************
sip:bill@upatras.gr
[sip:bill@192.168.2.2:5080]
1933790887290192.168.2.2
1
[70]
[1]
to bind den yparxei!!!
2!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
3!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
End While 1
insert into contact values("sip:bill@192.168.2.2:5080","1933790887290192.168.2.2",1,70,"2009-04-07 2
3:36:26");   perfectinsert!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
insert into bindings values("sip:bill@upatras.gr","sip:bill@192.168.2.2:5080",1);   perfectinsert!!!
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
200     1
```

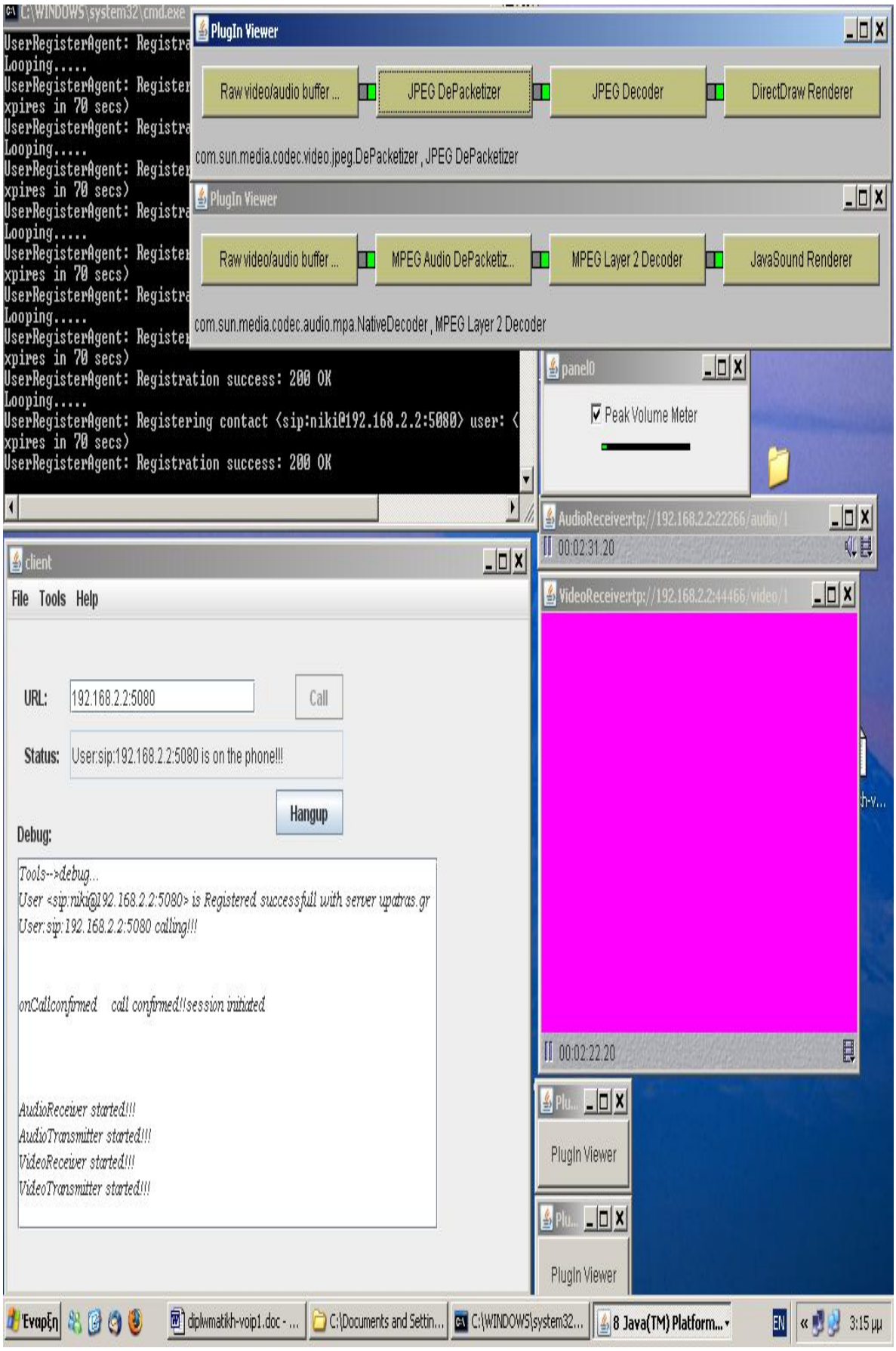
8.2 ΚΛΗΣΗ(INVITE)

User Agent:

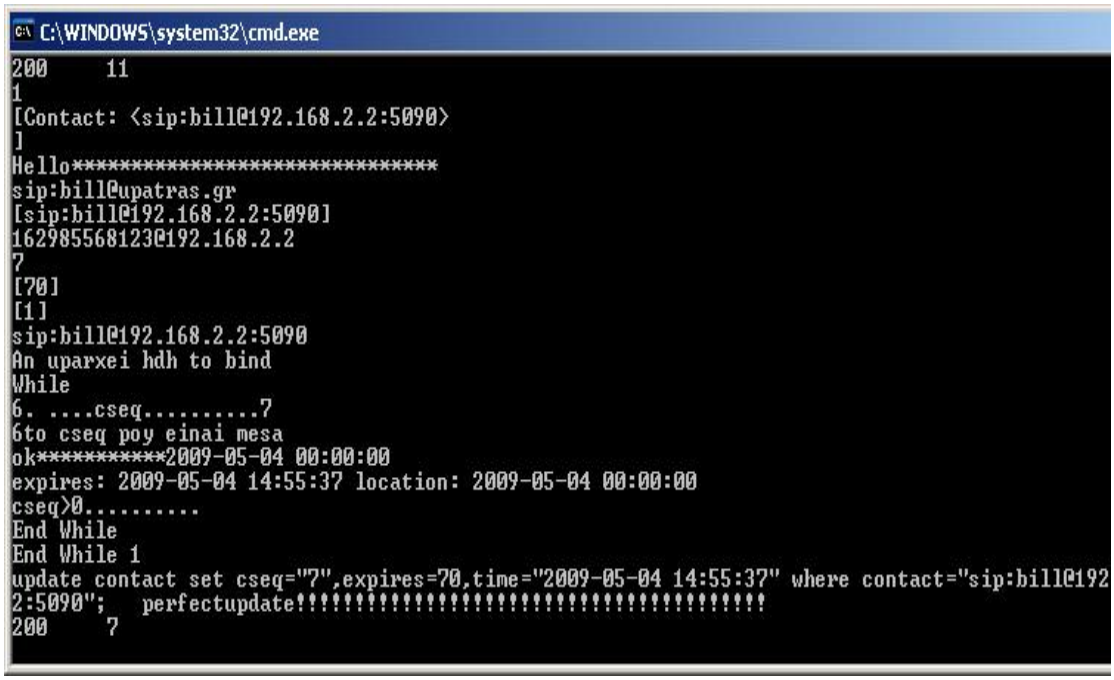



client
File Tools Help
URL:
192.168.2.2:5080
Call
Status:
User:sip:192.168.2.2:5080 is on the phone!!!
Answer
Hangup
Debug:
Tools-->debug...
User <sip:gen@192.168.2.2:5080> is Registered successfull with server teipat .gr
User:sip:192.168.2.2:5080 calling!!!


C:\WINDOWS\system32\cmd.exe
rrent=Prefetched,target=Prefetched,mediaTime=javax.media.Time@17494c8]
endofmedia
javax.media.MediaTimeSetEvent[source=com.sun.media.content.unknown.Handler@1592174,mediaTime=javax.media.Time@e3c624]
javax.media.StartEvent[source=com.sun.media.content.unknown.Handler@1592174,previous=Prefetched,current=Started,target=Started,mediaTime=javax.media.Time@e020c9,timeBaseTime=javax.media.Time@888e6c]
javax.media.EndOfMediaEvent[source=com.sun.media.content.unknown.Handler@1592174,previous=Started,current=Prefetched,target=Prefetched,mediaTime=javax.media.Time@331059]
endofmedia
javax.media.MediaTimeSetEvent[source=com.sun.media.content.unknown.Handler@1592174,mediaTime=javax.media.Time@32784a]
javax.media.StartEvent[source=com.sun.media.content.unknown.Handler@1592174,previous=Prefetched,current=Started,target=Started,mediaTime=javax.media.Time@104c575,timeBaseTime=javax.media.Time@3fa5ac]
javax.media.StopEvent[source=com.sun.media.content.unknown.Handler@1592174,previous=Started,current=Prefetched,target=Prefetched,mediaTime=javax.media.Time@11dba45]
javax.media.ControllerClosedEvent[source=com.sun.media.content.unknown.Handler@1592174]
Looping.....
UserRegisterAgent: Registering contact <sip:niki@192.168.2.2:5080> user: <sip:niki@upatras.gr> (it expires in 70 secs)
UserRegisterAgent: Registration success: 200 OK
Looping.....
UserRegisterAgent: Registering contact <sip:niki@192.168.2.2:5080> user: <sip:niki@upatras.gr> (it expires in 70 secs)
Έναρξη
Εισερχόμενα -...
2 Εξερεύνησ...
diplwmatikh-v...
3 Επεξεργα...
Notepad++ - ...
2 Java(TM)


C:\WINDOWS\system32\cmd.exe
UserRegisterAgent: Registra
Looping.....
UserRegisterAgent: Register
xpires in 70 secs)
UserRegisterAgent: Registra
Looping.....
UserRegisterAgent: Register
xpires in 70 secs)
UserRegisterAgent: Registra
Looping.....
UserRegisterAgent: Register
xpires in 70 secs)
UserRegisterAgent: Registra
Looping.....
UserRegisterAgent: Register
xpires in 70 secs)
UserRegisterAgent: Registration success: 200 OK
Looping.....
UserRegisterAgent: Registering contact <sip:niki@192.168.2.2:5080> user: (
xpires in 70 secs)
UserRegisterAgent: Registration success: 200 OK
PlugIn Viewer
Raw video/audio buffer ...
JPEG DePacketizer
JPEG Decoder
DirectDraw Renderer
com.sun.media.codec.video.jpeg.DePacketizer , JPEG DePacketizer
PlugIn Viewer
Raw video/audio buffer ...
MPEG Audio DePacketiz...
MPEG Layer 2 Decoder
JavaSound Renderer
com.sun.media.codec.audio.mpa.NativeDecoder , MPEG Layer 2 Decoder
panel0
Peak Volume Meter
AudioReceiver:rtp://192.168.2.2:22266/audio/1
00:02:31.20
VideoReceiver:rtp://192.168.2.2:44466/video/1
00:02:22.20
client
File Tools Help
URL:
192.168.2.2:5080
Call
Status:
User:sip:192.168.2.2:5080 is on the phone!!!
Hangup
Debug:
Tools-->debug...
User <sip:niki@192.168.2.2:5080> is Registered successfull with server upatras.gr
User:sip:192.168.2.2:5080 calling!!!
onCallconfirmed call confirmed!!session initiated
AudioReceiver started!!!
AudioTransmitter started!!!
VideoReceiver started!!!
VideoTransmitter started!!!
PlugIn Viewer
PlugIn Viewer
Έναρξη
diplwmatikh-voip1.doc - ...
C:\Documents and Settin...
C:\WINDOWS\system32...
8 Java(TM) Platform...
EN
3:15 μμ

Server:


C:\WINDOWS\system32\cmd.exe
200 11
1
[Contact: <sip:bill@192.168.2.2:5090>
]
Hello**********
sip:bill@upatras.gr
[sip:bill@192.168.2.2:5090]
162985568123@192.168.2.2
7
[70]
[1]
sip:bill@192.168.2.2:5090
An uparxei hdh to bind
While
6. ....cseq.....7
6to cseq poy einai mesa
ok**********2009-05-04 00:00:00
expires: 2009-05-04 14:55:37 location: 2009-05-04 00:00:00
cseq>0.....
End While
End While 1
update contact set cseq="7",expires=70,time="2009-05-04 14:55:37" where contact="sip:bill@192
2:5090"; perfectupdate!!
200 7

ΒΙΒΛΙΟΓΡΑΦΙΑ-ΑΝΑΦΟΡΕΣ

[1] RFC 3261 - SIP: Session Initiation Protocol-http://www.faqs.org/rfcs/rfc3261.html
[2] MjSip - Complete Java SIP stack implementation-http://www.mjsip.org/
http://jsip.sourceforge.net/api/index.html
[3]Overview (The JAIN-SIP-1.2 RI For the People !)- http://snad.ncsl.nist.gov/proj/iptel/jain-sip-1.2/javadoc/
[4]SIP: Papers-http://www.cs.columbia.edu/sip/papers.html
[5]JAIN and Java in Communications Documentation-
http://java.sun.com/products/jain/reference/docs/index.html
[6]Programming SIP / About SIP / Programming SIP-
http://www.sipcenter.com/sip.nsf/html/Programming+SIP
[7]Purpose of SIP | iptel.org-Internet Telephony-http://www.iptel.org/sip/intro/purpose
[8]Portal for contributing to the promotion and knowledge of SIP and associated technologies, by posting and maintaining white papers-http://www.tech-invite.com/
[9]sip: Sofia SIP User Agent Library - "sip" - SIP Parser Module-http://sofia-sip.sourceforge.net/refdocs/sip/index.html
[10]SIP proxy with a user location database and authentication of registration and call setup requests-http://switzernet.com/people/emin-gabrielyan/070424-sip-authentication/#_Toc165190448
[11]Session Initiation Protocol - Wikipedia, the free encyclopedia-
http://en.wikipedia.org/wiki/Session_Initiation_Protocol
[12]Theora RTP payload format-http://lists.xiph.org/pipermail/xiph-rtp/2005-April/000214.html
[13]IVCi's goal is to demonstrate the value of video conferencing and what it can do for your organization. We realize this commitment by helping our customers define the best configuration for their video conferencing needs while ensuring that they achieve the best return on their investment.-http://www.ivci.com/videoconferencing_index.html
[14]SIP, Session Initiation Protocol-http://www.networksorcery.com/enp/protocol/sip.htm
[15]Relation among Call, Dialog, Transaction, Message of SIP A Made Easy Tutorial-
http://www.geocities.com/intro_to_multimedia/SIP/relation.html
[16]jmf examples-
http://www.cs.odu.edu/~cs778/spring04/lectures/jmfsolutions/examplesindex.html
[17]JMF 2.1.1 Solutions-
http://java.sun.com/javase/technologies/desktop/media/jmf/2.1.1/solutions/index.html
[18]Transmitting RTP Media Streams-
http://www.cs.odu.edu/~cs778/jmflects/lect9RTPSending.html
[19] 6 Custom DataSource and DataSink for JMF Versions 2.0 and 2.1-
http://www.stanford.edu/dept/itss/docs/oracle/10g/appdev.101/b10840/mm_jmfext.htm
[20]MySQL :: MySQL 5.0 Reference Manual :: 27.4 MySQL Connector/J-
http://ftp.ntua.gr/pub/databases/mysql/doc/refman/5.0/en/connector-j.html
[21]Using JDBC with MySQL, Getting Started-
http://www.developer.com/java/data/article.php/3417381
[22]MySQL Conference & Expo 2009 - O'Reilly Conferences, April 20 - 23, 2009, Santa Clara, CA-http://en.oreilly.com/mysql2009/public/content/home

[23]RFC 1890 - RTP Profile for Audio and Video Conferences with Minimal Control-http://tools.ietf.org/html/rfc1890
[24]RFC 4317 - Session Description Protocol (SDP) Offer/Answer Examples-http://www.rfc-archive.org/getrfc.php?rfc=4317
[25]RFC 4566 - SDP: Session Description Protocol-http://tools.ietf.org/html/rfc4566
[26]NAT and VOIP - voip-info.org-http://www.voip-info.org/wiki/view/NAT+and+VOIP
[27]Linphone (en)- http://en.flossmanuals.net/Linphone/Configuring
[28]ΤΗΛΕΔΙΑΣΚΕΨΗ-http://homepages.pathfinder.gr/teiepvc/basic.htm
[29]Technology and Education in the 21st century-http://blog.edu.gr/archives/144
[30]Η επανάσταση τηλεφωνικών υπηρεσιών VoIP-http://www.answeringphoneservices.com/el/voip/voiptelephoneservices.php