



ΤΕΙ ΔΥΤΙΚΗΣ ΕΛΛΑΔΑΣ

ΤΜΗΜΑ ΔΙΟΙΚΗΣΗΣ ΕΠΙΧΕΙΡΗΣΕΩΝ - ΜΕΣΟΛΟΓΓΙ



Δίκτυα P2P

Τεχνικά Χαρακτηριστικά - Αρχιτεκτονική και Τρόποι Αξιοποίησης

Γράψας Αργύρης – Β.

Τσώτας Κων/νος

Επιβλέπων Καθηγητής: Θανόπουλος Αριστομένης

ΜΕΣΟΛΟΓΓΙ 2014

Περιεχόμενα

1.	ΕΙΣΑΓΩΓΗ	5
2.	Τί είναι το Peer to Peer	7
3.	Γενικά χαρακτηριστικά των συστημάτων ομότιμων κόμβων.....	9
3.1	Τεχνικά χαρακτηριστικά	10
3.2	Το υβριδικό μοντέλο	10
3.3	Αποκεντρωμένη τοπολογία.....	12
3.4	Υβριδικής τοπολογίας	13
3.5	Το καθαρό Peer to Peer μοντέλο	15
3.6	Πλεονεκτήματα του υβριδικού μοντέλου.....	15
3.7	Μειονεκτήματα του υβριδικού μοντέλου.....	16
3.8	Πλεονεκτήματα του καθαρού μοντέλου.....	16
3.9	Μειονεκτήματα του καθαρού μοντέλου	17
4.	Γιατί να χρησιμοποιήσει κανείς το μοντέλο Peer to Peer	18
4.1	Τεχνικά πλεονεκτήματα	18
4.2	Κοινωνικά πλεονεκτήματα	18
5.	Χρήσιμες εφαρμογές για δίκτυα P2P	19
5.1	Τεχνολογία peer-to-peer	23
5.2	Μειονεκτήματα	23
6.	Κατηγορίες Peer to Peer δικτύων.....	24
6.1	Κεντροποιημένα P2P συστήματα	24
6.1.1	Αρχιτεκτονική πελάτη-εξυπηρετητή (client – server)	26
6.1.2	Πλεονεκτήματα - Μειονεκτήματα.....	26
6.2	Μη κεντροποιημένα P2P συστήματα	28
6.2.1	Δομημένα P2P συστήματα	28
6.2.1.1	Δομές δικτύων και δρομολόγηση	28
6.2.1.2	CAN (Content-Addressable Network).....	29
6.2.1.3	Chord	30
6.2.1.4	Replication – Δημιουργία Αντιγράφων	31
6.2.2	Μη δομημένα P2P συστήματα.....	32
6.2.3	Μέθοδοι τοποθέτησης αντιγράφων (Replication).....	39
6.2.4	Μέθοδοι διατήρησης των αντιγράφων στο δίκτυο	40

7.	<i>Τεχνικές Διαχείρισης Αξιοπιστίας σε ένα Peer-to-Peer Σύστημα – Trust Management.</i>	45
7.1	XREP	46
7.2	P-GRID.....	47
7.3	NICE	50
7.4	Reputation Framework using Currency.....	52
7.7	Αξιολόγηση και Σύγκριση	54
7.8	Συμπέρασμα.....	60
8.	<i>Τα δίκτυα ανταλλαγής αρχείων</i>	62
8.1	Napster	62
8.2	OpenNap.....	64
8.3	Gnutella	64
8.4	FastTrack.....	65
8.5	OpenFT	66
8.6	eDonkey και OverNet	66
8.7	SoulSeek	67
8.8	Bittorent	67
8.9	Direct Connect.....	68
8.10	Συνδρομητικά Δίκτυα.....	68
8.11	Δημοφιλέστερα προγράμματα P2P	69
8.12	<i>Παρουσίαση του τρόπου λειτουργίας ενός δικτύου P2P.....</i>	70
8.13	Στατιστικές έρευνες.....	73
9.	<i>Παρουσίαση της λειτουργίας ενός προγράμματος P2P.....</i>	75
9.1	Προηγμένες Ρυθμίσεις.....	80
9.2	Σύγκριση των δημοφιλέστερων προγραμμάτων	85
10.	<i>Βελτιώσεις που χρειάζονται.....</i>	87
11.	<i>Βιβλιογραφία.....</i>	89

Πρόλογος

Αυτή η πτυχιακή εργασία έχει ως στόχο να κατανοήσει ο αναγνώστης την βασική αρχή λειτουργίας των δικτύων ανταλλαγής αρχείων P2P, αλλά και κάποιες τεχνικές αξιοπιστίας οι οποίες είναι ζωτικές για την ομαλή λειτουργία αυτών. Θα αναφερθούμε στα πλεονεκτήματα και τα μειονεκτήματα της κάθε τεχνικής, έτσι ώστε ο αναγνώστης να είναι σε θέση να επιλέξει πιο δίκτυο είναι κατάλληλο για την εφαρμογή που τον ενδιαφέρει.

Αναλυτικότερα, θα προβάλλουμε τα τεχνικά χαρακτηριστικά του κάθε μοντέλου, καθώς επίσης και τα πλεονεκτήματα-μειονεκτήματα αυτών. Έπειτα θα παρουσιάσουμε κάποιους από τους βασικούς λόγους που μπορεί να παροτρύνουν κάποιον να επιλέξει ένα τέτοιο δίκτυο σύμφωνα με τις ανάγκες του, καθώς επίσης και κάποιες εφαρμογές των δικτύων που έχουν ήδη υλοποιηθεί.

Ακόμη, θα δούμε τις κατηγορίες στις οποίες χωρίζονται τα δίκτυα Peer2Peer, τις αρχιτεκτονικές που χρησιμοποιούν και πώς αυτές εφαρμόζονται στην πράξη.

Μετά την αναφορά, αξιολόγηση και σύγκριση των τεχνικών διαχείρισης αξιοπιστίας, θα αναφέρθουμε στα κυριότερα δίκτυα ανταλλαγής αρχείων που χρησιμοποιούν το μοντέλο P2P, καθώς και τα δημοφιλέστερα προγράμματα που υπάρχουν διαθέσιμα στην αγορά.

Τέλος, θα παρουσιάσουμε την εφαρμογή uTorrent και κάποιες από τις βασικές λειτουργίες που μας παρέχει και θα την συγκρίνουμε με μια αντάξια του είδους εφαρμογή, το Vuze/Azureus προσπαθώντας να κατανοήσουμε τις διαφορές που υπάρχουν μεταξύ δύο αντίστοιχων εφαρμογών και ποιά είναι καταλληλότερη για τις ανάγκες του τελικού χρήστη.

ΠΕΡΙΛΗΨΗ

Τα Peer-to-Peer (p2p) είναι κατακεντρωμένα δίκτυα στα οποία υπάρχουν κόμβοι. Οι κόμβοι έχουν ίσους ρόλους και ικανότητες και ανταλλάσσουν πληροφορίες μεταξύ τους.

Οι κατηγορίες στις οποίες διακρίνονται είναι οι εξής δύο:

1. Τα κεντροποιημένα δίκτυα και 2. τα μη κεντροποιημένα. Τα μη κεντροποιημένα κατηγοριοποιούνται σε δομημένα (όπως είναι τα δίκτυα CAN και CHORD) και μη δομημένα όπως είναι το Gnutella ανάλογα πάντα με τον τρόπο που οργανώνονται οι κόμβοι και δρομολογούνται οι ερωτήσεις στο δίκτυο.

Ένα μεγάλο μέρος της βιβλιογραφίας, που αφορά τα Peer-to-Peer δίκτυα και ένας από τους λόγους που τα διαφοροποιεί μεταξύ τους, αναφέρεται στην αναζήτηση και εύρεση κάποιας πληροφορίας στους κόμβους τους.

Η δημιουργία αντιγράφων των δεδομένων (replication) σε κάποιους από τους κόμβους του δικτύου και η ενημέρωση αυτών είναι ένα άλλο θέμα που απασχολεί τους ερευνητές αυτού του χώρου.

Έχουν γίνει αρκετές προσπάθειες έτσι ώστε τα P2P δίκτυα να διατηρούν ιδιότητες όπως η κλιμάκωση (scalability) και η απλότητα χωρίς ταυτόχρονα να μειώνεται η καλή απόδοση του δικτύου και στο σύνολο τους αυτές οι ιδιότητες να εξάγουν ικανοποιητικά αποτελέσματα σε ερωτήσεις χρηστών. Στην προσπάθεια να επιτευχθούν τα παραπάνω δημιουργήθηκαν τα P2P δίκτυα με ιδιότητες όπως η ομαδοποίηση των κόμβων με βάση τα ενδιαφέροντα τους ή το είδος των δεδομένων που προσφέρουν στο δίκτυο (associative overlays, semantic overlay networks).

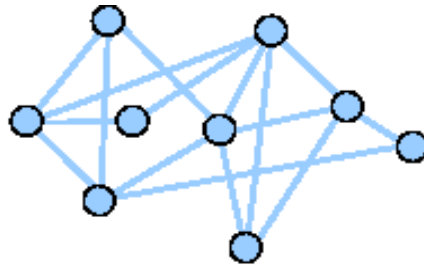
Επιπλέον αρκετές βελτιώσεις εισέρχονται στα P2P δίκτυα ώστε αυτά να υποστηρίζουν όχι μόνο απλές ερωτήσεις που αφορούν στο όνομα των δεδομένων αλλά και ερωτήσεις που αφορούν σε πολλά γνωρίσματα και σε ένα εύρος τιμών αυτών.

ΚΕΦΑΛΑΙΟ 1

1. ΕΙΣΑΓΩΓΗ

Τα P2P δίκτυα μπορούμε να πούμε πως είναι μία από τις πιο γρήγορα αναπτυσσόμενες και πιο γνωστές εφαρμογές του internet. Ο όρος αφορά τα κατανεμημένα δίκτυα όπου κάθε κόμβος (peer) είναι αυτόνομος, έχει ισότιμα δικαιώματα χρήσης των πόρων με τους υπόλοιπους και συμμετέχει στο δίκτυο προσφέροντας τα δεδομένα του.

Στα δίκτυα αυτά οι κόμβοι τους οργανώνονται με κάποιες ιδιότητες και όχι ανάλογα με την απόστασή τους στο δίκτυο, δηλαδή ένας κόμβος δεν συνδέεται απαραίτητα με αυτούς που είναι πιο κοντά στο δίκτυο. Ένα παράδειγμα P2P δικτύου φαίνεται στο Σχήμα 1.1.



Σχήμα 1.1: P2P δίκτυο

Το Napster ήταν η κύρια αιτία που τα P2P δίκτυα έγιναν ευρέως γνωστά. Όταν αναφερόμαστε στο Napster εννοούμε μια κεντροποιημένη αρχιτεκτονική δικτύου όπου όλα τα δεδομένα βρίσκονται καταχωρημένα σε ένα κεντρικό ευρετήριο από όπου τα μοιράζονται έπειτα οι κόμβοι μεταξύ τους. Η ιδέα των κεντροποιημένων συστημάτων άρχισε να μειονεκτεί, ενώ οι χρήστες του διαδικτύου στράφηκαν σε μία μη κεντροποιημένη αρχιτεκτονική P2P (Gnutella).

Ο τρόπος με τον οποίο θα γινόταν η αναζήτηση των αποθηκευμένων δεδομένων στους κόμβους ενός P2P δικτύου είναι πρόβλημα που απασχόλησε σημαντικά τους ειδικούς αυτού του χώρου, αφού ο κάθε τρόπος είχε τα πλεονεκτήματά του καθώς και τα μειονεκτήματά του.

Οι μηχανισμοί αναζήτησης σε ένα P2P δίκτυο ταξινομούνται στις εξής τρεις κατηγορίες:

- στους μηχανισμούς χωρίς τη χρήση ευρετηρίου,
- μηχανισμούς με τη χρήση ενός κεντρικού ευρετηρίου,
- και μηχανισμούς με ευρετήρια σε κάθε κόμβο.

Έτσι σε διάφορα δίκτυα διακρίνουμε την τυφλή αναζήτηση (blind search), αναζήτηση με βάση κάποιον πίνακα κατακερματισμού και την αναζήτηση με βάση κάποιες πληροφορίες-στατιστικά. Κοινός σκοπός σε κάθε νέα προσπάθεια είναι

- η απόδοση του δικτύου,
- και η επίτευξη και διατήρηση ιδιοτήτων (όπως είναι η κλιμάκωση και η προσαρμοστικότητα)

από τα οποία εξαρτάται άμεσα η επιτυχία του!Σημαντικό θέμα επίσης είναι αυτό που αφορά στην καλύτερη απόδοση του δικτύου (replication) καθώς επίσης και η ενημέρωση αυτών με αποδοτικούς αλγορίθμους.

Η ανάγκη για διατήρηση της απλότητας ενώ ταυτόχρονα απαιτείται και η απόκτηση της σημαντικής ιδιότητας της κλιμάκωσης, προκάλεσαν αρκετές προσπάθειες, νέες προσεγγίσεις και νέες ιδέες πάνω στα P2P.

Μερικές από αυτές είναι οι ιδέες των “interest-based shortcuts”, “associative overlays” και “semantic overlay networks”. Έχουμε να κάνουμε σε αυτήν την περίπτωση με P2P δίκτυα στα οποία οι κόμβοι ομαδοποιούνται με βάση τα ενδιαφέροντά τους ή με βάση τα δεδομένα που προσφέρουν στο δίκτυο.

Επιπλέον παρουσιάστηκαν νέες προσεγγίσεις των P2P δικτύων ώστε αυτά να υποστηρίζουν και σύνθετες ερωτήσεις, δηλαδή ερωτήσεις που τίθενται σε πολλά γνωρίσματα ή ερωτήσεις που τίθενται για ένα εύρος τιμών των γνωρισμάτων.

ΚΕΦΑΛΑΙΟ 2

2. Τί είναι το Peer to Peer

Συστήματα Ομοτίμων Κόμβων και η εξέλιξη τους

Οι ερμηνείες για το τι σημαίνει peer to peer ποικίλουν. Η πιο απλουστευμένη είναι ότι peer to peer δεν είναι πελάτης-εξυπηρετητής(client/server). Βέβαια μια πιο σωστή ερμηνεία είναι ότι πρόκειται για ένα πρότυπο επικοινωνιών στο οποίο κάθε συμβαλλόμενο μέρος έχει τις ίδιες δυνατότητες και κάθε συμβαλλόμενο μέρος μπορεί να ξεκινήσει μια σύνοδο επικοινωνίας.

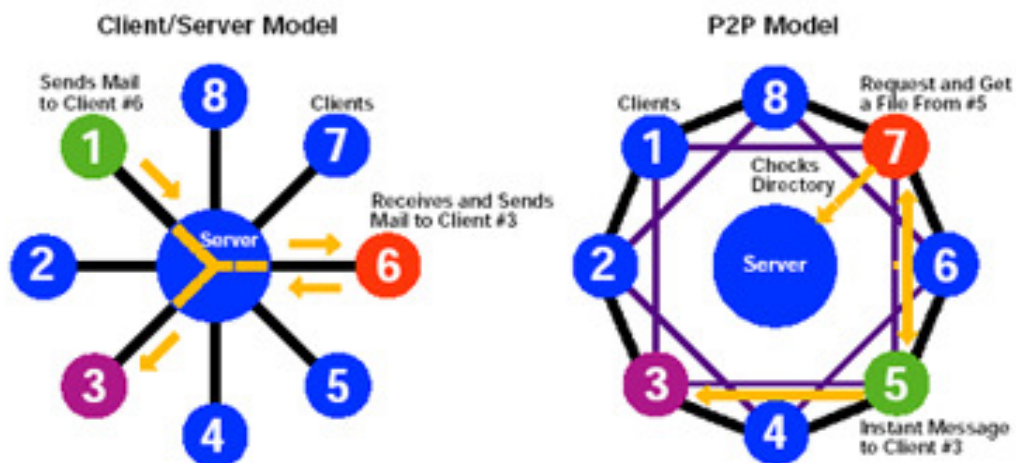
Η επικοινωνία αυτή σε ορισμένες περιπτώσεις επιτυγχάνεται όταν κάθε επικοινωνιακός κόμβος αναπτύσσεται με τέτοιο τρόπο ώστε να είναι ταυτόχρονα και εξυπηρετητής και πελάτης. Άρα ένα σύνολο υπολογιστών, έχει την δυνατότητα να μοιράζει πληροφορίες άμεσα ο ένας από τον σκληρό δίσκο του άλλου. Να αναφέρουμε ότι ένα καθαρό δίκτυο Peer-to-Peer είναι αυτό στο οποίο δεν υπάρχει καθόλου έλεγχος από server και ότι το πρόγραμμα που τρέχει σε κάθε υπολογιστή-κόμβο έχει ακριβώς την ίδια λειτουργικότητα.

Σε ένα δίκτυο P2P μεταφοράς αρχείων έννοιες όπως πελάτες και εξυπηρετητές δεν υπάρχουν καθώς υπάρχουν μόνο οι έννοιες των «peers» ή «ομότιμων κόμβων» που δρουν ταυτόχρονα και ως πελάτες αλλά και ως εξυπηρετητές ανταλλάσσοντας πληροφορίες επί ίσοις όροις με τους υπόλοιπους κόμβους του δικτύου.

Αυτό το μοντέλο διαφέρει από το μοντέλο πελάτη-εξυπηρετητή, όπου η επικοινωνία γίνεται μέσω ενός κεντρικού εξυπηρετητή. Η δήλωση αυτή αναλύεται στη συνέχεια.

Η αρχιτεκτονική ενός δικτύου Peer-to-Peer έχει τα εξής παρακάτω χαρακτηριστικά:

- Μια εφαρμογή Peer-to-Peer είναι διαθέσιμη σε όλο το δίκτυο.
- Ένας κόμβος μπορεί να είναι ταυτόχρονα και πελάτης (client) και εξυπηρετητής (server) “SERVer +cliENT = SERVENT” concept.
- Υπάρχει πλήρης αυτονομία από τους κεντρικούς servers.
- Η εύκολη χρήση του συστήματος.
- Χρησιμοποιεί κάθε πόρο του συστήματος (αποθηκευτικό χώρο, υπολογιστική δύναμη, άνθρωπος).
- Λειτουργεί πέρα από το DNS (Domain Server Name).
- Μπορεί να λειτουργεί σε εξαιρετικά ασταθές περιβάλλον όπου οι διευθύνσεις των υπολογιστών δε μπορούν να προβλεφθούν.
- Αυτοοργανώνεται και διαχειρίζεται μόνο του την απόδοση του συστήματος και την διαχείριση των πόρων.



Σχέδιο 1.2

Διαφορές Δόμησης μεταξύ κοινού Δικτύου και P2P

Πηγή: <http://pdos.csail.mit.edu/papers/>

ΚΕΦΑΛΑΙΟ 3

3. Γενικά χαρακτηριστικά των συστημάτων ομότιμων κόμβων

Δεν υπάρχει κεντρικός συντονισμός

Υπάρχει εξάρτηση του κόμβου με την επικοινωνία που έχει με τους γείτονες του. Κάθε χρήστης λειτουργεί ως πελάτης (client) αλλά και ως εξυπηρετητής (server). Δεν μπορεί να επικοινωνήσει ταυτόχρονα με όλο το δίκτυο.

Η πληροφορία στο δίκτυο είναι προσβάσιμη από κάθε κόμβο.

Ο τρόπος της επικοινωνίας είναι ικανός να δίνει πρόσβαση σε όλα τα δεδομένα του δικτύου.

Αυτονομία του κόμβου.

Μπορεί να μπει στο δίκτυο και να αποχωρήσει από το δίκτυο όποια στιγμή θελήσει, χωρίς να πρέπει να περιμένει κάποιον άλλον κόμβο (π.χ. ολοκλήρωση μεταφοράς αρχείου).

Αναξιόπιστία των κόμβων

Κάθε κόμβος θεωρείται αναξιόπιστος γιατί μπορεί να εισέρθει ή να εξέρθει στο σύστημα οποιαδήποτε στιγμή θέλει.

Τα συστήματα ως τώρα χρησιμοποιήσαν κυρίως τρεις γενικές αρχιτεκτονικές:

- **Κεντροποιημένη αρχιτεκτονική:**

Οι χρήστες του συνδεόντουσαν σε ένα κεντρικό υπολογιστή, και όλες οι αναζητήσεις αρχείων γινόντουσαν από εκεί. Τέτοια αρχιτεκτονική είχε το Napster. Ο υπολογιστής γνώριζε και τα αρχεία και τους χρήστες που βρίσκονταν στο σύστημα.

- **Κατανεμημένη αρχιτεκτονική**

Όλη η κίνηση στο δίκτυο γινόταν με την επικοινωνία μεταξύ των κόμβων. Συστήματα όπως το Gnutella και το Freenet χρησιμοποίησαν την τεχνική αυτή.

- **Ιεραρχική αρχιτεκτονική**

Ο συνδυασμός των δύο παραπάνω αρχιτεκτονικών καταλήγει σε αυτήν την αρχιτεκτονική που χρησιμοποίησαν συστήματα όπως το Kazaa δηλ. κάποιοι κόμβοι είναι υπερκόμβοι.

3.1 Τεχνικά χαρακτηριστικά

Οι εφαρμογές P2P μπορούν να διακριθούν, ανάλογα με τον τρόπο λειτουργίας τους, σε δύο βασικές κατηγορίες:

- **στις καθαρές (pure)**
- **και τις υβριδικές (hybrid).**

Υβριδικές είναι εκείνες οι οποίες χρησιμοποιούν κάποιον ή κάποιους servers οι οποίοι παίζουν το ρόλο του διαμεσολαβητή μεταξύ των διάφορων peers (ομότιμων κόμβων).

Σαν **καθαρές** εφαρμογές Peer to Peer αναφέρονται αυτές, των οποίων η λειτουργία δεν εξαρτάται από την ύπαρξη κάποιου κεντρικού server.

3.2 Το υβριδικό μοντέλο

Το υβριδικό μοντέλο είναι η μίξη του προηγούμενου μοντέλου με την δοκιμασμένη αρχιτεκτονική client-server. Εδώ εμπλέκεται και κάποιος κεντρικός server ο οποίος συντονίζει τις λειτουργίες των clients. Η τελική «συναλλαγή» θα γίνει μεταξύ των δύο ομότιμων οντοτήτων, εκ των οποίων η μία θα προσφέρει και η άλλη θα καταναλώνει την υπηρεσία ενώ θα μπορούσαν όλες να επικοινωνούν με έναν κεντρικό υπολογιστή για να διανείμουν την πληροφορία, όπως στο Napster.

Μία άλλη κατηγορία κατηγορία είναι οι εφαρμογές τύπου διανομής αρχείων (file sharing) όπως είναι το BitTorrent και το Napster και οι εφαρμογές κατανεμημένης επεξεργασίας (distributed computing) οι οποίες χρησιμοποιούν την αναξιοποίητη ισχύ πολλών υπολογιστών για να λύσουν τα προβλήματα που σε άλλη περίπτωση θα απαιτούνταν η χρήση ενός υπερυπολογιστή (SETI@Home). Παρακάτω θα εξετάσουμε αναλυτικότερα τις εφαρμογές διανομής αρχείων.

Όπως γνωρίζουμε, το TCP/IP είναι το θεμελιώδες πρωτόκολλο για τη μεταφορά των πληροφοριών μέσω του διαδικτύου. Το πρωτόκολλο HTTP χτίζεται πάνω από το TCP/IP και επιτρέπει την επικοινωνία μεταξύ των υπολογιστών για τη μεταφορά υπερκειμένου - ιστοσελίδων. Οι ομότιμοι κόμβοι συνδέονται ο ένας με τον άλλο μέσω του διαδικτύου βάση των παραπάνω τις περισσότερες φορές.

Η μεταφορά των δεδομένων μεταξύ των ομότιμων οντοτήτων πρέπει να γίνεται αποτελεσματικά και όσο το δυνατόν ταχύτερα (εξαρτάται και από την περίπτωση) και σε αυτό βοηθούν οι πολλές τοπολογίες που υπάρχουν.

Χωρίζονται σε:

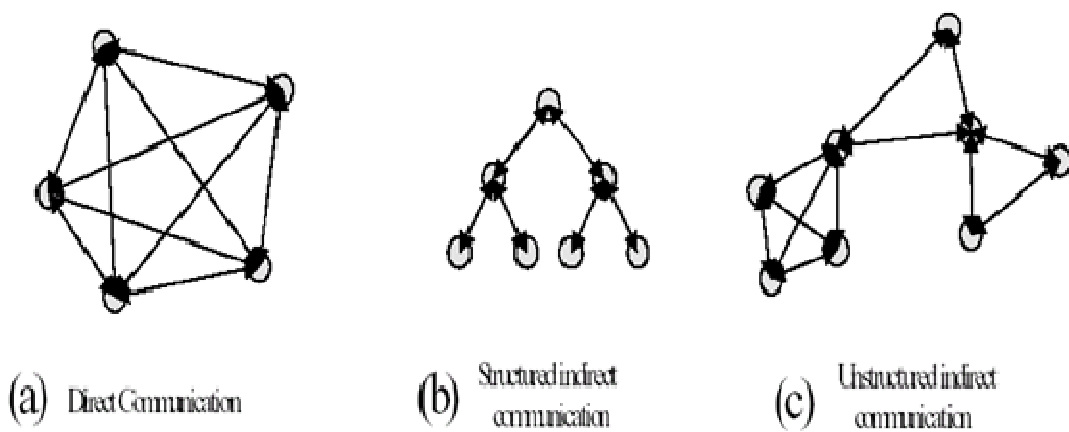
- Αποκεντρωμένη
- Υβριδική
- Ιεραρχική
- Συγκεντρωτική
- Τροπολογία δαχτυλιδιών

Μόνο σε δύο από τις παραπάνω τοπολογίες αξίζει να αναφερθούμε λίγο εκτενέστερα, στα αποκεντρωμένα και υβριδικά δίκτυα p2p, καθώς οι άλλες τοπολογίες υπερβαίνουν τα όρια αυτής της εργασίας.

3.3 Αποκεντρωμένη τοπολογία

Διατηρείται καθαρά η αυτονομία των κόμβων και πληροφορίες μπορούν να μεταφερθούν σε κάθε κόμβο.

- Με απευθείας επικοινωνία μεταξύ των κόμβων του δικτύου. (εικ. a)
- Με ιεραρχική ή δομημένη έμμεση επικοινωνία όπου επικοινωνεί έμμεσα με τους περισσότερους κόμβους αλλά με κάποιους επικοινωνεί άμεσα όπως με γόνους ή απόγονους σε ένα δέντρο ή μόνο με προηγούμενο κ επόμενο σε μια λίστα, δηλαδή με αυτούς που σχετίζεται στην δομή (εικ b).
- Με μη δομημένη έμμεση επικοινωνία, δηλαδή να γίνει έμμεση επικοινωνία με όλους εκτός από τους γειτονικούς που μπορεί να γίνει άμεση. (εικ. c)



Σχήμα 1.3 Τύποι αποκεντρωμένης τοπολογίας Πηγή: <http://pdos.csail.mit.edu/papers/>

3.4 Υβριδικής τοπολογίας

Περιέχουν τουλάχιστον ένα κεντρικό σημείο ελέγχου ή πληροφορίας.

- **Ένας κεντρικός εξυπηρετητής**

περιέχει έναν κατάλογο για σημείο αναφοράς στα δεδομένα και επεξεργασία του δικτύου (εικ d)

- **Υπολογιστικό μοντέλο**

Υπάρχει μεν ένας και μοναδικός εξυπηρετητής που επεξεργάζεται τους κόμβους και διαχειρίζεται τα δεδομένα οι κόμβοι όμως μπορούν και να ανταλλάσσουν δεδομένα / μηνύματα αυτόνομα.

- **Πολλαπλοί Εξυπηρετητές**
(Multiple servers ή super peers)

Μοντέλο με πολλούς εξυπηρετητές όπου μπορεί να είναι super peers εάν είναι και πελάτες ταυτόχρονα. Μηχανισμός που υλοποιείται πλέον και από το πρόγραμμα ανταλλαγής αρχείων kazaα. (εικ.g)

3.5 Το καθαρό Peer to Peer μοντέλο

Λειτουργεί με αρκετά πιο περίπλοκο τρόπο από ένα υβριδικό μοντέλο κάτι που φαίνεται και στο επίπεδο χρήσης της εφαρμογής. Το Gnutella βασίζεται στην "ιογενή εξάπλωση" κατά την οποία κάθε peer που δέχεται ένα μήνυμα το προωθεί και στους υπόλοιπους χρήστες που είναι συνδεδεμένοι με αυτόν.

Από έναν και μόνο peer έχουμε εικόνα όλου του δικτύου. Κάποιοι peers έχουν την δυνατότητα να συμπεριφέρονται ταυτόχρονα και ως clients αλλά και ως servers, αυτοί λέγονται ομότιμοι και είναι σε θέση να ανταλλάσσουν μηνύματα μεταξύ τους όπως π.χ. μηνύματα αναζήτησης.

Αυτό που, τελικά, κρίνει ποιο από τα δύο μοντέλα θα ακολουθηθεί κατά την υλοποίηση ενός Peer to Peer δικτύου είναι τα χαρακτηριστικά που επιθυμούμε να έχει το δίκτυο αυτό καθώς έχουμε να κάνουμε με δύο διαφορετικές προσεγγίσεις ως προς την υλοποίηση των Peer to Peer υπηρεσιών.

3.6 Πλεονεκτήματα του υβριδικού μοντέλου

Στοιχεία που έχουν τα δίκτυα με υβριδικό μοντέλο:

- Απλούστερα πρωτόκολλα επικοινωνίας και αποτελεσματικότερη διαχείριση
- Μεγαλύτερες ταχύτητες απόκρισης, όσον αφορά στην επικοινωνία peer και server

Φανταστείτε ένα μήνυμα στο δίκτυο της Gnutella, το οποίο θα πρέπει να διασχίσει αρκετούς peers που είναι συνδεδεμένοι με απλό modem και οι οποίοι κατεβάζουν ταυτόχρονα κάποια αρχεία, μέχρι να φτάσει στο προορισμό του, σε αντίθεση με μία απευθείας σύνδεση με έναν κεντρικό server του Napster.

- Αξιοπιστία σε real – time εφαρμογές οι οποίες είναι ευαίσθητες στις καθυστερήσεις
- Αποδοτικές αναζητήσεις

Οι απαντήσεις του server είναι πιο έγκυρες και πιο περιεκτικές συγκριτικά με άλλα queries αποκεντρωμένων αρχιτεκτονικών.

- Ασφάλεια

Μπορεί να γίνει ταυτοποίηση των κόμβων και έλεγχος της διακινούμενης πληροφορίας.

3.7 Μειονεκτήματα του υβριδικού μοντέλου

Από την άλλη πλευρά όσο διευκολύνει το σύστημα η ύπαρξη κάποιων κεντρικών διακομιστών άλλο τόσο μπορεί να αποτελέσει μεγάλο **μειονέκτημα** του υβριδικού μοντέλου, αφού στην περίπτωση που διακοπεί η καλή λειτουργία κάποιου απο τους κεντρικούς διακομιστές βρίσκεται εκτεθειμένο ολόκληρο το σύστημα.

Οι εφαρμογές του υβριδικού μοντέλου:

- *Είναι ευάλωτες σε Denial of Service επιθέσεις,*
- *Έχουν τρωτά σημεία (points of failure)*
- *Η υποκλοπή πληροφοριών από έναν server είναι καταστροφικό για το μεγαλύτερο πλήθος χρηστών.*

3.8 Πλεονεκτήματα του καθαρού μοντέλου

Η ανωνυμία πληροφορίας και χρηστών καθώς και η ανεξαρτησία από κεντρικά σημεία ή servers συγκαταλέγεται στα **πλεονεκτήματα των καθαρών Peer to Peer δικτύων**, αλλά ακόμα δεν είναι σε θέση να προσφέρουν την ταχύτητα και την ασφάλεια που χρειάζονται όλες οι υπηρεσίες.

Κάποια επιπλέον **πλεονεκτήματα** είναι

- Η αυτοργάνωση,
- Ο καταμερισμός φορτίου,
- Η ανοχή σε λάθη (fault tolerance)
- Η προσαρμοστικότητα.

3.9 Μειονεκτήματα του καθαρού μοντέλου

Τα **Μειονέκτηματα** ενός καθαρού μοντέλου Peer to Peer είναι

- Η διαχείριση,
- Η αποδοτικότητα των αναζητήσεων και του δικτύου,
- Η ασφάλεια χρηστών και πληροφορίας
- Γενικότερα τα πλεονεκτήματα των υβριδικών αρχιτεκτονικών.

Πάντα βέβαια τον σημαντικότερο ρόλο στην απόφαση για την επιλογή κάποιας αρχιτεκτονικής παίζουν τα χαρακτηριστικά των υπηρεσιών που θέλουμε προσφέρουμε.

ΚΕΦΑΛΑΙΟ 4

4. Γιατί να χρησιμοποιήσει κανείς το μοντέλο Peer to Peer

Το P2P μοντέλο κάνει τους χρήστες αυτόνομους.

Το μοντέλο αυτό έχει πολλά πλεονεκτήματα για τους χρήστες και τους οργανισμούς που το χρησιμοποιούν. Τα πλεονεκτήματα τα διαχωρίζουμε σε δύο κατηγορίες. **Τεχνικά** πλεονεκτήματα και **κοινωνικά**.

4.1 Τεχνικά πλεονεκτήματα

- Το κυριότερο πλεονέκτημα όλων είναι πως το σύστημα παράγει μόνο του την υπερβολικά μεγάλη υπολογιστική ισχύ που χρειάζεται μέσα από τους χρήστες του συστήματος κι έτσι το σύστημα με αυτόν τον τρόπο ξεπερνάει το πρόβλημα της μίας πηγής. Ο χρήστης δεν έχει ένα σημείο που θα εξυπηρετηθεί για να κάνει την δουλειά του οπότε δεν υπάρχει και φόβος το σύστημα να πληγεί ξαφνικά. Πάντα θα υπάρχει κάποιος υπολογιστής διαθέσιμος να εξυπηρετήσει το “query” που έμεινε σε εκκρεμότητα από αυτόν που είχε το πρόβλημα.
- Μπορεί να χρησιμοποιηθεί για την διανομή δεδομένων και ελέγχου ταυτόχρονα με τη χρήση κατανομής φόρτου (load balancing) κατά μήκος του δικτύου.
- Επιτρέπει την απευθείας πρόσβαση σε σημεία διανομής πόρων. Αυτό μπορεί να επιτρέψει επίσης τον απομακρυσμένο έλεγχο.

4.2 Κοινωνικά πλεονεκτήματα

- Οι χρήστες μπορούν πολύ εύκολα να σχηματίσουν τις δικές τους αυτόνομες Online κοινότητες σε ένα σημείο του δικτύου και να τις χρησιμοποιούν με τους κανόνες τους οποίους θα επιλέξουν.
- Οι κοινότητες αυτές ανάλογα με το επίπεδο χρήσης και εφαρμογής μπορούν να αλλάξουν ακόμη και τους τρόπους συμπεριφοράς των χρηστών στο δίκτυο.
- Κάποιοι χρήστες νιώθουν πιο άνετα όταν δεν υπάρχει κάποιος κεντρικός έλεγχος σε αυτά που κάνουν και έχουν όσο το δυνατόν τις λιγότερες πιθανότητες να πληγεί το δίκτυο τους.

ΚΕΦΑΛΑΙΟ 5

5. Χρήσιμες εφαρμογές για δίκτυα P2P

Τα δίκτυα p2p αναπτύσσονται κατάλληλα, σύμφωνα με τον σκοπό που πρόκειται να εξυπηρετήσουν. Αυτό έχει σαν συνέπεια, την ανάπτυξη δικτύων p2p για τις περισσότερες, αν όχι όλες τις κατηγορίες που μπορεί να εξυπηρετήσει. Οι κυριότερες από αυτές είναι:

- **Ανταλλαγή αρχείων**

Οι χρήστες του δικτύου μπορούν να ανταλλάσσουν μεταξύ τους αρχεία, χωρίς να είναι απαραίτητη η παρουσία ενός κεντρικού server για την αποθήκευσή τους

- **Διανεμημένος υπολογισμός**

Πολλοί υπολογιστές συνδεδεμένοι μεταξύ τους, χρησιμοποιώντας ο καθένας ένα μικρό μέρος από την υπολογιστική του ισχύ, επιτυγχάνουν καλύτερη απόδοση και σε χρόνο αλλά και σε ισχύ, σε σχέση με έναν υπερ-υπολογιστή ο οποίος θα κόστιζε και πολύ ακριβά. Απομακρυσμένη διαχείριση ή συνεργασία

- **Επικοινωνία με εικόνα και ήχο**

Εφαρμογές οι οποίες επιτρέπουν την επικοινωνία πολλών υπολογιστών απευθείας μεταξύ τους.

- **e-Business**

Τα P2P δίκτυα βοήθησαν πολύ στην εξέλιξη των επιχειρήσεων, βοηθώντας στην οργάνωση αλλά και τη διανομή καταναμημένων πληροφοριών, δεδομένων ή λογισμικού.

- **Online Παιχνίδια**

Οι υποδομές P2P επιτρέπουν στους προγραμματιστές ηλεκτρονικών παιχνιδιών να επικεντρώνονται στην ανάπτυξη του παιχνιδιού και όχι στην υποδομή των δικτύων, καθώς τα δίκτυα P2P προσφέρουν εύκολες και ολοκληρωμένες λύσεις για την ανάπτυξη και την υποστήριξη online κοινοτήτων, καθώς δεν χρειάζεται να ελέγχονται κεντρικά.

- **Ομαδική αντιμετώπιση Ιών**

Οι συνεργασίες μεταξύ των κόμβων σε ένα P2P δίκτυο καθιστά εφικτή τη διαδικασία ανίχνευσης και πρόληψης ιών, καθώς υπάρχουν αυτοματοποιημένες διαδικασίες για την αντιμετώπισή αλλά και την μελλοντική επίθεση από αυτούς.

- **Υπηρεσίες online Conference και e-Learning**

Εάν είναι επιθυμητό, ο server μπορεί να τοποθετήσει τα δεδομένα πιο κοντά στον client, με σκοπό την ευκολότερη πρόσβαση σε αυτά από τον τελευταίο.

- **Ομαδικό Development απλών αλλά και σύνθετων εφαρμογών**

Από τις πιο απλές εφαρμογές που μπορούν να αναπτυχθούν από μια ομάδα σε ένα project, μέχρι πολύπλοκες διαδικασίες επεξεργασίας και 3D rendering, είναι εφικτό με την χρήση P2P δικτύων.

- **Διανεμημένη Αποθήκευση**

Με βάση αυτή την αρχιτεκτονική, όλα τα μέλη μια ομάδας του δικτύου διαθέτουν ένα μέρος από τον αποθηκευτικό τους χώρο με σκοπό την κοινόχρηστη χρήση του από το υπόλοιπο δίκτυο. Αυτό γίνεται με σκοπό την επίτευξη όσο το δυνατόν μεγαλύτερου αποθηκευτικού, με το ελάχιστο κόστος. Το ίδιο ισχύει όπως είναι λογικό και για το κόστος συντήρησης.

Το γεγονός όμως ότι τέτοιες τεχνικές εφαρμόζονται κυρίως σε εργασιακό περιβάλλον, καθιστά το δίκτυο ευάλωτο και εν δυνάμει προσβάσιμα από τον καθένα.

Ακόμα και με την κωδικοποίηση των δεδομένων με πολύπλοκους αλγόριθμους, τα δεδομένα δεν είναι 100% ασφαλή από θέμα προσβασιμότητας.

Η ασφάλεια που εγγυάται ένα τέτοιο σύστημα είναι το γεγονός ότι αντίγραφα των δεδομένων είναι αποθηκευμένα σε πολλά σημεία του δικτύου, καθιστώντας έτσι δύσκολη την απώλειά τους, αυξάνοντας όμως την πολυπλοκότητα των πρωτοκόλλων επικοινωνίας.

- **Μηχανές Αναζήτησης**

Εξαιτίας της αμεσότητας που έχουν οι χρήστες να αλληλεπιδρούν με τα δεδομένα που υπάρχουν και διανέμονται στο δίκτυο, είναι λογικό πως οι πληροφορίες στις μηχανές αναζήτησης ανανεώνονται σχεδόν σε πραγματικό χρόνο.

- **Intelligent Agents**

Οι πρώτες προσπάθειες που ξεκίνησαν πριν αρκετά χρόνια για ένα μοντέλο υπηρεσιών, όπου οι πράκτορες συνεργάζονται μεταξύ τους σε ένα P2P δίκτυο στέφθηκαν με επιτυχία. Με τον όρο πράκτορες εννοούμε διεργασίες που εκτελούνται σε έναν κόμβο και διανέμουν μια πληροφορία, ενώ επίσης μπορούν να ξεκινήσουν μία διεργασία, μετά από εντολή ενός αντίστοιχου δικτύου.

Ακόμη, έχουν τη δυνατότητα να αλλάζουν την προτεραιότητα και τη ροή των εργασιών, να αναλαμβάνουν την προστασία των δικτύων από ιούς, ακόμη και την οργάνωση μηχανών αναζήτησης με σκοπό την μεταξύ τους επικοινωνία.

- **P2P Collaboration**

Στην κατηγορία αυτή ανήκουν μικρές αλλά και μεγάλες εταιρίες οι οποίες διατηρούν ομάδες που εργάζονται σε απόσταση από την έδρα τους και σε μεγάλη διασπορά, όπως ερευνητικές ομάδες αλλά και εργαζόμενοι οι οποίοι χρειάζεται να μετακινούνται συχνά. Σύμφωνα με αυτό το μοντέλο, τα μέλη μιας τέτοιας ομάδας μπορούν να ανταλλάσσουν μεταξύ τους πληροφορίες, να μοιράζονται αρχεία και να επικοινωνούν μεταξύ τους σε πραγματικό χρόνο, ή ακόμα και με τη χρήση αλληλογραφίας και κοινών εφαρμογών.

Το μειονέκτημα είναι προβλήματα που προκύπτουν με κυριότερο το θέμα της ασφάλειας. Παρόλα αυτά, η συγκεκριμένη αρχιτεκτονική έχει κινήσει το ενδιαφέρον αγορά με πολλά μεγάλα ονόματα του επιχειρηματικού και όχι μόνο χώρου να έχουν δείξει το ενδιαφέρον τους, με πιο χαρακτηριστικό το Υπουργείο Δικαιοσύνης των ΗΠΑ.

- **Remote Services**

Επιχειρήσεις που επιθυμούν να εξυπηρετήσουν αποτελεσματικότερα τους πελάτες τους μπορούν μεταφέροντας τα δεδομένα τους σε τοπικά υποκαταστήματα, να τα αποθηκεύουν εκεί και να μην είναι απαραίτητη η χρήση ενός κεντρικού server για την διανομή τους.

- **Κοινότητες Δικτύων**

Ομάδες με κοινά ενδιαφέροντα, δημιουργούν το δικό τους δίκτυο ώστε να μπορούν να επικοινωνούν.

- **Εικονικά και Κατανεμημένα Ιδιωτικά Δίκτυα**

Η πρώτη κατηγορία σύμφωνα με τον τίτλο αφορά τα εικονικά δίκτυα τα οποία αποτελούν υποσύνολα ενός μεγαλύτερου δικτύου ή του Διαδικτύου, τα οποία όμως θα συμπεριφέρονται σαν να είναι αυτόνομα και ιδιωτικά με ελεγχόμενη πρόσβαση. Αυτό που τα διαφοροποιεί σε σχέση με τα VPNs είναι η αρχιτεκτονική. Με βάση αυτή, δεν υπάρχουν IP διευθύνσεις και τοποθεσίες αρχείων, αλλά όλα τα μέλη θα μοιράζονται τη δομή και τα δεδομένα του δικτύου, σε μια διαφορετική προσέγγιση.

Η δεύτερη κατηγορία βασίζεται απλή ιδέα ότι εάν ένας αριθμός x χρηστών σε ένα σημείο του πλανήτη, ζητήσει το ίδιο ακριβώς δεδομένο με άλλους χρήστες από το ίδιο σημείο, το δεδομένο θα πρέπει να αποσταλεί x φορές. Σε ένα κατανεμημένο δίκτυο όμως η μεταφορά θα γινόταν λιγότερες φορές, εφόσον ο web server δεν θα έπρεπε να απαντήσει στην ίδια αίτηση πολλές φορές, αναζητώντας αντίγραφα σε κοντινότερους από αυτόν χρήστες.

Αν και ακούγεται αρκετά πρωτοπόρο, το μοντέλο αυτό δεν είναι εύκολα εφαρμόσιμο καθώς πλέον υπάρχουν λιγότερες στατικές σελίδες και οι χρήστες δεν επιθυμούν τα προσωπικά τους δεδομένα να είναι προσβάσιμα από όλους με τους οποίους ανήκουν στο ίδιο δίκτυο, ίσως όμως σε μία μελλοντική έκδοση του Διαδικτύου να ήταν χρήσιμο.

5.1 Τεχνολογία peer-to-peer

Αντίθετα με την αρχιτεκτονική client-server, τα δίκτυα P2P εκμεταλλεύονται πλήρως τον αποθηκευτικό χώρο αλλά και το αναξιοποίητο εύρος ζώνης, καθώς επίσης και την υπολογιστική ισχύ που δεν καταναλώνεται προς κάποιο όφελος.

Παράλληλα, δεν τα δίκτυα P2P δεν στηρίζονται σε έναν κεντρικό υπολογιστή για την παροχή και την πρόσβαση σε υπηρεσίες και λειτουργούν κυρίως έξω από το DNS. Σκοπός του δικτύου είναι να αποφεύγει την συγκεντρωμένη οργάνωση και χρησιμοποιώντας μία ιδιαίτερη αρχιτεκτονική.

Οι αποφάσεις στο σύστημα λαμβάνονται με αποκεντρωμένο τρόπο, καθώς οι υπολογιστές που ανήκουν στο δίκτυο επικοινωνούν περιοδικά μεταξύ τους, γεγονός που τους επιτρέπει να λειτουργούν και σαν client αλλά και ως server. Όλα τα παραπάνω αποτελούν μια εν συντομία περιγραφή των βασικών χαρακτηριστικών της τεχνολογίας P2P.

5.2 Μειονεκτήματα

Όπως όλες οι τεχνολογίες, έτσι και τα P2P δίκτυα, έχουν το αδύνατό τους σημείο. Ένα από αυτά είναι η διανεμημένη μορφή καναλιών επικοινωνίας στα δίκτυα αυτά, γεγονός το οποίο οδηγεί σε υπηρεσίες μη ντετερμινιστικής φύσης. Πιο απλά, πελάτες που αναζητούν το ίδιο δεδομένο από το δίκτυο μπορεί να συνδεθούν με διαφορετικές μηχανές αναζήτησης, με αποτέλεσμα τη λήψη διαφορετικής πληροφορίας.

Ακόμη, αιτήσεις μέσω ενός P2P δικτύου μπορεί να μην έχουν άμεση απάντηση ή να μην λάβουν και ποτέ. Τέλος, οι πόροι σε ένα P2P δίκτυο μπορεί να μην εμφανίζονται για μεγάλα διαστήματα, καθώς εάν ο χρήστης που τους διαθέτει αποσυνδεθεί και είναι ο μοναδικός που τα διαθέτει, τότε δεν υπάρχει άλλος τρόπος ανάκτησής τους.

Γεγονός το οποίο έρχεται σε αντιπαράθεση με τους όρους του Διαδικτύου, όπου θέλει την πληροφορία να είναι ανα πάσα στιγμή διαθέσιμη και προσβάσιμη. Παρά τα μειονεκτήματα όμως που προαναφέραμε, τα δίκτυα P2P διαθέτουν μηχανισμούς για να τα εξαλείψουν, όπως για παράδειγμα την εφαρμογή mirrors για την αδιάκοπη προσφορά των δεδομένων.

Όπως καταλαβαίνουμε, το σύστημα χρησιμοποιεί την ίδια τη δομή του για να αντιμετωπίσει ένα πρόβλημα που προκλήθηκε στο δίκτυο.

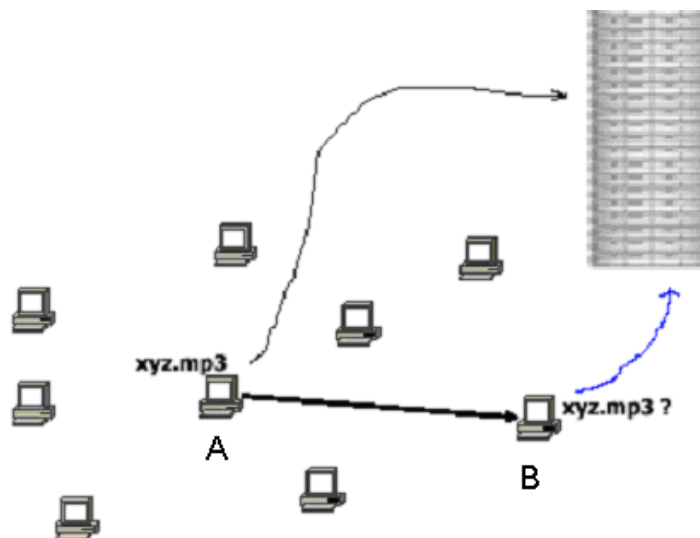
ΚΕΦΑΛΑΙΟ 6

6. Κατηγορίες Peer to Peer δικτύων

6.1 Κεντριοποιημένα P2P συστήματα

Κεντριοποιημένα συστήματα εννοούμε την μία απ' τις δύο κατηγορίες Peer to Peer εφαρμογών που έχουμε αναφέρει. Αυτού του είδους τα δίκτυα βασίζονται σε κάποιο κεντρικό σημείο στο οποίο διατηρούν κατάλογο και πληροφορίες για τους χρήστες όπως και για όλα τα δεδομένα που ανταλλάσσουν οι χρήστες και έχουν αποθηκευμένα στο σύστημα τους. Ο ένας τρόπος να δημιουργηθεί κατάλογος ευρετήριο είναι με την συνεργασία των κόμβων. Οι ίδιοι οι κόμβοι παρέχουν στο σύστημα τακτικά μία λίστα των δεδομένων που εισάγουν στο δίκτυο

Ο άλλος τρόπος είναι με αναζήτηση στο δίκτυο, όπως ακριβώς γίνεται και με μια αναζήτηση στο Internet. (Σχήμα 2.1)



Σχήμα 2.1:

Διαδικασία αναζήτησης

Όταν ο κόμβος A που προσφέρει το δεδομένο “xyz.mp3” εισάγεται στο δίκτυο ενημερώνει τον κατάλογο-ευρετήριο με τα δεδομένα που προσφέρει. Ένας χρήστης-κόμβος που θέλει να βρει ένα συγκεκριμένο δεδομένο στο δίκτυο, για παράδειγμα στο Σχήμα 2.1 ο κόμβος B που επιθυμεί το “xyz.mp3”, κάνει μια ερώτηση στο κεντρικό ευρετήριο κι αυτό μετά από μία τοπική αναζήτηση στις εγγραφές του επιστρέφει ως απάντηση στον κόμβο B την ακριβή τοποθεσία του δεδομένου μέσα στο δίκτυο.

Τέλος, ο χρήστης-κόμβος B ζητά απευθείας από τον κόμβο A που διαθέτει το δεδομένο να του το στείλει.

Πάρολο δηλαδή που συστήματα τέτοιας κατηγορίας, όπως το Napster, είναι μοντέλα ομότιμων κόμβων ως προς την μεταφορά των δεδομένων στο δίκτυο, ο εντοπισμός τους γίνεται κεντροποιημένα.

Τα P2P δίκτυα έγιναν ευρέως γνωστά και ίσως μία από τις πιο γρήγορα εξελισσόμενες εφαρμογές του διαδικτύου. Σε αυτό σπουδαίο ρόλο έπαιξε το Napster. Πλέον κανένα δίκτυο της συγκεκριμένης κατηγορίας δεν έχει επιβιώσει μέχρι τώρα και αυτό κυρίως λόγω νομικών προβλημάτων.

Στα πλεονεκτήματα τους συμπεριλαμβάνονται η αποδοτικότητα του δικτύου αφού ένα ερώτημα μόνο αρκεί για να απαντηθεί η ερώτηση ενός κόμβου, ο γρήγορος εντοπισμός σε δύσκολα δεδομένα και η υποστήριξη των “partial-match” ερωτήσεων (ερωτήσεις που περιλαμβάνουν ορθογραφικό λάθος ή υποσύνολο από λέξεις κλειδιά).

Τα μειονέκτηματα όμως της ξάφνικης κατάρρευσης του δικτύου λόγω της οποιασδήποτε βλάβης τυχόν πάθει ο κεντρικός κόμβος, και η δυσκολία για την σωστή ενημέρωση του κεντρικού καταλόγου και ευρετηρίου είναι καθοριστικό για την εγκατάλειψη τέτοιου είδους εφαρμογών.

6.1.1 Αρχιτεκτονική πελάτη-εξυπηρετητή (client – server)

Στην αρχιτεκτονική αυτή συμμετέχουν δύο ρόλοι, **οι πελάτες** και οι **εξυπηρετητές**, που ζητούν υπηρεσίες και που προσφέρουν υπηρεσίες αντίστοιχα. Μάλιστα οι περισσότερες υπηρεσίες διαδικτύου διανέμονται μέσω αυτής της αρχιτεκτονικής που θεωρείται πλέον παραδοσιακή.

Ο εξυπηρετητής χρησιμοποιεί κάποιο πρωτόκολλο βάση του οποίου επιτρέπεται να τον προσεγγίσει κάποιος άλλος κόμβος-πελάτης(client) ξεκινώντας την διαδικασία αναζητώντας με αίτημα πρώτος κάποιο συγκεκριμένο δεδομένο. Την περισσότερη δουλειά στο κομμάτι της προσφοράς μιας υπηρεσίας το έχει συνήθως ο ρόλος του εξυπηρετητή. Με το πρωτόκολλο μεταφοράς αρχείων(FTP) συνηθίζουν να λειτουργούν οι πιο δημοφιλείς εφαρμογές διαδικτύου.

Για παράδειγμα ένας πελάτης φορτώνει ένα αρχείο στον FTP εξυπηρετητή, έπειτα πολλοί πελάτες το μεταφορτώνουν από αυτόν, χωρίς να είναι αναγκαία η απευθείας σύνδεση των χρηστών που μεταφορτώνουν και αυτών που φορτώνουν.

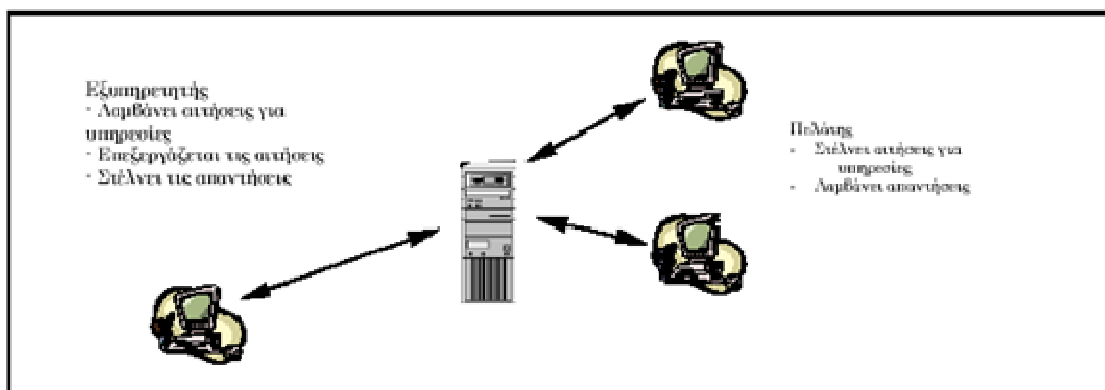
6.1.2 Πλεονεκτήματα - Μειονεκτήματα

Το **πλεονέκτημα** που συναντάμε σε αυτήν την αρχιτεκτονική είναι η λίγη υπολογιστική ισχύς που χρειάζεται από μέρους των clients-πελατών. Συμβαίνει πλέον οι αναβαθμίσεις στα συστήματα των χρηστών να έχουν ξεπεράσει κατά πολύ τις δημοφιλέστερες εφαρμογές διαδικτύου(ανάκτηση ηλεκτρονικού ταχυδρομείου, μεταφορά αρχείων και περιαγωγή στο διαδίκτυο).

Από την άλλη πλευρά το σπουδαίο **μειονέκτημα** της εφαρμογής αυτής είναι πως δεν μπορεί να ανταποκριθεί στην συνεχόμενη αύξηση των πελατών του ένας εξυπηρετητής με αποτέλεσμα να φτάνει στο σημείο να μην μπορεί να διαχειριστεί νέους πελάτες.

Οι πελάτες είναι σε θέση να αιτούνται στον εξυπηρετητή, χωρίς όμως να προσφέρουν και αυτοί κάποια υπηρεσία σε άλλο πελάτη, διατηρώντας έναν παθητικό ρόλο. Οι εταιρείες λοιπόν για να εξισορροπήσουν το φορτίο μοιράζουν την κίνηση κατάλληλα στους εξυπηρετητές (load balancing) και ανανεώνουν τους υπάρχοντες πόρους αναβαθμίζοντας συσκευές και επεκτείνοντας το εύρος ζώνης.

Εάν όλες οι μηχανές στο δίκτυο έτρεχαν και ως εξυπηρετητές και ως πελάτες, θα διαμόρφωναν την αρχή ενός στοιχειώδους P2P δικτύου.



Σχήμα 2.1 Μοντέλο πελάτη - εξυπηρετητή (client – server)

6.2 Μη κεντριοποιημένα P2P συστήματα

Είναι η δεύτερη κατηγορία των P2P συστημάτων η οποία χωρίζεται σε **δομημένα** και **μη δομημένα** δίκτυα. Η διαφορά αυτών των συστημάτων με τα κεντριοποιημένα συστήματα είναι ότι εδώ δεν υπάρχει κάποιο κεντρικό ευρετήριο-κατάλογος.

6.2.1 Δομημένα P2P συστήματα

Το CAN και το CHORD είναι ίσως τα πιο γνωστά αυστηρώς δομημένα συστήματα τα οποία στηρίζονται σε κατανεμημένους πίνακες κατακερματισμού (distributed hash tables- DHTs). Γενικότερα τα συστήματα αυτά λειτουργούν με κάποιους κανόνες όσον αφορά τις συνδέσεις μεταξύ κόμβων. Επιπλέον τα δεδομένα δεν τοποθετούνται τυχαία σε αυτούς αλλά σε προκαθορισμένες τοποθεσίες.

6.2.1.1 Δομές δικτύων και δρομολόγηση

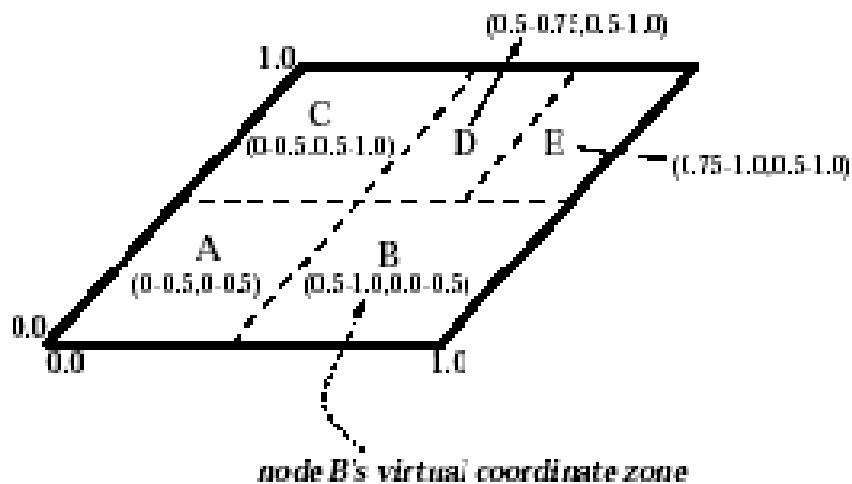
Εδώ θα αναλύσουμε τον τρόπο δρομολόγησης των CAN και CHORD
Όπως είπαμε και προηγουμένως είναι οι πιο γνωστές δομές δομημένων δικτύων.

6.2.1.2 CAN (Content-Addressable Network)

Το CAN λειτουργεί και φέρεται σαν ένας πίνακας κατακερματισμού ο οποίος αντιστοιχεί κλειδιά ("keys") σε τιμές ("values"). Είναι κατανεμημένο δίκτυο με ανεξάρτητους κόμβους ο καθένας από τους οποίους χρεώνεται ένα κομμάτι ("zone") από τον πίνακα κατακερματισμού και αποθηκεύει πληροφορίες για τους γειτονικούς του κόμβους.

Έχει τρεις βασικές λειτουργίες 1.εισαγωγή 2.αναζήτηση και 3.διαγραφή δεδομένων. Ένα εικονικό σύστημα συντεταγμένων d με τη μορφή ενός d -torus το οποίο μπορεί να το συναντήσουμε και με τον όρο "reality" φροντίζει για την οργάνωση του δικτύου. Ανά πάσα στιγμή ο κάθε κόμβος κατέχει την δική του ανεξάρτητη ζώνη στο σύστημα συντεταγμένων.

Έτσι στο Σχήμα 3.1 ο δισδιάστατος χώρος διαμερίζεται σε πέντε κόμβους και για παράδειγμα ο κόμβος B κατέχει το κομμάτι-ζώνη με συντεταγμένες $(0.5-1.0, 0.0-0.5)$. Το σύστημα συντεταγμένων αποθηκεύει ζεύγη (κλειδί, τιμή) όπου το κλειδί κάθε ζεύγους αντιστοιχίζεται σε ένα σημείο P αυτού του χώρου με μια ομοιόμορφη συνάρτηση κατακερματισμού. Τελικά ένα ζεύγος αποθηκεύεται στον κόμβο που κατέχει τη ζώνη με το σημείο P στο οποίο αντιστοιχίζεται το κλειδί του.



Σχήμα 3.1: Παράδειγμα ενός χώρου 2-διαστάσεων με πέντε κόμβους.

Σε ένα δίκτυο σαν κι αυτό που συνεχώς υπάρχουν εισαγωγές νέων κόμβων η οργάνωση έχει ως εξής. Ο κόμβος που επιθυμεί να εισέλθει στο δίκτυο επιλέγει ένα σημείο P στον χώρο τυχαία, ενώ άμεσα στέλνει ένα μήνυμα μέσω του κόμβου που γνωρίζει και είναι ήδη στο δίκτυο, στον κόμβο που κατέχει την ζώνη με το σημείο P.

Η διαδικασία συνεχίζει όταν λαμβάνει το μήνυμα στον τελευταίο κόμβο ο οποίος τώρα θα διαιρέσει την ζώνη που κατέχει σε δύο μέρη κατά μήκος μιας διάστασης και πλέον ο νέος κόμβος αναλαμβάνει συγκεκριμένο χώρο και συγκεκριμένα δεδομένα που είναι αποθηκευμένα εκεί. Επιλέον ενημερώνονται οι γείτονες των κόμβων που ενεπλάκησαν στον διαχωρισμό της ζώνης με τα νέα δεδομένα.

Η ίδια συνάρτηση κατακερματισμού που ακολουθείται στην περίπτωση του διαχωρισμού ζώνης εφαρμόζεται και στην περίπτωση που κάποιος κόμβος θέλει να ανακτήσει κάποιο δεδομένο.

6.2.1.3 Chord

Όσον αφορά το Chord, είναι ένα κατανεμημένο πρωτόκολλο το οποίο στοχεύει στην λύση του προβλήματος της αποδοτικής τοποθέτησης των κόμβων οι οποίοι κατέχουν ένα αντικείμενο. Με βάση ένα κλειδί, το Chord το αντιστοιχεί σε έναν κόμβο και με την εφαρμογή του κατακερματισμού εξισορροπεί το φορτίο, εφ'όσον κάθε κόμβος δέχεται τον ίδιο σχεδόν αριθμό κλειδιών. Σε σύστημα με X κόμβους, κάθε κόμβος διατηρεί περίπου $O(\log X)$ αριθμό πληροφοριών για άλλους κόμβους και τις χρησιμοποιεί εν συνεχεία για την δρομολόγηση.

6.2.1.4 Replication – Δημιουργία Αντιγράφων

Στο δίκτυο CAN διακρίνουμε 4 τρόπους δημιουργίας αντιγράφων:

1. Στην έκδοση του CAN με πολλαπλά realities είναι δυνατή η δημιουργία μόνο ενός αντιγράφου σε κάθε ένα.
2. Στην έκδοση του CAN όπου υπάρχουν πολλοί peers σε κάθε ζώνη, είναι δυνατή η δημιουργία αντιγράφων σε όλους τους κόμβους της ζώνης.
3. Σε περιπτώσεις όπου χρησιμοποιούνται χ συναρτήσεις κατακερματισμού, αντιστοιχεί ένα κλειδί σε χ διαφορετικά σημεία. Άρα, είναι δυνατή η δημιουργία αντιγράφων στους κόμβους ο οποίοι περιέχουν τα χ σημεία.
4. Εάν ένας κόμβος δέχεται πολλές αιτήσεις για ένα συγκεκριμένο αίτημα, με σκοπό να του αναλογεί μεγάλο φόρτο εργασίας, είναι δυνατή η δημιουργία αντιγράφων σε γειτονικούς του κόμβους, έτσι ώστε να δημιουργείται μια περιοχή γύρω του με πολλαπλά αντίγραφα.

Όσον αφορά τα δίκτυα CHORD, υπάρχει μόνο ένας τρόπος δημιουργίας αντιγράφων του ζεύγους κλειδί – τιμή. Αυτό επιτυγχάνεται με την χρήση successor-list, ένα αντίγραφο μπορεί να δημιουργηθεί σε όλους τους διαδόχους του κόμβου που ευθύνεται για το συγκεκριμένο κλειδί.

6.2.2 Μη δομημένα P2P συστήματα

Όταν ένα δίκτυο σχηματίζεται από μη δομημένα συστήματα ομότιμων κόμβων, τότε το δίκτυο αυτό δεν έχει κάποια συγκεκριμένη τοπολογία. Το γεγονός αυτό, καθιστά την αναζήτηση δύσκολη υπόθεση, καθώς εφαρμόζονται μόνο γενικοί αλγόριθμοι αναζήτησης. Στην κατηγορία αυτή όπως είναι φυσικό ανήκουν και τα πρώτα συστήματα που κατασκευάστηκαν, χωρίς να είναι απόλυτο ότι όλα είναι κεντροποιημένα.

Τα συστήματα αυτά, δεν είναι δομημένα όσον αφορά την οργάνωση των κόμβων αλλά και τον τρόπο με τον οποίο τα δεδομένα τοποθετούνται στο δίκτυο. Οι τοπικές αντιδράσεις καθορίζουν και τη συμπεριφορά ολόκληρου του δικτύου.

Η πιο κοινή μέθοδος αναζήτησης δεδομένων είναι η λεγόμενη πλημμύρα, δηλ. τη μετάδοση της ερώτησης σε όλους τους γειτονικούς κόμβους μέχρι την προκαθορισμένη ακτίνα.

6.2.2.1 Μέθοδοι Αναζήτησης

Υπάρχουν 2 τεχνικές αναζήτησης σε ένα μη δομημένο P2P δίκτυο. Οι τεχνικές αυτές, δεν ορίζουν τον τρόπο αποθήκευσης των δεδομένων και των αντιγράφων τους στο δίκτυο, όπως επίσης δεν έχουν το δικαίωμα να τροποποιήσουν την τοπολογία του δικτύου.

1. Τυφλή αναζήτηση

Η ερώτηση προωθείται σε έναν ικονοποιητικό αριθμό κόμβων μέχρι να βρεθούν τα επιθυμητά αποτελέσματα

2. Ένημερωμένη αναζήτηση

Η μετάδοση της ερώτησης βασίζεται σε συγκεκριμένες πληροφορίες ή στατιστικά

6.2.2.1.1 Μέθοδοι τυφλής αναζήτησης

Gnutella: Ο ίδιος αλγόριθμος εμφανίζεται και στο δίκτυο gnutella και χρησιμοποιεί τη τεχνική της πλημμύρας. Ο κόμβος ο οποίος επιθυμεί κάποιο δεδομένο, προωθεί την ερώτηση στους γειτονικούς τους κόμβους, οι οποίοι με τη σειρά τους κάνουν το ίδιο.

Η διαδικασία αυτή επαναλαμβάνεται μέχρι να βρεθεί το συγκεκριμένο δεδομένο σε κάποιον κόμβο ο οποίος δέχτηκε την ερώτηση, ή μέχρι να εξαντληθεί ο χρόνος ή η ακτίνα αναζήτησης που έχουν ορισθεί.

Παρόλο που ο αλγόριθμος είναι απλός, δεν βοηθά στη διατήρηση της κλιμάκωσης και επιβαρύνει σημαντικά το δίκτυο.

Modifies-BFS: Είναι μια παραλλαγή του παραπάνω αλγορίθμου και η μέθοδος αυτή μειώνει το μέσο όρο μηνυμάτων-ερωτήσεων, παρόλο που επικοινωνεί με ένα μεγάλο αριθμό κόμβων. Στην μέθοδο αυτή, κάθε κόμβος επιλέγει τυχαία έναν αριθμό γειτονικών κόμβων στους οποίους θα προωθήσει την ερώτηση που έχει δεχτεί.

Iterative Deepening: Η τεχνική αυτή ξεκινά με έναν αλγόριθμο BFS και με τελικό στόχο μία τιμή TTL και επαναλαμβάνεται αυξάνοντας το βάθος σε περίπτωση που η πρώτη εφαρμογή του BFS αποτύχει.

Σε περίπτωση που βρεθεί λύση με κάποιο κριτήριο τερματισμού ή πλημμύρα είναι μικρής έκτασης, τότε η τεχνική αυτή έχει πλεονεκτήματα, σε αντίθετη περίπτωση όμως επιβαρύνει σημαντικά το δίκτυο ακόμα και με τη χρήση απλής πλημμύρας.

Random Walks: Στον αλγόριθμο αυτό, κάθε κόμβος που δημιουργεί την ερώτηση, την προωθεί αυτόματα σε χ τυχαίους γείτονες και εν συνεχεία κάθε ένας από αυτούς επιλέγει ένα άλλο τυχαίο γειτονικό σύνολο κόμβων για την προώθηση της ερώτησης. Αυτό έχει ως αποτέλεσμα τη δημιουργία χ μονοπατιών προώθησης της ερώτησης, τα λεγόμενα walkers.

Για να τερματιστεί ένα μονοπάτι απαραίτητη προϋπόθεση είναι είτε να βρεθεί η συγκεκριμένη τιμή TTL, είτε με τη μέθοδο ελέγχου όπου κάθε walker ελέγχει ανα περιόδους τον κόμβο που έκανε την ερώτηση εάν ισχύ ακόμα το κριτήριο τερματισμού. Το δυνατό χαρακτηριστικό πλεονέκτημα του αλγορίθμου είναι η σημαντική μείωση των μηνυμάτων που διακινούνται στο δίκτυο, προκαλώντας επίσης ενός είδους εξισορρόπηση στο φορτίο του δικτύου τοπικά.

GUESS: Στον αλγόριθμο αυτό γίνεται χρήση των ultrapeers που είναι συνδεδεμένοι, οι οποίοι ενημερώνονται κάθε φορά που ένας κόμβος κάνει αναζήτηση και με τη σειρά τους κάνουν ερώτηση σε όλους τους απλούς κόμβους, επιλέγοντας παράλληλα και άλλους ultrapeers, γειτονικούς και μη, με σκοπό να βοηθήσουν στην αναζήτηση.

Gnutella2: Ο αλγόριθμος αυτός εφαρμόζεται σε περίπτωση που στο δίκτυο υπάρχουν super κόμβοι συνδεδεμένοι μεταξύ τους και κάθε ένας από αυτούς συνδέεται με έναν αριθμός από κόμβους φύλλα. Όταν ένας super κομβος δεχτεί μια ερώτηση από έναν κόμβο φύλλο, τότε την προωθεί και στους υπόλοιπους κόμβους φύλλα αλλά και στους γειτονικούς super κόμβους, οι οποίοι επεξεργάζονται την ερώτηση όχι μόνο τοπικά αλλά μεταδίδοντάς τη και σε δικούς τους κόμβους.

Μέθοδος Τυφλής Αναζήτησης	Επικοινωνία με όλους τους γείτονες	Χρήση υπερκόμβων	Βάθος αναζήτησης	Εύρεση όλων των αποτελεσμάτων στο βάθος αναζήτησης
Πλημμύρα	✓		Όλο το δίκτυο	✓
<i>Gnutella</i>	✓		TTL	✓
<i>Modified – BFS</i>			TTL	
<i>Iterative deepening</i>	✓		TTL	✓
<i>Random Walks</i>			Μέχρι να επιστραφεί ικανοποιητικός αριθμός αποτελεσμάτων	
<i>Guess</i>		✓		
<i>Gnutella2</i>		✓	1 (ένας υπερκόμβος)	

6.2.2.1.2 Μέθοδοι ενημερωμένης αναζήτησης

Intelligent BFS: Αποτελεί μια καινούργια έκδοση του αλγόριθμου Modified-BFS, στην οποία δεν υπάρχει πλέον η εκμετάλλευση ορισμένων στατιστικών για τη δρομολόγηση μιας ερώτησης. Κάθε κόμβος διατηρεί εγγραφές με πληροφορίες σχετικά με τις πρόσφατες ερωτήσεις που έχουν ανταποκριθεί γείτονες. Σε περίπτωση που ένας κόμβος θέλει να προωθήσει μια ερώτηση, αναζητά πρώτα στο ιστορικό των ερωτήσεων που έχει εξυπηρετήσει ο κόμβος, εάν ταιριάζει με κάποια η νέα ερώτηση και στη συνέχεια την προωθεί στο σύνολο των κόμβων που έδωσε τα πιο ικανοποιητικά αποτελέσματα στις προηγούμενες ερωτήσεις.

APS: Με βάση αυτό τον αλγόριθμο, κάθε κόμβος διατηρεί ένα τοπικό ευρετήριο στο οποίο υπάρχουν εγγραφές από κάθε δεδομένο που έχει αναζητήσει ένας γείτονας. Η εγγραφή αυτή αντιπροσωπεύει την πιθανότητα να επιλεγεί ο συγκεκριμένος γείτονας του κόμβου αυτή σε μία άλλη αναζήτηση για το ίδιο δεδομένο. Και σε αυτή την περίπτωση εφαρμόζεται η τεχνική των k ανεξάρτητων walkers με αποτέλεσμα κάθε ενδιαμέσος κόμβος να προωθεί την ερώτηση σε έναν γειτονικό, σύμφωνα πάντα με το τοπικό ευρετήριο, το οποίο ενημερώνεται κάθε φορά αυξάνοντας ή μειώνοντας τις πιθανότητες σε περίπτωση που ο walker επιτύχει ή αποτύχει αντίστοιχα.

Routing Indices (RIs): Ο βασικός άξονας στον οποίο στηρίζεται η συγκεκριμένη αρχιτεκτονική, είναι η δρομολόγηση των ερωτήσεων σε γειτονικούς κόμβους, όπου είναι πιο πιθανόν να υπάρξει απάντηση και ανήκει στην κατανεμημένη αναζήτηση με ευρετήριο σε κάθε κόμβο. Τα ευρετήρια δηλώνουν την κατεύθυνση που θα ακολουθήσει ένα δεδομένο και όχι την πραγματική του τοποθεσία. Αυτός είναι και ο λόγος για τον οποίο το μέγεθός του δεν αυξάνεται πολύ και είναι ανάλογο ως προς τον αριθμό των γειτόνων του.

Στην κατηγορία αυτή διακρίνονται 3 είδη ευρετηρίων τύπου RI:

1. *Compound CRI*
2. *Hop count HRI*
3. *Exponential ERI*

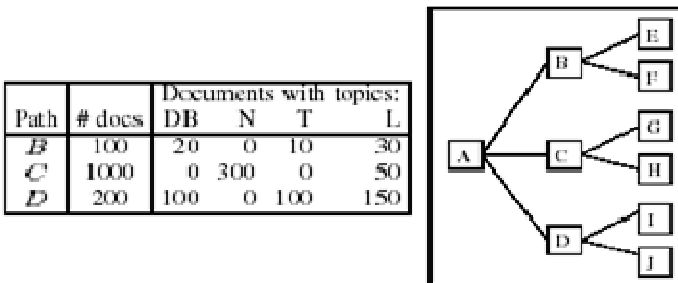
Με τον όρο RI εννοούμε μια δομή δεδομένων, η οποία σε μία ερώτηση επιστρέφει μια λίστα με ταξινομημένους γειτονικούς κόμβους σύμφωνα με τις πληροφορίες που διαθέτουν προς την ερώτηση και περιέχουν τον αριθμό των δεδομένων που είναι εφικτό να βρεθούν από τους γειτονικούς κόμβους.

Αυτό έχει σαν αποτέλεσμα κάθε κόμβος να διαθέτει ένα τοπικό ευρετήριο και ένα CRI το οποίο θα περιλαμβάνει τον αριθμό των δεδομένων τα οποία αποθηκεύονται σε κάθε μονοπάτι, καθώς επίσης και τον αριθμό των δεδομένων σε κάθε κατηγορία.

Σύμφωνα με την κάθε ερώτηση, ο κόμβος καλείται να επιλέξει τον καλύτερο γειτονικό κόμβο για να την προωθήσει. Η σχέση με την οποία αξιολογείται η ποιότητα ενός κόμβου είναι η εξής:

$$\chi_{\text{δεδομένων}} \times \prod_i (CRI(s_i) / \chi_{\text{δεδομένα}})$$

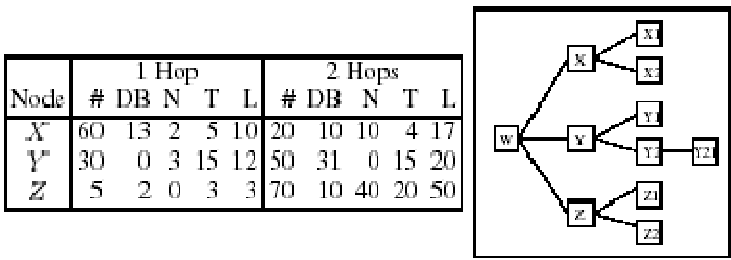
Όπως φαίνεται στο παρακάτω σχήμα, με τον μεταβλητή $(CRI(s_i))$ δηλώνουμε τον αριθμό των δεδομένων σε μία κατηγορία, που μπορούν να βρεθούν μέσω ενός γειτονικού κόμβου.



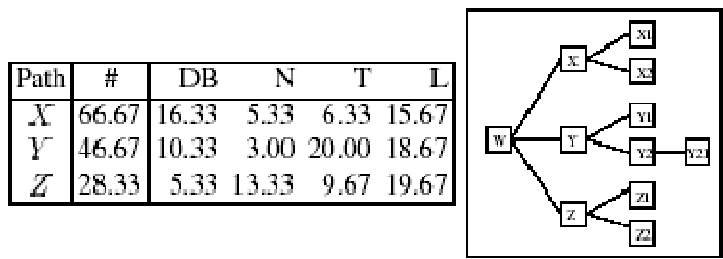
Στο παραπάνω παράδειγμα βλέπουμε το CRI ενός κόμβου A, ο οποίος μπορεί να βρει μέσω του γειτονικού του κόμβου B 100 δεδομένα, εκ των οποίων τα 20 αφορούν την κατηγορία databases, 0 την κατηγορία networks, 10 την κατηγορία theory και 30 την κατηγορία languages. Με την ίδια διαδικασία βλέπουμε ότι αναφέρεται και ο αριθμός των δεδομένων σε κάθε κατηγορία που μπορούν να βρεθούν μέσω των γειτόνων του A.

Σε ένα hop count RI σύστημα, σε αντίθεση με τα προηγούμενα, λαμβάνεται υπ'όψιν και ο αριθμός των hops, των βημάτων μετάβασης δηλαδή, από τον ένα κόμβο στον άλλο, που απαιτούνται για την εύρεση των δεδομένων. Αυτά αποθηκεύουν συγκεντρωτικά RIs για κάθε βήμα, σύμφωνα με έναν μέγιστο αριθμό που λέγεται horizon.

Σχετικά με τη δρομολόγηση, είναι αναγκαία η χρήση ενός εκτιμητή των στοιχείων του HRI, με σκοπό κάθε ερώτηση να προωθείται στον καλύτερο δυνατό γειτονικό κόμβο, ακολουθώντας την προηγούμενη διαδικασία. Στο παρακάτω σχήμα διακρίνεται το HRI του κόμβου w με 2 hops με τις πληροφορίες να αποθηκεύονται όπως και στο CRI.



Στο τελευταίο παράδειγμα, σε ένα ERI αποθηκεύονται τα αποτελέσματα ενός HRI εκτιμητή, με βάση τα οποία δρομολογούνται και οι ερωτήσεις σε έναν γειτονικό κόμβο. Στο παρακάτω σχήμα βλέπουμε το ERI του κόμβου w στο οποίο αποθηκεύονται τα αποτελέσματα που προκύπτουν από τον εκτιμητή του HRI του κόμβου στον οποίο αναφερόμαστε.



Distributed Resource Location Protocol – DRLP

Ο αλγόριθμος αυτός είναι πανομοιότυπος με τον APS, με τη διαφορά ότι η εκτίμηση της πιθανότητας δεν αφορά κάποιο συγκεκριμένο δεδομένο, παρά μόνο τον γειτονικό κόμβο. Η αναζήτηση θα προωθηθεί στον γειτονικό κόμβο ο οποίος έχει τη μεγαλύτερη πιθανότητα, ανεξάρτητα με το θέμα της αναζήτησης.

Τα αποτελέσματα επιστρέφουν στον αρχικό κόμβο ενημερώνοντας παράλληλα τους ενδιαμέσους κόμβους.

Ακολουθεί πίνακας με ορισμένα στοιχεία αναφορικά με τους αλγόριθμους αναζήτησης με χρήση ευρετηρίων.

Μέθοδος Τυφλής Αναζήτησης	Αξιολόγηση με βάση περιεχόμενο/ πλήθος αποτελεσμάτων	Ευρετήριο με	Ενημέρωση κόμβων ενδιάμεσης διαδρομής
<i>Intelligent BFS</i>	περιεχόμενο	δεδομένο ανά γείτονα	✓
<i>APS</i>	περιεχόμενο	δεδομένο ανά γείτονα	✓
<i>Routing Indexes</i>	περιεχόμενο	δεδομένο ανά γείτονα	
<i>Local Indices</i>	περιεχόμενο	δεδομένα	
<i>DRPL</i>	Πλήθος αποτελεσμάτων	απαντήσεις από γείτονες	

6.2.3 Μέθοδοι τοποθέτησης αντιγράφων (Replication)

Ένας τρόπος μείωσης του κόστους αναζήτησης σε ένα δίκτυο, είναι η δημιουργία αντιγράφων των δεδομένων ορισμένων χρηστών, σε διάφορα σημεία του δικτύου. Με τον τρόπο αυτό επιτυγχάνεται η αύξηση της διαθεσιμότητας των δεδομένων στο δίκτυο και συνεπώς τη μείωση του κόστους αναζήτησης.

Uniform replication: Η τεχνική αυτή δεν είναι πολύπλοκη και βασίζεται στη δημιουργία αντιγράφων όλων των δεδομένων, τα οποία δεν διακρίνονται σε δημοφιλή ή μη. Έτσι έχουμε μικρό κόστος αναζήτησης για όλα τα δεδομένα, τα οποία όπως είναι και λογικό, διαθέτουν τον ίδιο αριθμό αντιγράφων.

Proportional replication: Η μέθοδος αυτή εφαρμόζεται κυρίως στα πιο δημοφιλή δεδομένα. Ο αριθμός των αντιγράφων ενός δεδομένου, είναι ανάλογος με τον αριθμό των αναζητήσεων που έγιναν για το δεδομένο αυτό. Το γεγονός αυτό μειώνει το κόστος αναζήτησης για τα μη δημοφιλή δεδομένα, σαν φυσικό επακόλουθο και χρησιμοποιώντας τη θεωρία αυτή μπορούμε σε γενικές γραμμές να πούμε ότι το συνολικό κόστος είναι ίδιο με αυτό της uniform replication.

Square-root replication: Χρησιμοποιώντας αυτή τη μέθοδο, μπορούμε να βρούμε ένα κριτήριο για τη δημιουργία αντιγράφων δεδομένων το οποίο θα μειώνει το συνολικό κόστος των αναζητήσεων. Ο αριθμός των αντιγράφων ενός δεδομένου είναι ανάλογος με την τετραγωνική ρίζα του αριθμού των αναζητήσεων που έγιναν για το συγκεκριμένο δεδομένο.

Ακόμη ένα κριτήριο για τον αριθμό των αντιγράφων ενός δεδομένου είναι πόσες φορές χρησιμοποιήθηκε το αντίγραφο αυτό. Όπως είναι λογικό, τα αντίγραφα με τη μεγαλύτερη χρηστικότητα, είναι τα πιο δημοφιλή δεδομένα και άρα μπορούμε να δημιουργήσουμε περισσότερα αντίγραφα.

6.2.4 Μέθοδοι διατήρησης των αντιγράφων στο δίκτυο

Απο τη στιγμή που κάποιοι κόμβοι στο δίκτυο έχουν αποκτήσει αντίγραφα, θα πρέπει να υπάρχει και ένας τρόπος ενημέρωσης των αντιγράφων, σε περίπτωση επεξεργασίας του αρχικού δεδομένου.

Οι τρόποι διάδοσης ενός αρχικού δεδομένου στους κόμβους που διατηρούν αντίγραφο είναι οι εξής:

Μεθοδος Push

Ο κόμβος με το αρχικό δεδομένο στέλνει σε όσους κόμβους γνωρίζει, αντίγραφο του αναβαθμισμένου δεδομένου. Έτσι ο κόμβος-κάτοχος του αρχικού δεδομένου μπορεί να ενημερώνει τα αντίγραφα του δικτύου.

Μέθοδος Pull

Οι κόμβοι οι οποίοι διατηρούν ένα αντίγραφο του δεδομένου, ζητούν από τους κόμβους-ιδιοκτήτες να τους αποσταλεί, εφόσον υπάρχει, ένα πιο πρόσφατο αντίγραφο του δεδομένου. Η μέθοδος αυτή χρησιμοποιείται κυρίως όταν τα αντίγραφα διαθέτουν ημερομηνίας λήξης.

Παρακάτω θα δούμε έναν αλγόριθμο για τη διατήρηση ενημερωμένων εγγράφων, ο οποίος βασίζεται στις παραπάνω μεθόδους.

Ονομάζεται **Push&Pull** και αποτελείται από 2 φάσεις. Υποθετικά, οι κόμβοι που διατηρούν τα αντίγραφα είναι γνωστοί και εύκολο να εντοπιστούν.

Push φάση

Στην φάση αυτή, ο ιδιοκτήτης του δεδομένου, ενημερώνει ορισμένους γνωστούς όσον αφορά τα αρχεία, κόμβους και τους αποστέλει το νέο αντίγραφο. Οι κόμβοι που δέχονται το αντίγραφο, επαναλαμβάνουν τη διαδικασία για τους δικούς τους γνώριμους κόμβους, με έναν περιορισμό όμως για την επανάληψη της προώθησης του αντιγράφου, από τους κόμβους που το έλαβαν.

Ένας μετρητής αναλαμβάνει την καταμέτρηση των προωθήσεων και όταν φτάσει την εξ'αρχής ορισμένη μέγιστη τιμή, σταματάει η μετάδοση του αντιγράφου. Σε κάποιες περιπτώσεις, ορισμένοι κόμβοι λαμβάνουν πολλές φορές το ίδιο αίτημα.

Η περιττή διάδοση πληροφορίας μειώνεται, λόγω του ότι οι κόμβοι που προωθούν το νέο αντίγραφο δεν το στέλνουν σε όλους τους γνώριμους κόμβους αλλά σε μερικούς από αυτούς. Εάν είναι απαραίτητη η μετάδοση ενός αντιγράφου, τότε γίνεται push άμεσα, ώστε όλοι να αποκτήσουν την τελευταία έκδοση του αντιγράφου.

Pull φάση

Σε περίπτωση που ένας κόμβος του δικτύου είναι αποσυνδεδεμένος, είναι πολύ πιθανόν να μη διαθέτει την τελευταία έκδοση του αντιγράφου. Έτσι, θα ζητήσει από κάποιον κόμβο ο οποίος διαθέτει αντίγραφο, να του αποστείλει την πιο πρόσφατη έκδοση.

Εάν ο κόμβος αυτός δεν είναι σίγουρος για την έκδοση του αντιγράφου, τότε κάνει και αυτός αντίστοιχα pull σε κάποιον άλλο, με τη διαδικασία να επαναλαμβάνεται, μέχρι να βρεθεί το πιο πρόσφατο αντίγραφο.

Συμπέρασμα

Σύμφωνα με πειραματικές μετρήσεις, έχει αποδειχτεί ότι εάν ο αριθμός των κόμβων που βρίσκονται σε σύνδεση σε σχέση με το συνολικό αριθμό των κόμβων του δικτύου, είναι μικρός όταν ξεκινάει μία ενημέρωση, τότε η πιθανότητα ο κόμβος στον οποίο θα σταλεί μήνυμα να είναι διαθέσιμος, είναι αρκετά μικρή. Συνεπώς, η ενημέρωση δεν θα εξαπλωθεί επαρκώς στο δίκτυο. Ακόμη ένα συμπέρασμα έπειτα από σχετικά πειράματα, είναι το γεγονός ότι εάν κάθε κόμβος προωθεί την ενημέρωση σε έναν περιορισμένο αριθμό κατάλληλων κόμβων, αυτό είναι επαρκές ώστε η ενημέρωση να εξαπλωθεί σε όλα τα αντίγραφα.

Αντιθέτως, εάν ο αριθμός των κόμβων είναι μεγάλος, τότε προκαλείται συμφόρηση στο δίκτυο, λόγω των περιπτώσεων και διπλών μηνυμάτων. Ένας ακόμη τρόπος για τη μείωση της συμφόρησης του δικτύου είναι η μείωση της πιθανότητας προώθησης της ενημέρωσης από έναν κόμβο, στην περίπτωση που το μήνυμα είναι αρκετά μακριά από τον αρχικό κόμβο.

Αυτό οφείλεται στο γεγονός ότι όταν η ενημέρωση έχει εκτελέσει έναν μεγάλο αριθμό βημάτων, τότε απομένουν ελάχιστοι κόμβοι να ενημερωθούν και σε περίπτωση που οι κόμβοι συνεχίσουν την αποστολή του ίδιου αριθμού μηνυμάτων, τότε κάποια από αυτά θα επαναλαμβάνονται χωρίς λόγο.

Chord

Το πρωτόκολλο αυτό ανήκει στην κατηγορία των κατανεμημένων και λύνει το πρόβλημα της αποδοτικής τοποθέτησης των κόμβων που κατέχουν ένα συγκεκριμένο δεδομένο. Σύμφωνα με ένα κλειδί, το CHORD το αντιστοιχίζει σε έναν κόμβο.

Το πρωτόκολλο αυτό εφαρμόζει μία εξισορρόπηση στο φορτίο, αφού κάθε κόμβος δέχεται σχεδόν τον ίδιο αριθμό κλειδιών. Σε ένα σύστημα N κόμβων, κάθε κόμβος διατηρεί $O(\log N)$ περίπου πληροφορίες για τους υπόλοιπους κόμβους, τις οποίες και χρησιμοποιεί για τη δρομολόγηση.

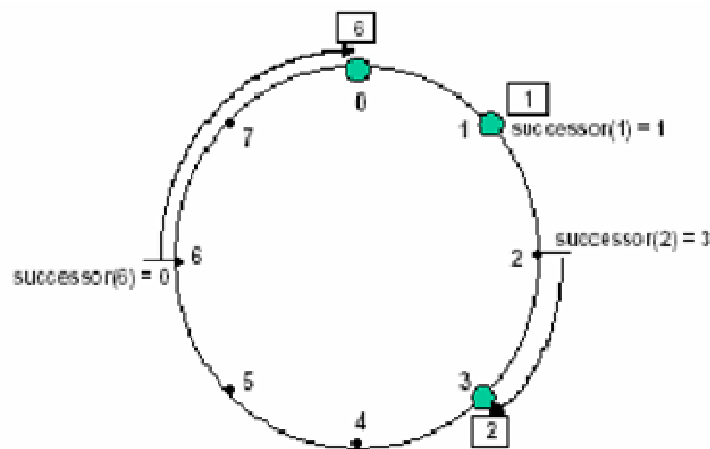
Η συνάρτηση που χρησιμοποιείται αντιστοιχίζει σε κάθε ζευγάρι κόμβου-κλειδί, ένα αναγνωριστικό μεγέθους m bits. Το αναγνωριστικό αυτό προκύπτει με βάση τη συνάρτηση που εφαρμόζεται πάνω στην IP του κόμβου και του κλειδιού αντίστοιχα. Τα στοιχεία ταξινομούνται σε έναν κύκλο με 2^m στοιχεία, όπως φαίνεται και στο παρακάτω σχήμα που εμφανίζεται ένα CHORD δίκτυο με 3 κόμβους, 0, 1 και 3.

Το αναγνωριστικό που αντιστοιχεί στο κλειδί που αποθηκεύεται στον πρώτο κόμβο, είναι ίσο με αυτό του κλειδιού ή στον επόμενο κόμβο πάνω στον κύκλο. Ο κόμβος αυτός ονομάζεται successor του κλειδιού. Στο σχήμα μας, το successor του κλειδιού 1 είναι ο κόμβος 1, του κλειδιού 2 ο κόμβος 3 και του κλειδιού 6 ο κόμβος 0.

Σε περίπτωση που ένας κόμβος εισαχθεί στο δίκτυο, μοιράζεται κάποια από τα κλειδιά που είχε ο successor του, ενώ όταν φεύγει από αυτό, δίνει όλα τα κλειδιά στο successor του. Πέρα από τη γνώση που έχει ένας κόμβος για τον successor του σύμφωνα με τον κύκλο, διαθέτει και έναν πίνακα δρομολόγησης με m εγγραφές, όπου σε κάθε μία περιλαμβάνεται το αναγνωριστικό και η IP διεύθυνση ενός κόμβου s .

Ο κόμβος της εγγραφής i , θα είναι και ο πρώτος για τον οποίο θα ισχύσει η σχέση:
 $s = \text{successor}(n + 2^{i-1})$

όπου $1 \leq i \leq m$ και n ο αριθμός κόμβων



Η δρομολόγηση σε ένα δίκτυο τέτοιου τύπου, γίνεται σύμφωνα με τον πίνακα δρομολόγησης. Ο κόμβος ο οποίος θέλει να ανακτήσει ένα δεδομένο, πρέπει να βρεί τον κόμβο που το κατέχει. Η φιλοσοφία είναι να βρεθεί το id του προηγούμενου κόμβου από αυτόν ο οποίος το κατέχει τώρα. Σε περίπτωση που ο n είναι ο προκάτοχος του ζητούμενου κόμβου, η διαδικασία τελειώνει.

Σε αντίθετη περίπτωση, ο κόμβος n σύμφωνα με τον πίνακά του, βρίσκει τον κοντινότερο κόμβο στον προκάτοχο. Η διαδικασία θα λάβει τέλος, όταν βρεθεί ο ζητούμενος κόμβος.

Pastry

Και αυτό το σύστημα έχει κυκλική δομή. Με κάθε μήνυμα να έχει ένα κλειδί, το οποίο στέλνεται στον κόμβο με id αριθμητικά πιο κοντά στο κλειδί του μηνύματος.

Σε κάθε κόμβο υπάρχει ένας πίνακας με L καταχωρήσεις, από τις οποίες οι μισές έχουν στοιχεία για τους $|L|/2$ κοντινότερους κόμβους με μεγαλύτερο id, ενώ οι άλλες μισές έχουν τους $|L|/2$ κόμβους με μικρότερο id. Αυτή είναι και η μόνη πληροφορία που χρειάζεται να δεχτεί σωστά το Pastry για να λειτουργήσει.

Για λόγους απόδοσης του συστήματος, σε κάθε κόμβο διατηρείται και ένας δεύτερος πίνακας με δείκτες σε διάφορους κόμβους του δικτύου.

Ένας αριθμός y bits αναπαριστά το id του κάθε κόμβου. Η εγγραφή no. i του πίνακα περιέχει τον κόμβο που έχει τα i πρώτα στοιχεία ίδια με τον κόμβο που διατηρεί τον πίνακα και διαφέρουν στο $i+1$ bit.

Σε περίπτωση μιας αναζήτησης, ένας κόμβος ψάχνει στον πρώτο πίνακα εάν είναι καταχωρημένος ο κόμβος αυτός. Στις περισσότερες περιπτώσεις αυτό δεν συμβαίνει, οπότε και αντλεί πληροφορίες από τον δεύτερο πίνακα, προωθώντας την αναζήτηση σε έναν κόμβο ο οποίος ανήκει στον δεύτερο πίνακα και έχει μεγαλύτερο id πρόθεμα από αυτό του προς αναζήτηση κόμβου.

Εάν δεν υπάρχει κόμβος με το μεγαλύτερο πρόθεμα, είτε επειδή δεν ανήκει στον ίδιο πίνακα, είτε επειδή δεν υπάρχει, τότε πρέπει να προωθηθεί σε έναν κόμβο ο οποίος διαθέτει ίδιο μήκος κοινού προθέματος, με id μικρότερο από το κλειδί αναζήτησης.

Η αναζήτηση θα προχωρήσει στον κόμβο που είναι πλησιέστερος στο κλειδί αλλά και πριν αυτό. Σε περίπτωση που η πληροφορία δεν υπάρχει στον πρώτο πίνακα, η αναζήτηση έχει φτάσει στον πλησιέστερό ή τον αμέσως προηγούμενο κόμβο.

ΚΕΦΑΛΑΙΟ 7

7. Τεχνικές Διαχείρισης Αξιοπιστίας σε ένα Peer-to-Peer Σύστημα – Trust Management

Οι τεχνικές διαχείρισης αξιοπιστίας σε ένα P2P σύστημα είναι αρκετές και στηρίζονται στον υπολογισμό της φήμης (reputation) των κόμβων (peers) που το αποτελούν. Χωρίζονται σε 2 βασικές κατηγορίες, σύμφωνα με το είδος της πληροφορίας που χρησιμοποιείται για τον υπολογισμό της φήμης.

Στην πρώτη κατηγορία υπολογίζεται η υποκειμενική άποψη, δηλαδή η τοπική πληροφορία, με βάση την οποία ο συγκεκριμένος κόμβος κάνει την αξιολόγηση, αλλά και επίσης η συνολική άποψη όλων των κόμβων του δικτύου για την φήμη του υπό αξιολόγηση κόμβου.

Οι τεχνικές οι οποίες ανήκουν σε αυτή την κατηγορία και παρουσιάζονται παρακάτω είναι οι εξής: XREP, P-Grid, NICE και Framework using Currency.

Στην δεύτερη κατηγορία υπολογίζονται οι σχέσεις εμπιστοσύνης που έχουν αναπτυχθεί μεταξύ των μελών του δικτύου. Οι σχέσεις αυτές μπορεί να είναι σχέσεις φιλίας, συνεργασίας, αμοιβαίας υποστήριξης κ.α., οι οποίες στην ουσία αντιπροσωπεύουν σχέσεις εμπιστοσύνης μεταξύ των κόμβων και προβάλλονται μέσω των κοινωνικών δικτύων. Με βάση τις πληροφορίες που δημιουργούνται σε ένα τέτοιο κοινωνικό δίκτυο, υπολογίζεται και η αξιοπιστία του κόμβου.

Σε αυτή την κατηγορία παρουσιάζονται οι εξής τεχνικές:

REGRET και NodeRanking.

7.1 XREP

Σύμφωνα με το μοντέλο XREP, για τον υπολογισμό της αξιοπιστίας ενός εγγράφου, το οποίο πρόκειται να κατεβάσει ο χρήστης του P2P συστήματος, λαμβάνεται υπ' όψη ο συνδυασμός της φήμης (reputation) του servent (ο πάροχος του resource) και του resource (εγγράφου) που πρόκειται να ληφθεί. Servent ονομάζονται οι κόμβοι κάθε συστήματος, καθώς σε ένα P2P δίκτυο κάθε κόμβος λειτουργεί τόσο σαν server, όσο και σαν client.

Σε κάθε servent αντιστοιχεί ένα αναγνωριστικό (servent_id) το οποίο είναι μοναδικό για κάθε servent. Αντίστοιχα, κάθε έγγραφο έχει το δικό του αναγνωριστικό (resource_id) το οποίο προκύπτει μέσω hash function στο περιεχόμενο του εγγράφου.

Για να αντιμετωπιστεί το θέμα της εμπιστοσύνης από το XREP, διεξάγεται ένα είδος ψηφοφορίας μεταξύ των κόμβων του δικτύου. Σε αυτή την ψηφοφορία, κάθε κόμβος του δικτύου εκφράζει την άποψή του όχι μόνο για το συγκεκριμένο έγγραφο, αλλά και για τον κάτοχό του, δηλαδή τον servant.

Η άποψη του κάθε κόμβου είναι δυνατόν να διαμορφωθεί μέσω των συναλλαγών που είχε ο κόμβος με τον servant αλλά και με το έγγραφο. Έτσι κάθε κόμβος μπορεί να μεταφέρει την εμπειρία του από οποιαδήποτε συναλλαγή είχε, στους υπολοίπους κόμβους.

Για να γίνει εφικτή η παραπάνω διαδικασία, κάθε peer διαθέτει 2 repositories: το resource repository και το servant repository.

Για το resource repository χρησιμοποιείται ένας πίνακας ο οποίος περιέχει όλα τα resource_id των εγγράφων με τα οποία ο peer έχει προηγούμενη εμπειρία. Σε κάθε resource_id αντιστοιχεί ένας δυαδικός αριθμός, θετικός ή αρνητικός. Η τιμή αυτή συμβολίζει την άποψη του κόμβου για το συγκεκριμένο έγγραφο.

Αντίστοιχα, για το servant repository χρησιμοποιείται ένας πίνακας με τα servant_id των κόμβων που γνωρίζει ο peer και σε κάθε id αντιστοιχούν 2 αριθμοί: num_plus και num_minus, οι οποίοι δηλώνουν αντίστοιχα τα επιτυχή και ανεπιτυχή download του peer από τον συγκεκριμένο servant. Με βάση τις τιμές αυτές αποδίδονται οι ψήφοι από τον peer όποτε του ζητηθεί η γνώμη για ένα έγγραφο ή για έναν servant.

Το πρωτόκολλο XREP για των σωστό υπολογισμό της εμπιστοσύνης ενός κόμβου, σύμφωνα πάντα με τα προηγούμενα στοιχεία που κρατά κάθε κόμβος χωρίζεται σε 5 φάσεις: Resource Searching, Resource Selection & Vote Polling, Vote Evaluation, Best Servant Check και Resource Downloading.

Σε 1^η φάση, κάποιος κόμβος στέλνει μία ερώτηση μέσω ενός Query message στους κόμβους του δικτύου και αυτό με τη μέθοδο flooding προωθείται. Κάθε peer που διατηρεί έγγραφα με keyword ίδιο με αυτό της ερώτησης απαντά με ένα Query Hit Message,

επιστρέφοντας το `servant_id` του, τα `resource_id` των εγγράφων που ικανοποιούν την ερώτηση αλλά και κάποια ακόμη στοιχεία απαραίτητα για το download του εγγράφου (IP Address, Port Number).

Σε 2^η φάση, ο κόμβος ο οποίος έκανε την ερώτηση, ζητά από τους υπόλοιπους `peers` του συστήματος να δώσουν την ψήφο τους για το `resource` που θέλει να κάνει download αλλά και για τους κόμβους οι οποίοι το κατέχουν. Αυτό γίνεται στέλνοντας `Poll Messages` στο δίκτυο ζητώντας ψήφους για ένα `resource` αλλά και για τους `servant` που το προσφέρουν.

Σε κάθε `Poll message` υπάρχει ένα `rkroll` κλειδί, το οποίο θα χρησιμοποιηθεί για την κρυπτογράφηση των ψήφων. Κάθε κόμβος που λαμβάνει μια `Poll` αίτηση, ελέγχει τα `repositories` του και απαντά στον `peer` με ένα `PollReply message`, κρυπτογραφημένο με το `rkroll` κλειδί, στο οποίο περιέχονται οι ψήφοι, καθώς και τα IP Address & Port Number του κόμβου.

Σε 3^η φάση ο κόμβος αφού έχει λάβει όλες τις ψήφους πρέπει να αποφασίσει ποιο `resource` θα κάνει download. Για να γίνει αυτό, πρέπει πρώτα να γίνει η επιβεβαίωση της αξιοπιστίας των ψήφων που έστειλαν οι διάφοροι `peers`. Πρώτα αποκρυπτογραφεί τις ψήφους και εντοπίζει ποιοι κόμβοι έχουν κάνει πλαστογράφηση και τους απορρίπτει.

Έπειτα για να απομονώσει ομάδες κακών κόμβων, ομαδοποιεί τις ψήφους των κόμβων που έχουν το ίδιο `net_id` και υπολογίζει τον μέσο όρο των ψήφων τους, λαμβάνοντάς τον ως την τελική ψήφο του `cluster`. Στη συνέχεια διαλέγει ένα σύνολο κόμβων, ζητώντας τους να επιβεβαιώσουν την ψήφο τους. Στέλνει ένα `True Vote Message` και κάθε κόμβος απαντά με ένα `True Vote Reply Message`.

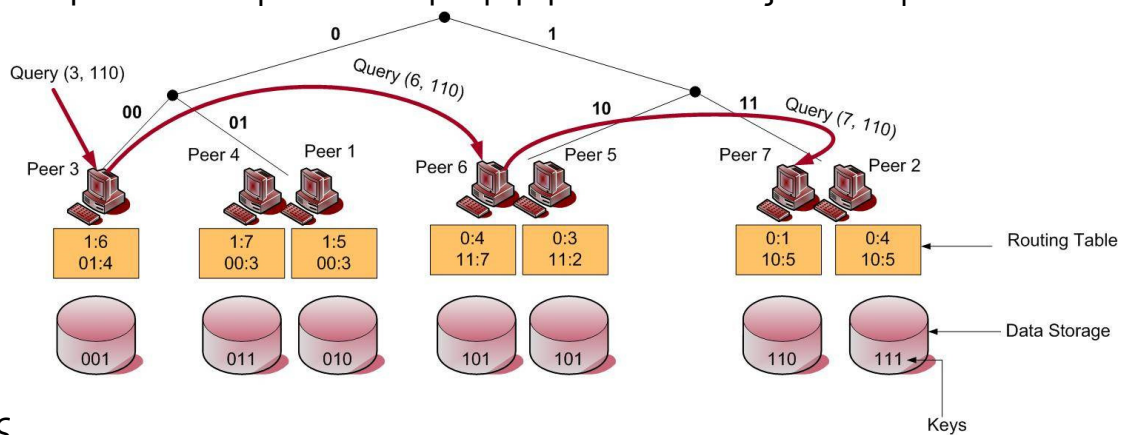
Σύμφωνα με τις αξιόπιστες ψήφους στις οποίες έχει καταλήξει, αποφασίζει και ποιο `resource` θα κάνει download καθώς και από ποιόν `servant` θα γίνει το download. Σαν τελευταίο βήμα, γίνεται ερώτηση στον κόμβο με την μεγαλύτερη φήμη εάν κατέχει το συγκεκριμένο `resource` και αν η απάντηση είναι θετική πραγματοποιείται η λήψη του, παράλληλα με την ενημέρωση των `repositories`.

7.2 P-GRID

Το συγκεκριμένο μοντέλο είναι ένα ακόμη `reputation-based` πρωτόκολλο για τον υπολογισμό της αξιοπιστίας ενός κόμβου σε ένα P2P δίκτυο. Η φήμη κάθε κόμβου διαμορφώνεται σύμφωνα με την ανάλυση των προηγούμενων συναλλαγών του με άλλους κόμβους του συστήματος. Έτσι κάθε `peer` χρησιμοποιώντας μία μέθοδο εξόρυξης δεδομένων (`data mining`) θα ήταν εφικτό να μπορεί να έχει μία εκτίμηση για την φήμη ενός άλλου κόμβου.

Αυτό γίνεται με τη χρήση της τεχνικής που προτείνεται χρησιμοποιώντας τη δομή του P-Grid, για την αποθήκευση των στατιστικών στοιχείων, τα οποία στην περίπτωση μας είναι παράπονα που εκφράζουν κάποιοι κόμβοι για τη συμπεριφορά κάποιου άλλου.

Λέγοντας παράπονα εννοούμε μία εγγραφή δεδομένων που δηλώνει την αρνητική άποψη για τη συμπεριφορά ενός peer του



συστήματος

Η παραπάνω προσέγγιση προτείνει ως τρόπο υπολογισμού της φήμης ενός peer (p) το γινόμενο $T(p)$ των παραπόνων που έχουν εκφράσει οι κόμβοι του δικτύου για τη συμπεριφορά του συγκεκριμένου peer επί των παραπόνων που έχει αναφέρει ο συγκεκριμένος peer για όλους τους υπόλοιπους κόμβους του δικτύου. Όσο μεγαλύτερο είναι το $T(p)$, τόσο μικρότερη είναι και η αξιοπιστία του peer.

Όμως η ανάκτηση όλων των παραπόνων που έχουν καταθέσει όλοι οι κόμβοι του συστήματος για έναν peer δεν είναι εύκολα εφικτή και περιορίζει πάρα πολύ τη κλίμακα της μεθόδου, έγινε αναγκαία η αποθήκευση των παραπόνων στη δομή του P-Grid.

Το P-Grid είναι ένα δυαδικό δέντρο, τα φύλλα του οποίου είναι οι peers του κάθε δικτύου. Επομένως, κάθε peer σχετίζεται με ένα συγκεκριμένο μονοπάτι του δέντρου. Σε κάθε κόμβο του δικτύου αποθηκεύεται ένα μέρος από το σύνολο των παραπόνων που έχουν εκφράσει όλοι οι κόμβοι του P2P δικτύου.

Η ανάθεση των παραπόνων δεν είναι τυχαία, αλλά αποθηκεύονται στους κόμβους εκείνους των οποίων το path που τους αντιστοιχεί είναι prefix part του κλειδιού του κάθε παραπόνου. Το κλειδί του παραπόνου σχετίζεται με τον αριθμό του peer για τον οποίο εκδόθηκε. Ακόμη κάθε κόμβος διαθέτει ένα routing table για να προωθεί τις ερωτήσεις που δέχεται σε άλλους κόμβους οι οποίοι είναι υπεύθυνοι να απαντήσουν, καθώς διατηρούν τα κατάλληλα στοιχεία, δηλ. παράπονα. Με τη βοήθεια των συναρτήσεων $intser(a,k,u)$ & $query(a,k)$ μπορεί εύκολα ο κάθε κόμβος του συστήματος να εισάγει και να ανακτήσει ένα παράπονο για κάποιο peer του δικτύου.

Έτσι, όταν κάποιος θέλει να υπολογίσει την αξιοπιστία ενός peer του δικτύου, αυτό που πρέπει να κάνει για αρχή είναι να βρει όλα τα παράπονα που είναι αποθηκευμένα στο P-Grid.

Αφού το κάνει αυτό, είναι σε θέση να υπολογίσει την αξιοπιστία του peer χρησιμοποιώντας τη συνάρτηση $T(p)$. Για να έχει σωστό αποτέλεσμα όμως η διαδικασία αυτή, είναι απαραίτητη η φερεγγυότητα των κόμβων που έχουν αποθηκευμένα τοπικά τα συγκεκριμένα παράπονα. Άρα είναι απαραίτητο να γίνει ερώτηση για τους συγκεκριμένους κόμβους, καθώς επίσης και για τα παράπονα τα οποία τους αφορούν.

Επειδή όμως αυτό μπορεί να οδηγήσει σε μία αναζήτηση εντός ολόκληρου του δικτύου, η αναζήτηση αυτή μπορεί να περιοριστεί λαμβάνοντας παράπονα από ένα μικρό σύνολο κόμβων r και ελέγχοντας αν διαθέτουν τα ίδια δεδομένα για τον συγκεκριμένο κόμβο που γίνεται ο έλεγχος.

Αν κάτι τέτοιο ισχύει, τότε δεν χρειάζεται η περαιτέρω διερεύνηση της φερεγγυότητας του peer. Τελευταίο βήμα είναι η χρήση της $T(p)$ με τα complaints που ανακτήθηκαν από peers για τους οποίους η αξιοπιστία είναι πλέον δεδομένη.

7.3 NICE

Σε αυτό το μοντέλο θα παρουσιάσουμε την τεχνική που χρησιμοποιείται για την αποθήκευση της πληροφορίας στους κόμβους, η οποία βοηθά στον υπολογισμό της φήμης κάθε peer του δικτύου, καθώς επίσης και τον τρόπο με τον οποίο ο κάθε κόμβος αξιοποιεί την πληροφορία αυτή, έτσι ώστε να λάβει την απόφαση για την αξιοπιστία του peer με τον οποίο θα γίνει η αλληλεπίδραση.

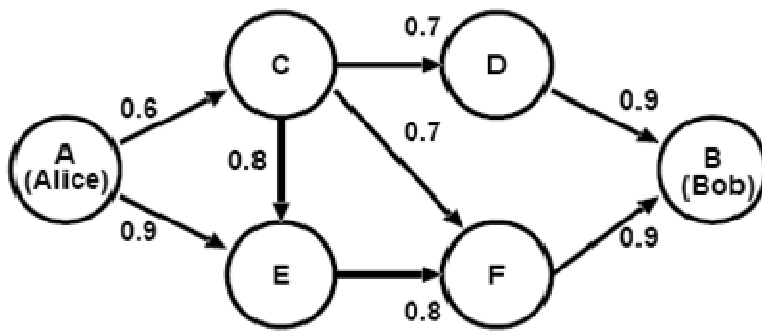
Για να γίνει αυτό, γίνεται χρήση του NICE, μια πλατφόρμα η οποία υλοποιεί cooperative εφαρμογές, δηλαδή εφαρμογές οι οποίες διαθέτουν ένα κομμάτι από τους πόρους τους για χρήση από τους υπόλοιπους peers της εφαρμογής.

Το ιδιαίτερο χαρακτηριστικό αυτής της τεχνικής είναι ότι κάθε κόμβος χρειάζεται λίγες πληροφορίες, αυτές που έχουν σχέση με το κατά πόσο οι υπόλοιποι κόμβοι του δικτύου τον εμπιστεύονται. Η πληροφορία αυτή αποτελείται από ένα σύνολο cookies, δηλαδή μια δήλωση υπογεγραμμένη από έναν κόμβο, στην οποία φαίνεται η ποιότητα εξυπηρέτησης που έλαβε ο συγκεκριμένος κόμβος, από κάποιον άλλο στο δίκτυο.

Τα cookies τα εκδίδει εκείνος ο οποίος εξυπηρετείται, ενώ αποθηκεύονται σε αυτούς που παρείχαν την εξυπηρέτηση. Έτσι σε κάθε κόμβο αποθηκεύεται η πληροφορία που είναι μόνο προς δικό του όφελος και δηλώνει την αξιοπιστία των υπηρεσιών που προσφέρει. Τα negative cookies διατηρούνται στον κόμβο τον οποίο εκδόθηκαν, καθώς δεν τους οφείλει να διατηρούν τοπικά αυτή την πληροφορία.

Για να υπολογιστεί η αξιοπιστία των κόμβων, γίνεται χρήση ενός κατευθυνόμενου γράφου T που ονομάζεται trust γράφος. Στον γράφο αυτό, κάθε κορυφή είναι ένας κόμβος του δικτύου και κάθε κατευθυνόμενη ακμή από ένα κόμβο A σε ένα κόμβο B , τη τιμή της οποίας δηλώνει κατά πόσο ο A εμπιστεύεται τον B , το οποίο εξαρτάται άμεσα από τα cookies του A που διατηρεί ο B . Το μέτρο αυτής της εμπιστοσύνης προσδιορίζεται από την τιμή η οποία υπάρχει στην κατευθυνόμενη αυτή ακμή.

Σε περίπτωση που δεν υπάρχει άμεση σύνδεση μεταξύ δύο κόμβων, τότε μπορεί να χρησιμοποιηθεί ένα μονοπάτι του γράφου για τον υπολογισμό της τιμής εμπιστοσύνης. Για την εύρεση του ισχυρότερου (strongest path) απ' όλα τα μονοπάτια μεταξύ δύο κόμβων, επιλέγεται κάθε φορά η μετάβαση στον κόμβο με την μεγαλύτερη τιμή στην ακμή. Στο μονοπάτι αυτό αντιστοιχίζεται η μικρότερη τιμή ακμής που βρίσκεται στο μονοπάτι.



Στο παραπάνω σχήμα βλέπουμε ένα παράδειγμα της τεχνικής του strongest path, καθώς ο κόμβος A θα επιλέξει το μονοπάτι AEFB με τιμή 0.8 για να αξιολογήσει την αξιοπιστία του κόμβου B.

Αν κάποια στιγμή ο A θελήσει να χρησιμοποιήσει ένα τμήμα από τους πόρους του B, θα του στείλει όλα τα cookies που έχει στην κατοχή του και έχουν την υπογραφή του B. Σε περίπτωση που ο A δεν έχει cookies από τον B, δηλ. δεν υπάρχει ακμή BA στον γράφο T, τότε ο A ψάχνει για άλλους κόμβους τους οποίους εμπιστεύεται και οι οποίοι κατέχουν cookies του B.

Στο τέλος της αναζήτησης, ο A θα παρουσιάσει στον B ένα σύνολο από κατευθυνόμενα μονοπάτια τα οποία ξεκινούν από τον B και καταλήγουν στον A.

Στη συνέχεια ο B μπορεί να χρησιμοποιήσει την τεχνική που αναφέρθηκε και να υπολογίσει την αξιοπιστία του A. Τέλος, μία βελτίωση του συγκεκριμένου μοντέλου είναι η χρήση των preference lists, τις οποίες διατηρεί ο κάθε κόμβος. Οι κόμβοι που διατηρεί κάθε peer σε μία preference list είναι ένα σύνολο από κόμβους με τους οποίους δεν είχε καμία συναλλαγή αλλά τους θεωρεί αρκετά αξιόπιστους.

Στην λίστα ενός peer A αποθηκεύονται οι κόμβοι ενός μονοπατιού από τον A στον B οι οποίες έχουν αρκετά μεγάλο δείκτη εμπιστοσύνης (trust values).

7.4 Reputation Framework using Currency

Σε αυτό το μοντέλο διαχείρισης φήμης των κόμβων ενός P2P, όλοι οι κόμβοι συμπεριφέρονται εγωιστικά, προσπαθώντας ο καθένας να αυξήσει την δική του φήμη. Για να εμπιστευτεί ο ένας κόμβος τον άλλον, δημιουργούνται κοινότητες εμπιστοσύνης (trusted communities) οι οποίες ομαδικά συνεργάζονται για να αντιμετωπίσουν τη συμπεριφορά των άλλων κόμβων, καθώς επίσης και την άσχημη συμπεριφορά των κόμβων που θέλουν να αποσταθεροποιήσουν το δίκτυο και την αξιοπιστία του.

Για την παροχή υπηρεσιών και τις συναλλαγές μεταξύ των κόμβων γίνεται χρήση μιας εικονικής χρηματικής ανταμοιβής.

Έτσι είναι αναγκαίο να δημιουργηθεί ένα εικονικό νόμισμα (virtual currency) σύμφωνα με το οποίο θα υπολογίζεται η φήμη του κάθε κόμβου, αναλογικά με τον πλούτο του. Σε μία ανταλλαγή υπηρεσιών μεταξύ δύο κόμβων, εκείνος που παρέχει την υπηρεσία ανταμείβεται με ένα εικονικό ποσό, δηλαδή αυξάνεται η φήμη του, ενώ αυτός ο οποίος εξυπηρετείται επιβαρύνεται το αντίστοιχο ποσό, δηλαδή μειώνεται η φήμη του, καθώς έκανε χρήση των πόρων.

Δημιουργούνται κοινότητες εμπιστοσύνης, ανάλογα με την υπηρεσία που παρέχει ο κάθε κόμβος, έτσι ώστε να μην υπάρχουν στην ίδια κοινότητα κόμβοι οι οποίοι παρέχουν τις ίδιες υπηρεσίες. Κάθε κόμβος ανήκει σε μία το πολύ κοινότητα έτσι ώστε να αναπτυχθεί η εμπιστοσύνη μεταξύ των μελών της και να αποφευχθεί η εγωιστική συμπεριφορά μέσα στις κοινότητες. Η φήμη της κοινότητας ως προς τους κόμβους που δεν ανήκουν σε αυτή, υπολογίζεται από τον μέσο όρο των τιμών της φήμης όλων των μελών της.

Η τιμή αυτή μπορεί να κυμαίνεται από -1 έως 1 και αυξάνεται ή μειώνεται, έπειτα από μια θετική ή αρνητική συναλλαγή αντίστοιχα, αναλογικά με την φήμη του κόμβου που εξυπηρετείται. Σε περίπτωση που δεν παρέχονται υπηρεσίες από τους κόμβους, η τιμή αυτή μειώνεται με το χρόνο. Άρα κάθε κόμβος επιθυμεί να εξυπηρετεί κόμβους με μεγάλη φήμη.

Σε κάθε κόμβο διατηρούνται τρεις κατηγορίες στις οποίες αποθηκεύονται οι κόμβοι. Στην κατηγορία Good ανήκουν οι κόμβοι για τους οποίους υπάρχει καλή άποψη, Bad για αυτούς με κακή άποψη και φ οι κόμβοι για τους οποίους δεν υπάρχουν στοιχεία.

Ακόμη, αποθηκεύεται τοπικά η πληροφορία σχετικά με τη φήμη των κοινοτήτων οι οποίες είναι γνωστές αλλά και για τα υπόλοιπα μέλη της ίδιας κοινότητας. Η φήμη είναι αποτέλεσμα της υποκειμενικής άποψης του κόμβου για τις υπόλοιπες κοινότητες αλλά και για τους συνεργάτες του.

Υπάρχουν δύο προϋποθέσεις για να ενταχθεί ένας κόμβος σε μία κοινότητα.

1^η Δεν πρέπει κανένα μέλος της κοινότητας να έχει αποθηκευμένο τον συγκεκριμένο κόμβο στην κατηγορία Bad

2^η Ο κόμβος θα πρέπει να έχει μεγαλύτερη φήμη από κάθε άλλο κόμβο της κοινότητας στην οποία επιθυμεί να ενταχθεί.

Ένας κόμβος μπορεί να διαγραφεί από μία κοινότητα όταν η φήμη του μειωθεί σε τέτοιο βαθμό που την επηρεάζει αρνητικά. Η φήμη ενός κόμβου μπορεί να μειωθεί σε περίπτωση που υπάρχει κακή εξυπηρέτηση από αυτόν ή όταν ο κόμβος δεν εξυπηρετεί για μεγάλο χρονικό διάστημα. Η ενημέρωση των κόμβων για μία συναλλαγή αν ήταν επιτυχής ή όχι, εξαρτάται άμεσα από την εγωιστική συμπεριφορά των κόμβων.

Η πληροφορία έπειτα από μία επιτυχημένη ή όχι συναλλαγή, προωθείται από τον κόμβο ο οποίος έκανε την εξυπηρέτηση, σε ένα μικρότερο σύνολο κόμβων οι οποίοι αποτελούνται από μέλη της ομάδας του αλλά και γνωστούς ή ακόμα και φίλους του. Ένας κόμβος θα αυξήσει τη φήμη του για τον εξυπηρετητή ανάλογα με την τιμή της φήμης του εξυπηρετούμενου, όταν λάβει μήνυμα επιτυχούς εξυπηρέτησης.

Σε περίπτωση που τα μηνύματα αυτά φτάσουν σε κόμβους οι οποίοι δεν ανήκουν στην κοινότητα του εξυπηρετητή, εκείνοι αυξάνουν τη φήμη της κοινότητας του εξυπηρετητή αλλά λόγω της εγωιστικής τους συμπεριφοράς, το μήνυμα δεν προωθείται περαιτέρω.

Αντίθετα, σε μία αποτυχημένη εξυπηρέτηση, η πληροφορία μεταδίδεται από τον εξυπηρετούμενο με τη μέθοδο flooding (πλημμύρα), σε όσο το δυνατόν περισσότερους κόμβους. Εάν ένας κόμβος λάβει κάποιο complaint (παράπονο), θα μειώσει τις τιμές της φήμης και των δύο κόμβων που περιλαμβάνονται στο μήνυμα, δηλαδή και του εξυπηρετητή αλλά και του εξυπηρετούμενου.

Αυτό γίνεται για να αποφευχθεί η σκόπιμη συκοφαντία ενός κόμβου από έναν άλλο «κακό» κόμβο. Τα μέλη μέσα σε μία ομάδα υποστηρίζουν το ένα το άλλο και σε περίπτωση που κάποιος λάβει ένα παράπονο για κάποιο μέλος της κοινότητάς του που εμπλέκεται στο παράπονο, λειτουργεί υπέρ του μέλους της ομάδας του, μειώνοντας ελάχιστα την φήμη του, σε αντίθεση με τη φήμη της κοινότητας στην οποία ανήκει το άλλο μέλος, την οποία μειώνει αρκετά.

Σε περίπτωση που ο παραλήπτης ανήκει στην ίδια κοινότητα και με τους δύο κόμβους, τότε η φήμη τους μειώνεται ελάχιστα, θεωρώντας το σαν ατυχές γεγονός.

Εάν το παράπονο περιέχει πληροφορίες για δύο διαφορετικές κοινότητες σε καμία από τις οποίες δεν ανήκει ο παραλήπτης, τότε μειώνεται η φήμη και των δύο. Τέλος, εάν το παράπονο περιέχει μέλη της ίδιας κοινότητας στην οποία δεν ανήκει ο παραλήπτης δεν γίνεται καμία ενημέρωση.

7.7 Αξιολόγηση και Σύγκριση

Ακολουθεί μία προσπάθεια αξιολόγησης όλων των συστημάτων που παρουσιάστηκαν στην παραπάνω ενότητα. Η αξιολόγηση αυτή, βασίζεται σε 3 σημεία:

1^ο Την γενική κατηγοριοποίηση των συστημάτων σε *centralized* και *decentralized* (κεντροποιημένα και μη αντίστοιχα).

2^ο Τα γενικά χαρακτηριστικά τα οποία εν μέρει εξαρτώνται από την παραπάνω κατηγοριοποίηση. Τέτοια χαρακτηριστικά μπορεί να είναι η αξιοπιστία (*reliability*), η κλιμάκωση (*scalability*) και το κόστος (*bandwidth and storage cost*).

3^ο Τα χαρακτηριστικά τα οποία εμφανίζονται στο σύστημα και είναι απαραίτητα για τον υπολογισμό της φήμης και της αξιοπιστίας ενός κόμβου. Κάποια από αυτά είναι η τιμή της φήμης και της αξιοπιστίας που υποστηρίζει το κάθε σύστημα.

Κατηγοριοποίηση Συστημάτων

Όλα τα συστήματα τα οποία παρουσιάστηκαν, εκτός του P-Grid και του NodeRanking, ανήκουν στα μη-κεντριοποιημένα συστήματα (decentralized). Το P-Grid δεν είναι τελείως εκτός αυτής της κατηγορίας, καθώς η πληροφορία για τη φήμη ενός κόμβου, η οποία προκύπτει έπειτα από την αποθήκευση παραπόνων, δηλ. complaints, μοιράζεται στους υπολοίπους κόμβους του δικτύου μέσω μιας hash function.

Έτσι, οι κόμβοι δεν ελέγχουν πλήρως τα δεδομένα που έχουν δημιουργήσει οι ίδιοι, αλλά ο έλεγχος εκτελείται από τους κόμβους οι οποίοι είναι υπεύθυνοι για την αποθήκευσή τους, αφού εκτελεστεί η hash function.

Η ύπαρξη ενός κεντρικού κόμβου ο οποίος αρχικά δημιουργεί ένα σύνολο με τη βοήθεια της πληροφορίας που λαμβάνεται από κάθε κόμβο, σχετικά με τις σχέσεις που υπάρχουν μεταξύ αυτού και κάποιου άλλου κόμβου, είναι απαραίτητη στο NodeRanking.

Έπειτα ο κεντρικός κόμβος, κατά τη διάρκεια εκτέλεσης του αλγορίθμου, υπολογίζει την τιμή του κύρους κάθε κόμβου, την αποθηκεύει και με βάση αυτή επαναπροσδιορίζει το κοινωνικό δίκτυο μετά το τέλος του αλγορίθμου.

Bandwidth & Storage Cost

Το κόστος αναφορικά ως προς τις έννοιες του bandwidth και storage cost, αντιστοιχεί στο ποσοστό κατανάλωσης bandwidth του δικτύου καθώς επίσης και στον αποθηκευτικό χώρο που απαιτεί η λειτουργία του κάθε συστήματος. Όσον αφορά το bandwidth cost, το μοντέλο XREP παρουσιάζει το μεγαλύτερο.

Αυτό οφείλεται εν μέρει στην αποστολή των Poll Queries (για την εύρεση της φήμης ενός resource αλλά και του server) αλλά και στα TrueVotes (για την αποστολή και επιβεβαίωση των ψήφων κάθε κόμβου). Ακόμη, σχετικά με τον αποθηκευτικό χώρο, το XREP απαιτεί την αποθήκευση εγγράφων που απεικονίζουν προηγούμενες άμεσες συναλλαγές μεταξύ των κόμβων, στα repositories των resources και των server.

Αυτό έχει ως αποτέλεσμα να μην απαιτείται μεγάλος αποθηκευτικός χώρος. Ο αριθμός των μηνυμάτων είναι της τάξης του $O(\log n)$, καθώς τα μηνύματα προωθούνται ανάλογα με την πληροφορία των routing tables των n κόμβων. Επίσης, εφ' όσον έχουμε αποθήκευση μόνο των complaints (παράπονα-κακή φήμη) ο αποθηκευτικός χώρος που απαιτείται είναι πολύ μικρός.

Στο μοντέλο NICE, το bandwidth cost δεν είναι μεγάλο, διότι παρόλο που ακολουθεί ένα είδος flooding για να βρει το ισχυρότερο μονοπάτι (strongest path), στον υπολογισμό έχουμε την εισαγωγή της έννοιας της πιθανότητας, όπου μεγαλύτερη πιθανότητα ισοδυναμεί με μεγαλύτερη εμπιστοσύνη, άρα η επιλογή του επόμενου κόμβου γίνεται βασισμένη στην πιθανότητα αυτή.

Σχετικά με τον κόστος του αποθηκευτικού χώρου στο μοντέλο NICE, είναι απαραίτητη η αποθήκευση τόσο της αρνητικής, όσο και της θετικής πληροφορίας, η οποίες είναι το αποτέλεσμα των συναλλαγών μεταξύ των κόμβων, επομένως το κόστος είναι μικρό.

Σχετικά με το μοντέλο που χρησιμοποιεί το εικονικό νόμισμα, το bandwidth cost προκύπτει εφ' όσον κάποιο παράπονο δημιουργηθεί και προωθηθεί σε όλο το δίκτυο με τη μέθοδο του flooding. Σε περίπτωση μιας επιτυχούς συναλλαγής, η προώθηση των μηνυμάτων περιορίζεται σε μία μικρή ομάδα γειτόνων του κόμβου.

Συμπεραίνουμε λοιπόν ότι το bandwidth cost, θα είναι μεγάλο σε περίπτωση που έχουμε κάποιο παράπονο. Το κόστος του αποθηκευτικού χώρου είναι αρκετά μικρό, καθώς κάθε κόμβος αποθηκεύει πληροφορία σχετικά με την φήμη των μελών της ομάδας του και τη φήμη κάποιων άλλων ομάδων που γνωρίζει.

Στο REGRET ένα μικρό ποσοστό της πληροφορίας αποθηκεύεται σε κάθε κόμβο, η οποία περιέχει θετικές και αρνητικές τιμές της φήμης, οι οποίες αντιστοιχίζονται και από αυτόν αλλά και από τα υπόλοιπα μέλη της ομάδας του. Επίσης το bandwidth cost, περιορίζεται στον μικρό αριθμό μηνυμάτων που ανταλλάζουν μεταξύ τους τα μέλη της ομάδας.

Τέλος το NodeRanking είναι το δεύτερο κατά σειρά μοντέλο το οποίο έχει και αυξημένο bandwidth cost το οποίο επιβαρύνει τον κεντρικό κόμβο του δικτύου, καθώς σε αυτόν θα καταλήξουν όλα τα τελικά authorities των κόμβων μετά την εφαρμογή του αλγορίθμου, για τον επαναπροσδιορισμό του κοινωνικού δικτύου.

Αναφορικά με το storage cost, εφ' όσον κάθε κόμβος κρατά πληροφορίες μόνο για το authority των κόμβων που εμπιστεύεται (δηλαδή κόμβοι στους οποίους καταλήγει ακμή η οποία έχει αυτόν ως πηγή), συμπεραίνουμε ότι το κόστος είναι πολύ μικρό.

Scalability

Η κλιμάκωση (scalability) σε κάθε ένα από τα προαναφερθέντα συστήματα συνδέεται άμεσα με το bandwidth και το storage cost. Τα μοντέλα XREP και NodeRanking δεν θα ήταν εύκολα εφικτό να εφαρμοστούν σε δίκτυα τα οποία περιλαμβάνουν μεγάλο αριθμό κόμβων, το XREP εξαιτίας του μεγάλου bandwidth cost, ενώ το NodeRanking, επειδή προαπαιτεί έναν κεντρικό κόμβο ο οποίος θα έχει μεγάλη κατανάλωση σε bandwidth.

Σε μια τέτοια περίπτωση θα μπορούσαμε να εισάγουμε περισσότερους κεντρικούς κόμβους, ώστε να έχουμε καλύτερη ανταπόκριση στο σύστημα.

Όλα τα υπόλοιπα δίκτυα έχουν τη δυνατότητα να επεκταθούν σε δίκτυα με πολλούς κόμβους, ενώ αξίζει να σημειώσουν ότι το σύστημα το οποίο παρουσιάζει τη μεγαλύτερη κλιμάκωση είναι το σύστημα με το εικονικό νόμισμα (Reputation Framework using Currency) καθώς δημιουργήθηκε εξ' αρχής ώστε να υποστηρίζεται από δίκτυα P2P.

Reliability

Η αξιοπιστία των παραπάνω συστημάτων, ορίζεται με βάση το κατά πόσο ένα σύστημα ανταποκρίνεται σωστά σε περίπτωση που ένας ή παραπάνω «κακοί» κόμβοι προσπαθήσουν να το εκμεταλλευτούν προς όφελός τους και να επικρατήσουν ανατρέποντάς το.

Έστω ότι σε ένα σύστημα XREP μια ομάδα «κακών» peers κάνει προσπάθειες για να πείσει έναν peer να δώσει την προτίμηση του σε ένα μέλος της ομάδας τους, δίνοντας σε αυτόν όλους τους ψήφους.

Για να αντιμετωπίσει την κατάσταση το XREP, ομαδοποιεί τις ψήφους που έχουν κοινό net_id και δέχεται σαν μία ψήφο τον μέσο όρο τους. Μερικά παραδείγματα επιθέσεων που μπορεί να αντιμετωπίσει το XREP είναι το Pseudospoofing, το ID Steal και το Shilling.

Σε ένα άλλο παράδειγμα, σε ένα P-Grid σύστημα, ένας ή περισσότεροι «κακοί» peers, δίνουν λανθασμένες πληροφορίες σε έναν κόμβο ο οποίος πραγματοποιεί ερώτηση αξιολόγησης για κάποιον άλλο, που αφορούν παράπονα (complaints) για αξιολόγηση άλλου κόμβου.

Σε μία τέτοια περίπτωση, ο κόμβος που πραγματοποιεί την αξιολόγηση, θα αναζητήσει στο δίκτυο οποιοδήποτε πληροφορία υπάρχει σχετικά με παράπονα εις βάρος του κόμβου που γίνεται η αξιολόγηση. Εάν υπάρχει διαφορά στην πληροφορίας του «κακού» κόμβου και του κόμβου που κάνει την αναζήτηση, ο «κακός» κόμβος θα αποκαλύψει τις προθέσεις του.

Σε μια διαφορετική περίπτωση όπου μια ομάδα από «κακούς» χρήστες, χρησιμοποιεί παράπονα για να αναδείξει έναν κόμβο, το σύστημα ανταποκρίνεται με τον ίδιο σχεδόν τρόπο.

Ο κόμβος που αναζητά πληροφορία για την αξιολόγηση ενός άλλου, ζητά και να αξιολογηθούν οι κόμβοι οι οποίοι καταχώρησαν τα παράπονα, δηλαδή οποιαδήποτε καταχώρηση παραπόνου έχει γίνει από τους κόμβους του δικτύου εις βάρος τους.

Σε περίπτωση που ο αριθμός των παραπόνων που έχουν καταχωρήσει οι «κακοί» κόμβοι, σε συνδυασμό με τον αριθμό των παραπόνων που έχουν υποβάλλει οι άλλοι κόμβοι γι' αυτούς είναι μεγάλος, τότε είναι πολύ πιθανό οι κόμβοι να είναι «κακοί».

Σε ένα σύστημα NICE, αυτό που προσπαθούν να πετύχουν οι «κακοί» κόμβοι, είναι να πετύχουν την εισαγωγή τους στις λίστες προτιμήσεων (preference lists) των άλλων κόμβων, ανταλλάσσοντας μεταξύ τους τιμές εμπιστοσύνης με μεγάλο βαθμό.

Ακόμη και να καταφέρουν όλα ή τα περισσότερα μέλη μιας «κακής» ομάδας να εισχωρήσουν στις λίστες κάποιων «καλών» κόμβων, έπειτα από την πρώτη αποτυχημένη συναλλαγή μαζί τους, οι «καλοί» κόμβοι θα τους διαγράψουν από τη δική τους λίστα και θα αναζητήσουν άλλους κόμβους με τους οποίους μπορούν να κάνουν συναλλαγή.

Αυτό οφείλεται στο γεγονός ότι οι «κακοί» κόμβοι δεν έχουν τη δυνατότητα να σταματήσουν την επικοινωνία μεταξύ δύο κόμβων. Έτσι σε κάποια φάση ένας «καλός» κόμβος θα μπορέσει να επικοινωνήσει με έναν άλλο «καλό» κόμβο.

Το σύστημα με την μεγαλύτερη αξιοπιστία εναντίον μιας απόπειρας ανατροπής του (ομαδοποιημένης ή μη), είναι αυτό με τον εικονικό χρηματισμό (Reputation Framework using Currency). Έστω ότι όλα τα μέλη μιας ομάδας θελήσουν να μειώσουν τη φήμη σε έναν ή περισσότερους δημοφιλείς (high reputated) κόμβους τους δικτύου, με την αποστολή πολλών παραπόνων εναντίον τους.

Κάθε φορά που αποστέλλεται ένα παράπονο που αφορά έναν κόμβο, η φήμη και των δύο κόμβων μειώνεται κατά ένα ποσοστό, ανάλογο με την φήμη που έχει ο άλλος κόμβος.

Έτσι όταν ο «κακός» κόμβος στείλει ένα παράπονο για έναν «καλό», από τον πρώτο θα αφαιρεθεί το ποσοστό της φήμης που έχει ο δεύτερος. Άρα όσο μεγάλη και να είναι η φήμη του «κακού» κόμβου αρχικά, εάν συνεχίζει να διαδίδει παράπονα θα αποκτήσει αρνητική φήμη, καθώς η τιμή της θα μειώνεται συνέχεια, ανάλογα με τον αριθμό των κόμβων τους οποίους θέλει να βλάψει.

Στην περίπτωση του REGRET, το σύστημα μπορεί να αντιμετωπίσει μία ομάδα «κακών» κόμβων και να παρέχει αξιοπιστία, μόνο εάν το πλήθος των «καλών» κόμβων είναι μεγαλύτερο από το πλήθος των κακών.

Σε αντίθετη περίπτωση, δηλαδή όταν ένας κόμβος βασίζεται στην άποψη του συνόλου του, σε θέμα εμπιστοσύνης η απόφασή του θα είναι λανθασμένη, καθώς βασίζεται σε απόψεις «κακών» κόμβων.

Το NodeRanking υστερεί σημαντικά σε θέματα αξιοπιστίας, διότι υπάρχει ένας μόνο κεντρικός κόμβος με τον οποίο επικοινωνούν οι υπόλοιποι για να αποκτήσουν πληροφορίες σχετικά με τη δικαιοδοσία των κόμβων.

Για την αξιολόγηση των συστημάτων, θα γίνει ομαδοποίηση ανάλογα με τον τύπο και τις τιμές που παρέχει κάθε ένα στους κόμβους. Το σύστημα XREP χρησιμοποιεί μόνο θετικές τιμές για τις ψήφους, καθώς το μήνυμα στέλνεται μόνο για τα resources των servers που εμπιστεύεται κάθε κόμβος. Αντίστοιχα στο P-GRID, ο κόμβος καταχωρεί complaints μόνο για τους κόμβους με τους οποίους είχε μια αποτυχημένη συναλλαγή.

Μετά το τέλος του αλγορίθμου η εμπιστοσύνη μπορεί να πάρει διακριτές τιμές από $\{-1,0,1\}$. Στο σύστημα NICE η αξιολόγηση πραγματοποιείται δημιουργώντας αρνητικά και θετικά cookies από κάποιον κόμβο με τον οποίο υπήρξε συναλλαγή.

Για να αποτιμηθεί η φήμη ενός κόμβου ως προς κάποιον άλλο, καθώς επίσης και το ποσοστό της εμπιστοσύνης που δείχνει σε αυτόν ο κόμβος που τον αξιολόγησε, χρησιμοποιούνται τιμές από το διάστημα $[0,1]$. Για τα συστήματα REGRET και Reputation Framework Using Currency, οι τιμές που αποδίδονται είναι θετικές αλλά και αρνητικές, για μία επιτυχημένη ή μια αποτυχημένη συναλλαγή αντίστοιχα και ανήκουν στο διάστημα $[-1,1]$.

Τέλος, στο NodeRanking επειδή η εμπιστοσύνη προκύπτει από την πρόσθεση της αρχικής τιμής, η οποία δίνεται σε όλους τους κόμβους κατά την κατασκευή του δικτύου, με ένα ποσοστό της εμπιστοσύνης των κόμβων που καταλήγουν σε αυτόν, οι τιμές που λαμβάνει είναι θετικές και ανήκουν στο διάστημα $[1, +\infty]$, καθώς θεωρούμε ότι η αρχική τιμή της εμπιστοσύνης όλων των κόμβων είναι ίση ή μεγαλύτερη του μηδενός.

Αξίζει να σημειωθεί ότι κανένα από τα παραπάνω μοντέλα δεν παρέχει ανωνυμία στους χρήστες. Η ανωνυμία λειτουργεί αρνητικά ως προς την επίτευξη εμπιστοσύνης μεταξύ κόμβων και χρηστών, καθώς για να εμπιστευτεί κάποιον θα πρέπει είτε να τον γνωρίζει ο ίδιος, είτε να υπάρχει ένας κοινός έμπιστος γνωστός.

7.8 Συμπέρασμα

Τα συστήματα διαχείρισης εμπιστοσύνης εκμεταλλεύονται τη φήμη που αποκτά ένας peer εντός του δικτύου, σύμφωνα με την αξιολόγηση των συναλλαγών που έχει πραγματοποιήσει. Σκοπός της εργασίας είναι η παρουσίαση και αξιολόγηση των συστημάτων διαχείρισης εμπιστοσύνης σε ένα P2P δίκτυο, ως παράμετρο της ασφάλειας που θα πρέπει να παρέχεται στους χρήστες, ενάντια στις κακόβουλες επιθέσεις άλλων χρηστών.

Κάθε σύστημα κατηγοριοποιείται με βάση την πληροφορία που εκμεταλλεύεται. Έτσι έχουμε δύο κύριες κατηγορίες. Στα reputation based trust management systems, λαμβάνεται υπ' όψη η φήμη που έχει αποκτήσει ένας κόμβος έπειτα από τις συναλλαγές του.

Η απόφαση για το αν ένας κόμβος εμπιστευτεί έναν άλλο για να ολοκληρωθεί η συναλλαγή, λαμβάνεται έπειτα από την αξιολόγηση (trust-φήμη) του κόμβου αυτού από τους υπόλοιπους κόμβους του δικτύου. Κάθε μοντέλο χρησιμοποιεί και διαφορετική τεχνική, όπως τη δημιουργία παραπόνων έπειτα από μία αποτυχημένη συναλλαγή η οποία μειώνει τη φήμη του κόμβου που την πραγματοποίησε, την ψηφοφορία των κόμβων του δικτύου για τον κόμβο που παρέχει το resource αλλά και για το resource που παρέχεται, την κατασκευή εικονικού νομίσματος, όπου ο πλούτος κάθε κόμβου αντιπροσωπεύει το ποσοστό της φήμης του, άρα και την εμπιστοσύνη που δείχνουν οι υπόλοιποι peers του δικτύου προς αυτόν έπειτα από μια επιτυχημένη συναλλαγή.

Το καλύτερο σύστημα διαχείρισης εμπιστοσύνης για ένα P2P δίκτυο είναι αυτό με το εικονικό νόμισμα.

Στην δεύτερη κατηγορία, τα social-based network trust management systems, τα μοντέλα χρησιμοποιούν τους κοινωνικούς δεσμούς που δημιουργούνται μεταξύ των

κόμβων του δικτύου. Οι δεσμοί αυτοί μπορεί να είναι δεσμοί φιλίας, αμοιβαίας υποστήριξης κ.α. μεταξύ των κόμβων που δημιουργούν σύνολα, groups.

Για να λάβει ένας κόμβος την απόφαση εάν θα εμπιστευτεί έναν άλλον για τη συναλλαγή, πέρα από την υποκειμενική του άποψη, παίζει σημαντικό ρόλο και η άποψη του συνόλου στο οποίο ανήκει ο κόμβος που κάνει την αξιολόγηση.

Σύμφωνα με τη μελέτη των συστημάτων, παρατηρήθηκε ότι η διατήρηση της ανωνυμίας ως ιδιότητα των P2P συστημάτων αποτελεί ανατρεπτικό παράγοντα για τη δημιουργία δεσμών εμπιστοσύνης μεταξύ των μελών ενός συνόλου.

Με βάση το γεγονός ότι δεν μπορεί να δημιουργηθεί εμπιστοσύνη μεταξύ δύο αγνώστων, είναι ξεκάθαρο ότι κανένα από τα μοντέλα αυτά δεν προσπαθούν να εισάγουν ή να διατηρήσουν την ανωνυμία των χρηστών.

Τέλος, είναι δυστυχώς κοινώς αποδεκτό, ότι πάντα θα υπάρχουν οι χρήστες οι οποίοι εκμεταλλεύονται τις ιδιότητες των P2P συστημάτων θα προσπαθήσουν να πλήξουν την αξιοπιστία αυτών και των χρηστών τους.

Έτσι, η δημιουργία δεσμών εμπιστοσύνης μεταξύ των κόμβων ενός P2P συστήματος αποτελεί σημαντικό παράγοντα για την ασφάλεια των συστημάτων αυτών.

ΚΕΦΑΛΑΙΟ 8

8. Τα δίκτυα ανταλλαγής αρχείων

Αρχικά το Διαδίκτυο χρησιμοποιήθηκε για να ενώσει υπολογιστές που βρίσκονται σε απόσταση, με σκοπό την αποστολή και λήψη μηνυμάτων μεταξύ τους. Καθώς όμως η τεχνολογία προχωρούσε στην ψηφιακή εποχή, όπως ήταν λογικό άρχισε και η ανταλλαγή εικόνας, βίντεο και ήχου.

Οι χρήστες εξοικειώθηκαν εύκολα με τα νέα πρότυπα ανταλλαγής δεδομένων χάρη στη δημιουργία του WWW (World Wide Web) αλλά και των IRC (Internet Relay Chat) servers. Το φαινόμενο αυτό εξαπλώθηκε ραγδαία, καθώς νέες τεχνολογίες συμπίεσης αρχείων αλλά και γρήγορες συνδέσεις Internet έκαναν την εμφάνισή τους. Και κάπου εκεί εμφανίστηκε το Napster.

Ολοένα και περισσότερα προγράμματα ανταλλαγής αρχείων peer to peer έχουν αρχίσει να κάνουν την εμφάνισή τους καθημερινά από τότε. Ακολουθεί η ιστορία των δικτύων ανταλλαγής αρχείων με αναφορές στα πρωτόκολλα που χρησιμοποιούν και συνδέσεις με τα προγράμματα που τα αξιοποιούν.

8.1 Napster

Ένας 19χρονος φοιτητής του Πανεπιστημίου Northeastern της Βοστώνης, ο Shawn Fanning, το φθινόπωρο του 1999 δημιούργησε στην εστία του την αρχική έκδοση του Napster, με σκοπό να βοηθήσει τους συμφοιτητές του να ανταλλάζουν αρχεία mp3 μεταξύ τους, καθώς οι συνδέσεις του πανεπιστημίου ήταν πολύ γρήγορες και παρέμεναν αναξιποίητες. Το πρόγραμμα είχε τεράστια επιτυχία όχι μόνο στο πανεπιστήμιο αλλά και σε όλο το διαδίκτυο και αυτό είχε σαν αποτέλεσμα ο Fanning να εγκαταλείψει το κολλέγιο και να ανοίξει την επιχείρησή του με σκοπό να προωθήσει το Napster.

Το Napster από τότε έμεινε στην ιστορία για την επανάσταση στην ανταλλαγή αρχείων μέσω Διαδικτύου αλλά επίσης και για την αναπόφευκτη απότομη πτώση του. Αυτό συνέβει όταν οι μεγάλες εταιρίες παραγωγής μουσικής κατάλαβαν πως ένα πρόγραμμα έκανε την ανταλλαγή αρχείων μουσικής mp3 πολύ εύκολη και άρχισαν τις μηνύσεις εναντίον του μέσω του συλλόγου τους (RIAA). Ο δημιουργός του Napster, απτόητος προσέλαβε προγραμματιστές για να βελτιώσουν το πρόγραμμά του, με σκοπό την κυκλοφορία του Napster 2.0, μία βελτιωμένη έκδοση του αρχικού προγράμματος.

Τα πράγματα δυσκόλεψαν το 2000, όταν πολλοί καλλιτέχνες στράφηκαν εναντίον του Napster, καθώς πολλά ακυκλοφόρητα τραγούδια έκαναν την εμφάνισή τους παράνομα

στο πρόγραμμα. Σαν πρώτη αντίδραση από την εταιρία ήταν η διαγραφή όλων των χρηστών που είχαν κατεβάσει κάποιο παράνομο κομμάτι. Αυτό δεν ήταν αρκετό όμως για να κατευνάσει τα πνεύματα. Το Napster λειτουργικά δεν ήταν ένα «καθαρόαιμο» peer to peer δίκτυο, καθώς χρησιμοποιούσε έναν κεντρικό server για την μηχανή αναζήτησής του. Αυτό είχε σαν αποτέλεσμα οι μουσικές εταιρίες να απαγορεύσουν κάποια συγκεκριμένα searches που περιείχαν ονόματα των καλλιτεχνών που εκπροσωπούσαν.

Οι μηνύσεις όμως συνέχισαν και τον Ιούλιο του 2001 σε δικαστήριο κατά του Napster αποφασίστηκε η διακοπή της λειτουργίας των server του, όπως και έγινε εν μέρη τον Οκτώβριο του 2001.

Ο Fanning στην προσπάθεια του να πληρώσει τα πρόστιμα που του αναλογούσαν δημιούργησε το Napster 3 το οποίο υποτίθεται πως θα λειτουργούσε συνδρομητικά, αλλά αυτό δεν έγινε ποτέ, καθώς καμία εταιρία δεν του έδινε άδεια για να πουλήσει τα τραγούδια.

Το Napster τελικά έκλεισε τον Μαΐο του 2002 και έγινε μία προσπάθεια να αγοραστεί από την εταιρία Berelmann AG προς 8.000.000 δολάρια, αλλά το δικαστήριο δεν το επέτρεψε και η επιχείρηση δήλωσε τελικά πτώχευση και βγήκε στον πλειστηριασμό οπου και αγοράστηκε από την Roxio Inc. η οποία χρησιμοποίησε τα ίδια λογότυπα αλλά δικές τις τεχνολογίες, δημιούργησε ένα συνδρομητικό πρόγραμμα μουσικής με τίτλο Napster 2.

Το πρόγραμμα αυτό λειτουργεί μέχρι και σήμερα έχοντας αρκετούς συνδρομητές, όχι όμως όσους είχε φτάσει το αρχικό διάστημα λειτουργίας του και αυτό οφείλεται κυρίως στην μεγάλη εξάπλωση παρόμοιων προγραμμάτων.

8.2 OpenNap

Ένα από τα πρώτα προγράμματα που ακολούθησαν ήταν το OpenNap, όταν στις 10 Ιανουαρίου του 2000 ένας χρήστης με το ψευδώνυμο drscholl ανέβασε στη σελίδα του SourceForge ένα πρωτόκολλο προσομοιώνοντας τις δυνατότητες του Napster αλλά με την δυνατότητα να μοιράζει οποιοδήποτε τύπου αρχεία και με τους servers να συνδέονται μεταξύ τους.

Ονομάστηκε OpenNap επειδή έμοιαζε με το Napster αλλά ο κώδικας ήταν προσβάσιμος σε όλους και ανοιχτός σε επεκτάσεις. Οι servers που βασίζονταν σε αυτό το πρωτόκολλο είχαν τη δυνατότητα να χρησιμοποιηθούν και από το Napster χρησιμοποιώντας το Napigator, το οποίο άλλαζε τις διευθύνσεις με τις οποίες συνδέεται το πρόγραμμα.

Υπήρξαν πολλοί clients που βασίστηκαν σε αυτό το δίκτυο, το πιο διαδεδομένο πρόγραμμα όμως ήταν το WinMX. Ανήκε στην εταιρία Frontcode Technologies και στην αρχική του έκδοση υποστήριζε αποκλειστικά το δίκτυο OpenNap.

Στις επόμενες εκδόσεις η εταιρία δημιούργησε το δικό της πρωτόκολλο με το όνομα WPNP (WinMX Peer Network Protocol) το οποίο υποστήριζε παράλληλα με το OpenNap. Μια διαφορετική έκδοση του OpenNap έκανε την εμφάνισή της στις 26 Μαΐου 2002 με τον τίτλο Opennap NG (New Generation), σαν μία προέκταση του αρχικού από κάποια άτομα της εταιρίας καθώς ο drscholl σταμάτησε να ασχολείται με αυτό. Το OpenNap και το NG μπορούν να χρησιμοποιηθούν από τους ίδιους clients.

8.3 Gnutella

Μετά την πρώτη εμφάνιση του Napster η AOL ανέθεσε στους προγραμματιστές της να κατασκευάσουν κάτι παρόμοιο. 2 υπάλληλοι της Nullsoft, ο Justin Frankel και ο Tom Pepper δημιούργησαν ένα καθαρά peer to peer πρωτόκολλο το οποίο και διέθεσαν δωρεάν στο Διαδίκτυο.

Στις 14 Μαρτίου 2000, η AOL φοβούμενη την προϊστορία του Napster θέλησε να το αποσύρει. Ήταν όμως πολύ αργά καθώς προγραμματιστές με πολύ καλές ικανότητες είχαν αποκτήσει το πρωτόκολλο και κατάφεραν να δημιουργήσουν ένα πρόγραμμα εκ νέου προσφέροντας το επίσης δωρεάν, αλλά αυτή τη φορά με ανοιχτό κώδικα και με το όνομα Gnutella.

Το όνομα αυτό προήλθε από το GNU και το Nutella, όσο κι αν φαίνεται περίεργο, και αυτό καθώς οι δημιουργοί της Nullsoft θα το προσέφεραν με την GNU άδεια (κάτι το οποίο δεν έγινε ποτέ) αλλά και επειδή κατανάλωσαν πολύ Nutella όσο έγραφαν το πρόγραμμα.

Αυτό που το διαφοροποιεί από το Napster και το OpenNap είναι ότι δεν χρησιμοποιεί κεντρικούς servers αλλά δρα αποκλειστικά βάση peer to peer. Έτσι κάθε υπολογιστής

εξυπηρετεί το δίκτυο και σαν client αλλά και σαν server, βοηθώντας έτσι και τους άλλους να συνδεθούν αλλά και στην αναζήτηση δικτύου.

Αυτό καθιστά δύσκολο το να κλείσει το δίκτυο καθώς δεν υπάρχει κεντρική διαχείριση ακόμα και αν κάποιοι χρήστες είναι εκτός δικτύου, κάποιοι άλλοι θα πάρουν τη θέση τους. Το δίκτυο έγινε εναλλακτικό του OpenNap, χάρη στη χρήση των γνωστών Limewire, Morpheus καθώς επίσης και του MLdonkey, όμως λόγω κατασκευής το δίκτυο ήταν ιδιαίτερα αργό. Αυτό είχε ως συνέπεια την κυκλοφορία του Gnutella 2 (26 Μαρτίου 2003), το οποίο χρησιμοποιούσε διαφορετικές παραμέτρους με σκοπό τη γρηγορότερη αναζήτηση.

Οι περισσότεροι δημιουργοί παρόμοιων προγραμμάτων το αππέριπταν, παρόλο που έγινε γνωστό μέσα από το Shareaza, καθώς ο Strokes έδωσε το όνομα Gnutella χωρίς την άδεια του οργανισμού (GDF – Gnutella Developers Forum) που ήταν υπεύθυνος για την επέκταση του πρωτοκόλλου. Έτσι κάποιο το αποκαλούσαν απλά MP (Mike's Protocol). Το Gnutella και το Gnutella 2 είναι από τα μεγαλύτερα δίκτυα στον κόσμο, ενώ αρκετοί είναι αυτοί που τα συνδιάζουν και με άλλα μεγάλα δίκτυα.

8.4 FastTrack

Αναπτύχθηκε από την εταιρεία Kazaa BV από τον σουηδό Zennstrom Niklas, με τον τίτλο Fast Track και κυκλοφόρησε το Μάρτιο του 2001. Βασισμένο στο ανοιχτό πρωτόκολλο Gnutella, ήταν σχεδιασμένο σαν επέκταση του Gnutella και καταστούσε εφικτή την λήψη ενός αρχείου από πολλές πηγές ταυτόχρονα, μειώνοντας έτσι τον χρόνο λήψης του αρχείου. Μετά από μηνύσεις εναντίον της εταιρίας, εξαγοράστηκε από διάφορες Αμερικάνικες επιχειρήσεις, με βασικό ιδιοκτήτη την Sharman Networks, η οποία και επέκτεινε ακόμα τα πρωτόκολλα Kazaa και FastTrack, με άσχημες συνέπειες για την κοινότητα.

Το Morpheus έπειτα από τη συγχώνευση του με το MusicCity (όταν έκλεισε το δεύτερο), βασίστηκε στο πρωτόκολλο FastTrack. Έπειτα από τις αλλαγές της Sharman Networks, το Morpheus και όλοι οι χρήστες μέσα από αυτό, κλειδώθηκαν εκτός και αναγκαστικά στράφηκαν σε άλλου τύπου δίκτυα και το μόνο δίκτυο που είχε πρόσβαση πλέον ήταν το Kazaa. Το Morpheus υιοθέτησε το Gnutella και το Gnutella 2 εν συνεχεία, ενώ ανέπτυξε και ένα δικό του δίκτυο διαφορετικού τύπου με την ονομασία NEOnet. Στην επόμενη έκδοση του FastTrack, η Sharman Networks επέτρεψε την πρόσβαση και στα υπόλοιπα προγράμματα και το Morpheus συνδεόταν πλέον ξανά. Το Kazaa ήταν ένα από τα πρώτα προγράμματα που χρησιμοποιούσε γνωστά ad-ware και spyware.

Η Sharman Networks απέκτησε ξαφνικά ενοχλητικά προγράμματα με τη δυνατότητα καταγραφής των επιλογών του χρήστη και τη μετάδοση αυτών σε διαφημιστικές εταιρίες, κυρίως στατιστικές, καθυστερώντας έτσι σημαντικά τις συνδέσεις όλων των χρηστών.

Το παράδειγμα αυτό ακολούθησαν και άλλα προγράμματα δυστυχώς, πιστεύοντας ότι θα κερδίσουν μέσα από αυτά. Άλλο ένα σημαντικό πρόβλημα με το Kazaa, ήταν η

αρχική σχεδίασή του, καθώς επέτρεπε πολύ εύκολα σε κάποιον να διαμοιράσει ένα κακής ποιότητας κομμάτι, χωρίς αυτό να ανιχνεύεται από το πρόγραμμα, με αποτέλεσμα την διάδοσή του.

Οι δισκογραφικές εταιρίες, παράλληλα με τις εκατοντάδες μηνύσεις εναντίων των p2p προγραμμάτων, βρήκαν την ευκαιρία να γεμίσουν το δίκτυο με τέτοια κομμάτια, ρίχνοντας έτσι την ποιότητα του δικτύου. Εν συνεχεία, διάφορες εναλλακτικές εκδόσεις των προγραμμάτων (Kazaa Lite) κυκλοφόρησαν, χωρίς την παρουσία των διαφημιστικών spyware, όμως τα περισσότερα προγράμματα συνέχισαν να γεμίζουν τους υπολογιστές με αυτά, πολλές φορές αφήνοντας τρύπες για ιούς.

8.5 OpenFT

Ως εναλλαγή του FastTrack κυκλοφόρησε το OpenFT, κάτι παρόμοιο με το Napster και το OpenNap, δηλ. μια ανοιχτού κώδικα έκδοση του FastTrack, με αρκετές βελτιώσεις όμως. Αξιοποιώντας τα κύρια χαρακτηριστικά των FastTrack, OpenNap και Gnutella, δημιουργείται ένα πολύ πιο δυνατό δίκτυο. Το πρωτόκολλο αυτό έχει τις ρίζες του από το gift project το οποίο ακόμη εξελίσσεται.

8.6 eDonkey και OverNet

Το πρωτόκολλο αυτό πρωτοεμφανίστηκε στα μέσα του 2001, από την εταιρί MetaMachine, και ενώ δεν ήταν peer to peer, ακολουθούσε τη φιλοσοφία του client – server. Το πασίγνωστο eDonkey2000 διαδώθηκε πολύ γρήγορα σαν ένα νέο εναλλακτικό δίκτυο. Πολύ σύντομα όμως ανακοινώθηκε από την ίδια εταιρία το OverNet σαν ένα καθαρόαιμο p2p δίκτυο αλλά πιο γρήγορο από το eDonkey, χάρη στους νέους αλγόριθμους αναζήτησης. Αργότερα η MetaMachine ένωσε τα δύο αυτά δίκτυα συνδιάζοντας τα καλύτερα χαρακτηριστικά τους. Σε αυτό το δίκτυο συνδέονταν πολλά δίκτυα με πιο γνωστά το eMule και το MLdonkey.

8.7 SoulSeek

Έκανε την εμφάνισή του το τέλος του 2001 και η φιλοσοφία του θύμιζε αρκετά εκείνη του Napster, καθώς ο κεντρικός server αναλάμβανε όλα τα search. Το δίκτυο ήταν γεμάτο κομμάτια underground ηλεκτρονικής μουσικής με τα περισσότερα να είναι δημιουργίες των χρηστών οι οποίοι τα μοιράζονταν ελεύθερα. Είχε φτάσει και σε σημείο να αποτελεί το βασικό πρόγραμμα διαμοιρασμού των μελών του IDM (Intelligent Dance Music) Mailing List.

Όμως μετά το κλείσιμο του AudioGalaxy, το μεγαλύτερο μέρος των χρηστών που είχαν φύγει από δίκτυα που έκλεισαν, άρχισε να καταλαμβάνει το δίκτυο με αποτέλεσμα να γεμίσει παράνομα κομμάτια. Σε αντίθεση με τις προβλέψεις όμως το 2002 η εταιρία άνοιξε την δική της δισκογραφική με παραγωγή ηλεκτρονικών albums και όχι μόνο.

8.8 Bittorent

Το δημιούργησε το 2002 ο Bram Cohen και έκανε την πρώτη του εμφάνιση στο CodeCon , ενώ από τότε φημίζεται για το νόμιμο αλλά και το παράνομο downloading. Ένα ακόμη αξιόλογο και ιδιόμορφο χαρακτηριστικό, είναι ότι δεν υπάρχει ένας κεντρικός server, αλλά αρχεία με την κατάληξη .torrent που κυκλοφορούν στο Διαδίκτυο και είναι διαθέσιμα στον καθένα. Τα αρχεία αυτά συνδέονται σε μία συγκεκριμένη διεύθυνση για να ανακαλύψουν αυτούς που τα προσφέρουν (seeders) αλλά και αυτούς που τα κατεβάζουν (leechers).

Απαραίτητη προϋπόθεση για τον διαμοιρασμό του αρχείου είναι να υπάρχει τουλάχιστον ένας seeder ο οποίος θα έχει ολοκληρω το αρχείο. Ο διαμοιρασμός ξεκινάει από τη στιγμή που οι leechers συνδεθούν με τους seeders. Κάθε ένας κατεβάσει ένα μικρό τμήμα τυχαία από τον καθένα και στη συνέχεια τα μοιράζει με τους υπόλοιπους ανάλογα με το ποιο χρειάζεται ο καθένας.

Η αποστολή και η λήψη είναι ταυτόχρονη και αμοιβαία μεταξύ των seeders και των leechers με αποτέλεσμα να παρατηρείται το φαινόμενο distributed downloading. Τα αρχεία torrent είναι μια ανασκόπηση του γνήσιου αρχείου που είναι διαθέσιμο για κατέβασμα και τα δημιουργεί ο αρχικός seeder που τα προσφέρει.

Αν το πρόγραμμα δεχθεί λάθος δεδομένα σε σχέση με το αρχικό torrent τα απορρίπτει, ενώ σε περίπτωση που όλα τα τμήματα του αρχείου είναι σωστά, το κατεβάζει επιτυχώς. Μεγάλες σελίδες χρησιμοποιούν το BitTorrent με σκοπό να γλιτώσουν την υπερφόρτωση των server τους κατά το μαζικό download από τους χρήστες. Πολλές είναι εκείνες που προσέφεραν μόνο νόμιμα αρχεία, όμως κυκλοφορούν πολλές με πειρατικό περιεχόμενο.

8.9 Direct Connect

Μια φρέσκια ιδέα παρουσιάστηκε το 1999 από τη NeoModus, τότε που το Napster έκανε τα πρώτα του βήματα και ακόμα δεν είχαν ξεκινήσει οι δικαστικές διαμάχες με τις δισκογραφικές. Η φιλοσοφία βασιζόταν στη δημιουργία ενός πρωτοκόλλου που θα οργάνωνε τους χρήστες σε hubs αντί για servers. Το Direct Connect είναι ευρέως διαδεδομένο ακόμα και σήμερα, κυρίως επειδή δημιουργούνται συνεχώς μικρές κοινότητες απο διαμοιραστές. Τα hubs αναρτούνται μέσα σε λίστες στο Διαδίκτυο και το καθένα θέτει τους δικούς του όρους για να δεχτεί τα μέλη του. Για παράδειγμα, ένα hub απαιτεί από τον κάθε χρήστη να προσφέρει για διαμοιρασμό X MBs. Εφ'όσον ένας χρήστης πληρεί τις προϋποθέσεις για να γίνει μέλος, μπορεί να συνδεθεί με το hub και να επικοινωνήσει με τα υπόλοιπα μέλη του. Κάθε ένα από αυτά ανήκει σε έναν server, οπότε το μέγεθος του δικτύου το καθιστά σαν ένα εναλλακτικό IRC δίκτυο ανταλλαγής αρχείων. Οι πιο γνωστοί clients είναι το DC++ και το DC Pro. Ο όγκος των δεδομένων που είναι διαθέσιμα προς ανταλλαγή, καθιστά το δίκτυο αυτό ένα από τα μεγαλύτερα που υπάρχουν.

8.10 Συνδρομητικά Δίκτυα

Όπως είναι λογικό, δημιουργήθηκαν και αρκετά συνδρομητικά δίκτυα, τα οποία απαιτούν πληρωμή για την σύνδεση και τη χρήση του. Χρησιμοποιώντας δικά τους πρωτόκολλα και ιδιωτικούς servers προσφέρουν τραγούδια ή αρχεία κάθε είδους επί πληρωμή στα μέλη τους. Γνωστά τέτοια προγράμματα είναι το Napster 2, iTunes, Rhapsody, MusicMatch κ.α.

Τα δίκτυα αυτά έχουν μεγάλο αριθμό χρηστών, όμως πολλές φορές απογοητεύουν τα μέλη τους με τους περιορισμούς που εφαρμόζουν στα αρχεία αυτά, καθώς πολλές εταιρίες επιλέγουν να κλειδώνουν τα κομμάτια ώστε να μην είναι δυνατόν να χρησιμοποιηθούν σε άλλον υπολογιστή, πέρα του αρχικού κατόχου.

Οι κερδισμένες εταιρίες είναι λίγες πλέον και σκοπός τους είναι με μηνύσεις και δικαστικές διαμάχες να καταπολεμήσουν τα ελεύθερα προγράμματα προς όφελός τους. Μένει μόνο να δούμε πόσο καιρό θα αντέξουν και οι δύο πλευρές.

8.11 Δημοφιλέστερα προγράμματα P2P

Ακολουθεί μια μικρή παρουσίαση των δημοφιλέστερων προγραμμάτων ανταλλαγής P2P, με την διεύθυνση από την οποία μπορεί κανείς να το κατεβάσει.

Vuze

Είναι η επόμενη έκδοση του γνωστού προγράμματος *Azureus*, ενός από τα πιο δημοφιλείς προγράμματα ανταλλαγής αρχείων για δίκτυα P2P. Είναι γραμμένο σε γλώσσα *Java* και χρησιμοποιεί την μηχανή του *Azureus*. Οι χρήστες μπορούν μέσω του *Vuze* να προβάλλουν, να δημοσιεύουν και να μοιράζονται γνήσιας ποιότητας DVD και ταινίες σε HD ποιότητα. Το περιεχόμενο διαμοιράζεται μέσα από κανάλια και κατηγορίες, που περιλαμβάνουν εκπομπές τηλεόρασης, μουσικά βίντεο κλιπ, ταινίες, παιχνίδια κ.α. Ακόμη οι χρήστες μπορούν να διαμοιράσουν και να δημοσιεύσουν το δικό τους υλικό και αν το επιθυμούν ακόμη και να το πουλήσουν προς χρηματικό όφελος.

<http://www.vuze.com/>

Tixati

Ένα γρήγορο πρόγραμμα ανταλλαγής αρχείων *torrent* το οποίο προέρχεται από τον αρχιτέκτονα του προγράμματος *WinMiX OpenNap*. Είναι αρκετά προχωρημένο και με αρκετές προοπτικές εξέλιξης, αφού ήδη σχεδιάζεται να προχωρήσει στο επόμενο βήμα, δηλαδή τον διαμοιρασμό αρχείων *torrent* χωρίς τη βοήθεια *tracker*. Σύνδεσμοι *magnet*, *PEX* και *DHT* λειτουργούν άψογα στο *Tixati*. Τα προηγμένα του χαρακτηριστικά και η ποικιλία των ρυθμίσεων όσον αφορά την προτεραιότητα αφήνει τους χρήστες του αρκετά ευχαριστημένους, αφού η ταχύτητα λήψης είναι εφάμιλλη των προγραμμάτων *Vuze* και *uTorrent*. Η εμφάνιση του προγράμματος με τα προηγμένα γραφήματα απεικόνισης και τους αποκλειστικούς πίνακες, κάνουν το πρόγραμμα να δείχνει μια επαγγελματική δουλειά, κατατάσσοντας το *Tixati* σε ένα από τα καλύτερα προγράμματα της κατηγορίας του σήμερα.

<http://www.tixati.com/>

uTorrent

Γνώστό επίσης και ως *micro-torrent*, είναι το πιο δημοφιλές πρόγραμμα στην κατηγορία. Περιέχει όλα τα χαρακτηριστικά που θα χρειαστεί ένας χρήστης από ένα πρόγραμμα ανταλλαγής αρχείων *torrent* και απαιτεί μόλις 1MB αποθηκευτικού χώρου και μνήμης για να εγκατασταθεί. Χρησιμοποιώντας όλα τα χαρακτηριστικά των υπολοίπων προγραμμάτων που κυκλοφορούν, καταφέρνει να τα συνδιάσει όλα έχοντας ελάχιστο αντίκτυπο στην απόδοση του υπολογιστή κατά την λειτουργία του.

<http://www.utorrent.com/>

ABC (Another Bittorent Client)

Είναι ένα αρκετά απλό προϊόν εμφανισιακά, χωρίς αυτό να του στερεί κάτι σε θέματα απόδοσης. Προσφέρει αξιόπιστία ενώ είναι εύκολα προσαρμόσιμο ώστε να εμφανίζει στατιστικά και πληροφορίες για την λήψη ή την αποστολή των αρχείων. Είναι εύκολο στη χρήση ενώ περιέχει ακόμα και λειτουργία απόστολής αρχείων με επιλογή χρονικής διακοπής, σύμφωνα με τις επιθυμίες του χρήστη, μια λειτουργία που πολλοί χρήστες επιθυμούν να υπάρχει σε ένα τέτοιο πρόγραμμα. Είναι γραμμένο σε γλώσσα Python/wxPython.

<http://pingpong-abc.sourceforge.net/>

BitComet

Ένα πολύ καλό πρόγραμμα, το οποίο όμως έχει χάσει ένα μεγάλο μέρος χρηστών εξαιτίας των uTorrent, Transmission, Vuze και Tixati. Για προχωρημένους χρήστες που δίνουν βάση στην ιδιαίτερα στην εμφάνιση, αξίζει σίγουρα να του δώσουν μια ευκαιρία. Όσον αφορά νέους χρήστες, καλό θα ήταν να ξεκινήσουν με κάποιο άλλο πρόγραμμα όπως το uTorrent ή το Vuze. Προς ενημέρωση, πολλοί ιδιωτικοί trackers έχουν αποκλείσει τον συγκεκριμένο πρόγραμμα από τα δίκτυά τους, λόγω της πεποίθησης που υπάρχει ότι το BitComet εμφανίζει λανθασμένες αναφορές σχετικά με τον λόγο αποστολής/λήψης.

<http://www.bitcomet.com/>

TurboBT

Είναι ένα πρόγραμμα που παρουσιάστηκε σε πολλαπλές γλώσσες. Είναι δωρεάν και έχει χαρακτηριστεί ως ένα σταθερό προϊόν γραμμένο σε γλώσσα Python.

<http://turbobt.sourceforge.net/indexen.htm>

Transmission Bittorent

Προορίζεται για χρήστες OS X (Mac) και Linux. Είναι το πρόγραμμα που χρησιμοποιεί τους λιγότερους πόρους από θέμα υπολογιστικής ισχύς σε σύγκριση με όλα τα υπόλοιπα. Είναι ανοιχτού κώδικα, με λιτή αλλά επαρκή εμφάνιση και είναι το αγαπημένο πρόγραμμα χρηστών Macintosh.

<http://www.transmissionbt.com/>

Deluge Torrent

Ακόμη ένα πρόγραμμα που προορίζεται για χρήστες λειτουργικού Linux και OS X. Είναι πολύ φιλικό προς τον χρήστη αλλά και προς το λειτουργικό σύστημα, ενώ έχει λαβεί πολύ καλές κριτικές από αρκετά δημοφιλή μέσα ενημέρωσης. Έχει μεταλλική εμφάνιση αλλά σε απαλό τόνο και περιέχει τα περισσότερα από τα στατιστικά στοιχεία που θα χρειαστεί ένας μέσος χρήστης. Όπως και τα περισσότερα προγράμματα, είναι δωρεάν και έχει δημιουργηθεί από προγραμματιστές που πιστεύουν στην ελεύθερη ανταλλαγή αρχείων.

<http://deluge-torrent.org/>

8.12 Παρουσίαση του τρόπου λειτουργίας ενός δικτύου P2P

Για να κατανοήσουμε καλύτερα από τεχνικής άποψης το πώς λειτουργεί ένα δίκτυο P2P, θα κάνουμε μια σύντομη αναφορά στο Gnutella, ένα πρωτόκολλο κατανεμημένης

αναζήτησης, το οποίο διαθέτει αρκετά στοιχεία επηρεασμένα από την κλασσική αναζήτηση χρησιμοποιώντας client/server, με την διαφορά ότι χρησιμοποιεί το αποκεντρωμένο μοντέλο.

Στο μοντέλο αυτό παρατηρούμε τους Gnutella servants, όπως χαρακτηριστικά αναφέρονται, οι οποίοι στην ουσία είναι clients, οι οποίοι λειτουργούν και σαν servers ταυτόχρονα. Παρέχουν διεπαφές από την πλευρά του client, με τις οποίες πραγματοποιούν αναζητήσεις σε άλλους υπολογιστές, ενώ παράλληλα δέχονται και επεξεργάζονται αιτήσεις άλλων.

Εξ'αιτίας της κατανεμημένης του φύσης, ένα δίκτυο βασισμένο σε αυτό το πρωτόκολλο δεν επηρεάζεται από λάθη (fault tolerant), με την προϋπόθεση ότι υπάρχουν κάποιοι servers οι οποίοι θα είναι σε θέση να εξυπηρετήσουν, σε περίπτωση που κάποιος τεθεί εκτός λειτουργίας.

Για την επικοινωνία των δεδομένων μεταξύ των servants υπάρχει ένα σύνολο εντολών (descriptors) καθώς επίσης και ένα σύνολο κανόνων που καθορίζουν την ανταλλαγή εντολών μεταξύ των servants. Για την έκδοση του παραδείγματος, υπάρχουν οι εξής εντολές:

Εντολές (Descriptor)	Περιγραφή
Ping	<i>Χρησιμεύει στην ενεργή αναζήτηση υπολογιστών μέσα σε ένα δίκτυο. Όταν ένας υπολογιστής δεχθεί μια εντολή Ping, θα πρέπει να απαντήσει με μία Pong.</i>
Pong	<i>Είναι η απάντηση στην εντολή Ping, στην οποία περιλαμβάνεται και η διεύθυνση ενός συνδεδεμένου servant, καθώς επίσης και σχετικές πληροφορίες για την ποσότητα των δεδομένων που υπάρχουν διαθέσιμες.</i>
Query	<i>Ο κύριος μηχανισμός αναζήτησης σε ένα κατανεμημένο δίκτυο. Όταν ένας servant δεχτεί μια εντολή Query, θα απαντήσει με QueryHit, εφ'όσον βρεθεί κάτι σύμφωνα με τα κριτήρια της αναζήτησης.</i>
QueryHit	<i>Είναι η απάντηση στην εντολή Query. Η εντολή αυτή περιλαμβάνει τις απαραίτητες πληροφορίες ώστε ο παραλήπτης να γνωρίζει τα δεδομένα που ταιριάζουν στην εντολή Query.</i>
Push	<i>Αποτελεί τον μηχανισμό που επιτρέπει σε έναν servant πίσω από firewall, να προσφέρει δεδομένα σε μορφή αρχείων στο δίκτυο</i>

Ένας servant σε ένα δίκτυο Gnutella, αναζητά στο δίκτυο κάποιον άλλον ο οποίος είναι ήδη συνδεδεμένος σε αυτό και μόλις εντοπιστεί, πραγματοποιείται μια σύνδεση TCP/IP μεταξύ τους και αποστέλλεται η παρακάτω συμβολοσειρά (σε κωδικοποίηση ASCII):

GNUTELLA CONNECT/<protocol version string>\n\n

Protocol version string → περιέχει την έκδοση του πρωτοκόλλου

Σε περίπτωση που ο server επιθυμεί να δεχτεί την σύνδεση, θα απαντήσει με την ακόλουθη συμβολοσειρά:

GNUTELLA OK\n\n

Εάν υπάρχει διαφορετική απάντηση, σημαίνει ότι ο συγκεκριμένος server δεν επιθυμεί να γίνει η σύνδεση μεταξύ τους. Ο λόγος που μπορεί να συμβεί αυτό δεν είναι μόνο ένας. Κάποιες περιπτώσεις που αυτό μπορεί να συμβεί, είναι να μην υπάρχουν διαθέσιμες θέσεις για σύνδεση ή να μην υποστηρίζεται από την ίδια έκδοση πρωτοκόλλου.

Σε περίπτωση που ένας server συνδεθεί επιτυχώς στο δίκτυο, μπορεί να επικοινωνεί με τους υπόλοιπους στέλνοντας και λαμβάνοντας εντολές αντίστοιχες του πρωτοκόλλου.

Σε κάθε εντολή, υπάρχει και η κατάλληλη επικεφαλίδα, όπου παρέχει πληροφορίες σχετικά με το ID της εντολής, το TTL, δηλ. τον αριθμό των δρομολογήσεων της εντολής μέχρι αυτή να αφαιρεθεί από το δίκτυο, τα Hops, δηλ. τον αριθμό των servers από τους οποίους πέρασε η εντολή και τέλος το payload length, δηλ. την τιμή του μεγέθους της εντολής. Για να λήξει μια εντολή, λαμβάνεται υπ'όψιν το TTL, τιμή η οποία παρακολουθείται συνεχώς, διότι σε περίπτωση που αυτή είναι μεγάλη, το δίκτυο θα γεμίσει άχρηστο φορτίο με αποτέλεσμα να μειωθεί αισθητά η απόδοσή του. Από την τιμή Payload Length, ένας server μπορεί να εντοπίσει πότε ξεκινάει μια νέα εντολή, ώστε να τις ξεχωρίζει μεταξύ τους.

Μόλις ένας server δεχτεί μια εντολή QueryHit, είναι έτοιμος να αρχίσει να κατεβάσει απευθείας κάποιο από τα αρχεία που έχουν εμφανιστεί στην αναζήτησή του. Τα αρχεία μεταφέρονται έξω από το δίκτυο, μέσω απευθείας σύνδεσης μεταξύ των δύο υπολογιστών. Τα αρχεία δεν μεταφέρονται ποτέ μέσω του δικτύου του Gnutella και για την δουλειά αυτή χρησιμοποιείται το πρωτόκολλο HTTP. Ο server ο οποίος επιθυμεί να λάβει κάποιο αρχείο, στέλνει στον server ο οποίος το διαθέτει ένα string της ακόλουθης μορφής:

GET /get/<File Index>/<File Name>/ HTTP/1.0\r\n

Connection: Keep-Alive\r\n

Range: bytes=0-\r\n

User-Agent: Gnutella\r\n

\r\n

Για παράδειγμα, για την λήψη ενός αρχείου με τα παρακάτω στοιχεία:

File Index= 3369

File Size= 287653

File Name= PAO.mp3

Τότε η διαδικασία κατεβάσματος του αρχείου μέσω του πρωτοκόλλου HTTP θα ήταν ως εξής:

```
GET /get/3369/PAO.mp3/ HTTP/1.0\r\n
```

```
Connection: Keep-Alive\r\n
```

```
Range: bytes=0-\r\n
```

```
User-Agent: Gnutella\r\n
```

```
\r\n
```

Εν συνεχεία, ο server ο οποίος θα δεχτεί την απάντηση, θα απαντήσει σύμφωνα με το πρωτόκολλο HTTP:

```
HTTP 200 OK\r\n
```

```
Server: Gnutella\r\n
```

```
Content-type: application/binary\r\n
```

```
Content-length: 287653\r\n
```

```
\r\n
```

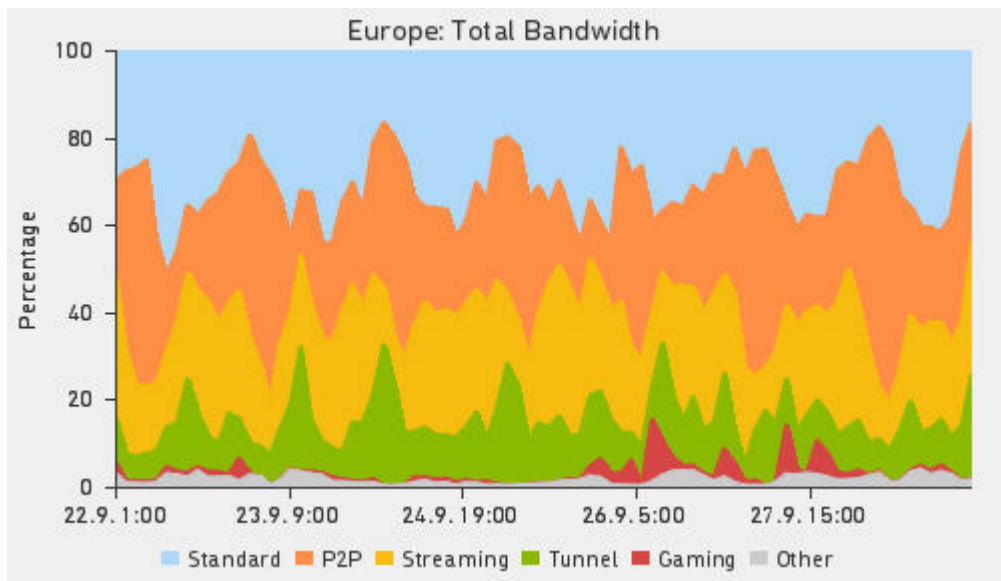
Αφού τα δεδομένα του αρχείου διαβιβαστούν μέχρι τέλος, το μέγεθος αυτός θα πρέπει να συμφωνεί με το Content-length το οποίο δόθηκε πριν από τον server. Τέλος, μέσω της παραμέτρου Range του HTTP, σε περίπτωση διακοπής κατά την μετάφορα αρχείου, η διαδικασία μπορεί να συνεχιστεί από το σημείο το οποίο σταμάτησε.

8.13 Στατιστικές έρευνες

Λόγω τις αρχιτεκτονικής των δικτύων P2P, είναι δύσκολο να υπάρχει συνεχής παρακολούθηση τις κίνησης όλων των δικτύων. Παρ'όλα αυτά όμως μια εταιρία με βάση τη Γερμανία που ειδικεύεται στην ανάπτυξη λύσεων για εφαρμογές διαχείρισης του εύρους ζώνης πάνω στα δίκτυα πανεπιστημίων και μεγάλων ISP (Internet Service Providers) δημιούργησε έναν ιστότοπο στον οποίο μπορεί κανείς να παρατηρήσει σε πραγματικό χρόνο την κατανομή του εύρους ζώνης για κάθε εφαρμογή.

Με μια γρήγορη ματιά στα στατιστικά που αφορούν την κίνηση των P2P δικτύων στην Ευρώπη, παρατηρούμε ότι καταλαμβάνουν πάνω από το ¼ της κίνησης του δικτύου και 40% του συνολικού αριθμού πακέτων που διαμοιράστηκαν. Αξίζει να σημειωθεί ότι

σχεδόν όλα προέρχονται από το δίκτυο BitTorrent, ενώ κατά την διάρκεια της νύχτας το μέρος που αντιστοιχεί στο BitTorrent ξεπερνά το 50%.



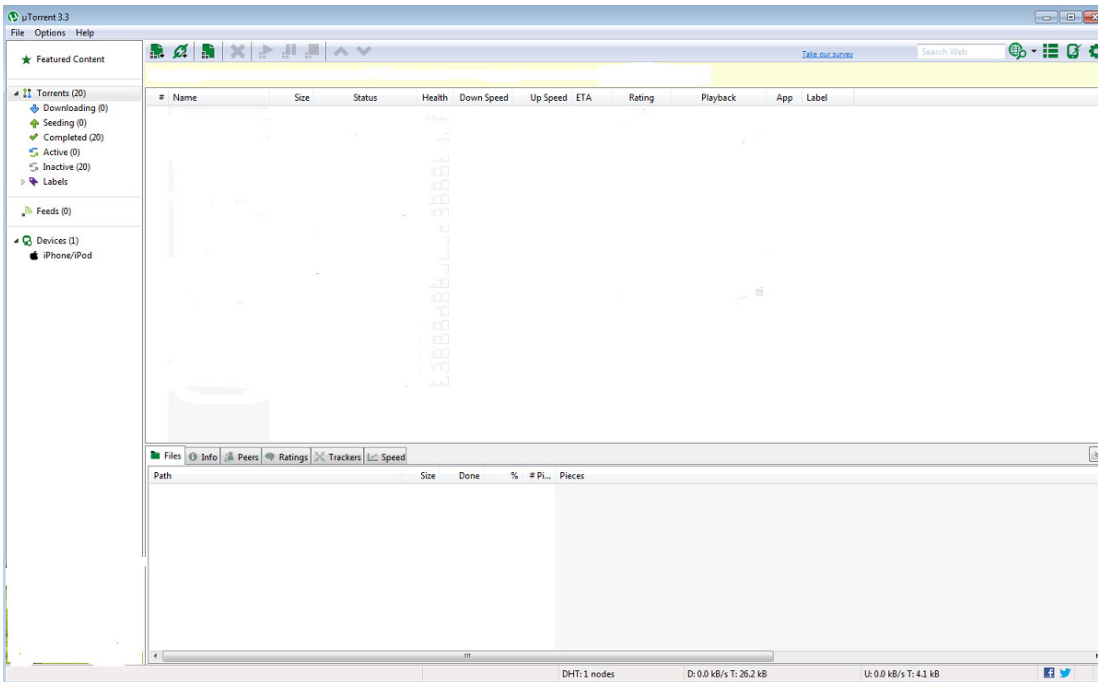
Σύμφωνα με την εταιρία, τα στατιστικά είναι διαθέσιμα μόνο για περιοχές τις Ευρώπης, ενώ στο άμεσο μέλλον θα προστεθούν και άλλες περιοχές.

9. Παρουσίαση της λειτουργίας ενός προγράμματος P2P

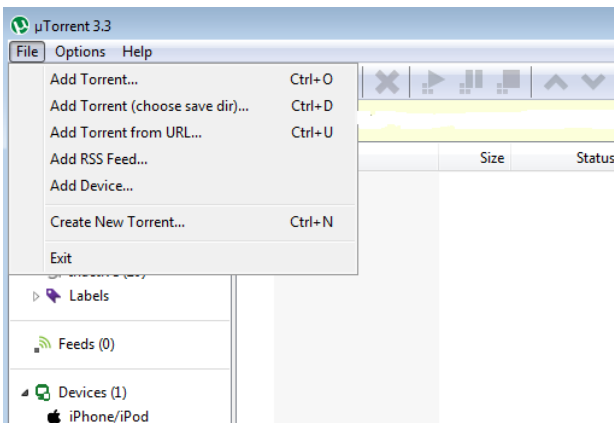
Θα ακολουθήσει μια παρουσίαση του προγράμματος uTorrent, με αναφορά στις ρυθμίσεις που μπορεί να κάνει ο χρήστης, έτσι ώστε να το προσαρμόσει στις ανάγκες του.

Αφού ο χρήστης κατεβάσει και εγκαταστήσει την εφαρμογή από την επίσημη ιστοσελίδα (<http://www.utorrent.com>), μπορεί πλέον να διαχειριστεί αρχεία τύπου .torrent ή magnet.

Ανοίγοντας την εφαρμογή, ο χρήστης βλέπει τη βασική επιφάνεια εργασίας της εφαρμογής όπως φαίνεται στην παρακάτω εικόνα.

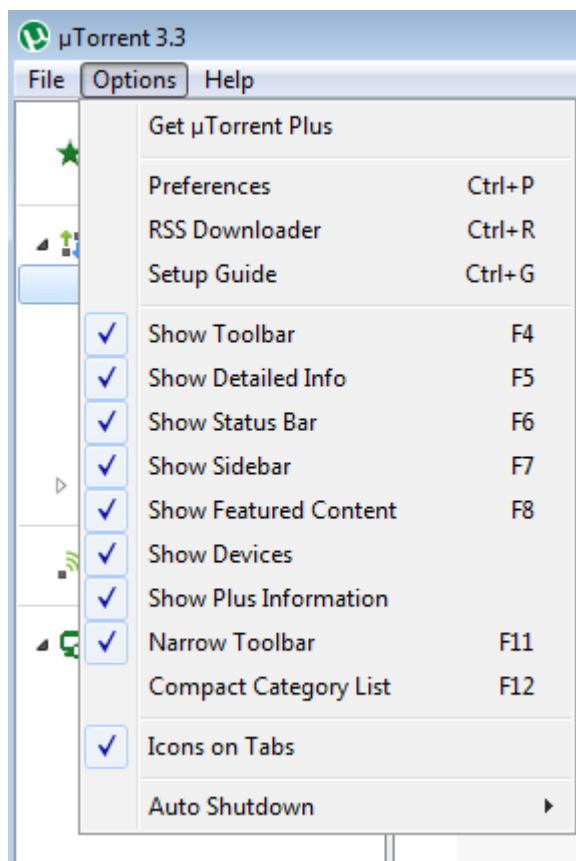


Μέσα από το απλό αλλά εύχρηστο μενού, ο χρήστης μπορεί να προσθέσει ένα ή περισσότερα αρχεία για να κατεβάσει στον υπολογιστή του ή ακόμα και να διαμοιράσει ένα δικό αρχείο στους υπόλοιπους χρήστες.

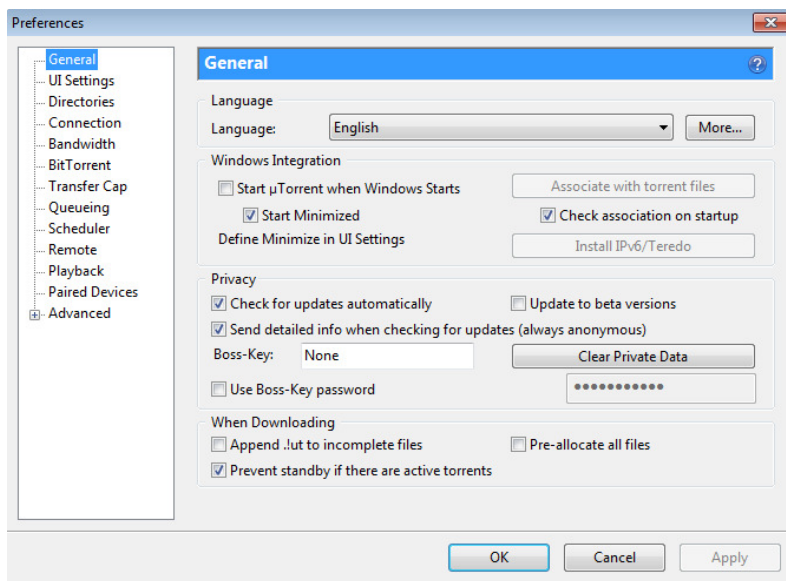


Υπάρχει ακόμα και η επιλογή για σύνδεση ενός smartphone ή άλλης υποστηριζόμενης συσκευής με την εφαρμογή για κοινή χρήση και άμεση ανταλλαγή των αρχείων.

Από το μενού των επιλογών, μπορούμε να επιλέξουμε τις πληροφορίες που θέλουμε να εμφανίζονται στην αρχική οθόνη του προγράμματος, να ρυθμίσουμε το αυτόματο κλείσιμο της εφαρμογής ή ακόμα και του υπολογιστή μετά το πέρας των εργασιών που έχουμε αναθέσει στην εφαρμογή και φυσικά να ρυθμίσουμε τις επιλογές που μας παρέχονται, σύμφωνα με τα πρότυπα του κάθε χρήστης επιθυμεί.



Ακόμη, δίνεται η δυνατότητα να περιορίσουμε την ταχύτητα αποστολής ή λήψης αρχείων καθώς επίσης και τον μέγιστο αριθμό συνδέσεων, έτσι ώστε να μην απορροφάται όλη η ταχύτητα του δικτύου από την εφαρμογή, εάν αυτό δεν είναι επιθυμητό.



Μέσα από το κεντρικό μενού επιλογών του uTorrent, μπορούμε να δούμε και να διαχειριστούμε όλες τις ρυθμίσεις που μπορεί να μας προσφέρει το uTorrent. Από τις πιο απλές ρυθμίσεις που είναι η επιλογή γλώσσας εμφάνισης της εφαρμογής και τον φάκελο αποθήκευσης των αρχείων, μέχρι πιο σύνθετες ρυθμίσεις, όπως η απομακρυσμένη διαχείριση του προγράμματος από οποιονδήποτε browser, προγραμματισμένες αυτοματοποιημένες διαδικασίες με βάση κανόνες που έχει ορίσει ο εκάστοτε χρήστης κ.α.

Ακόμη και για έναν αρχάριο χρήστη, η λήψη ενός αρχείου μέσω του uTorrent, είναι μια πολύ εύκολη και απλή διαδικασία. Το μόνο που πρέπει να κάνει ο χρήστης είναι να βρει τον κατάλληλο σύνδεσμο ή αρχείο .torrent/magnet και αφού το ανοίξει με την εφαρμογή uTorrent, μένει απλά να επιλέξει τα αρχεία που θέλει να λάβει (εάν ο σύνδεσμος περιέχει πάνω από ένα αρχείο) σε ποια τοποθεσία θέλει να το αποθηκεύσει. Αμέσως μετά, αφού η εφαρμογή συνδεθεί με τους διαθέσιμους διαμοιραστές, ξεκινάει την λήψη των αρχείων, προβάλλοντας παράλληλα λεπτομέρειες για την λήψη των αρχείων (ταχύτητα λήψης, εκτιμώμενη ώρα ολοκλήρωσης, πλήθος διαμοιραστών κλπ).

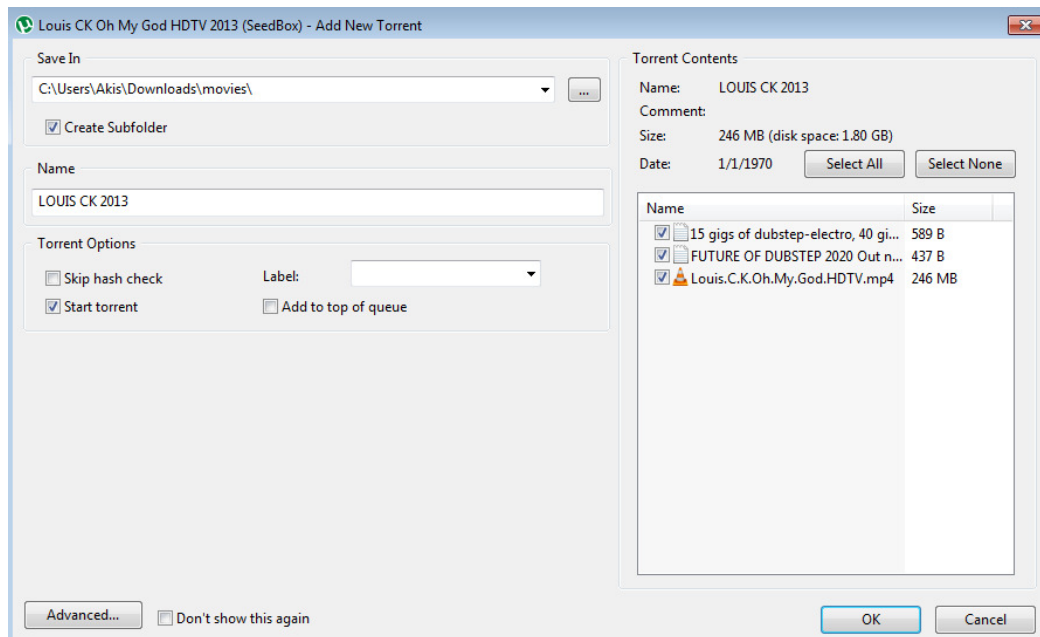
Louis CK Oh My God HDTV 2013 (SeedBox)

Τύπος: Βίντεο > Σειρές Τηλεόρασης	Ανέβηκαν: 2013-04-14 05:54:25
Αρχεία: 3	GMT
Μέγεθος: 246.04 MiB (257996465 Bytes)	By: atndigcrk
Tag(s): louis god 2013 stand	Διαμοιραστές: 396
	Απορροφούν: 6
	Σχόλια: 25

GET THIS TORRENT
(Problems with magnets links are fixed by upgrading your torrent client!)

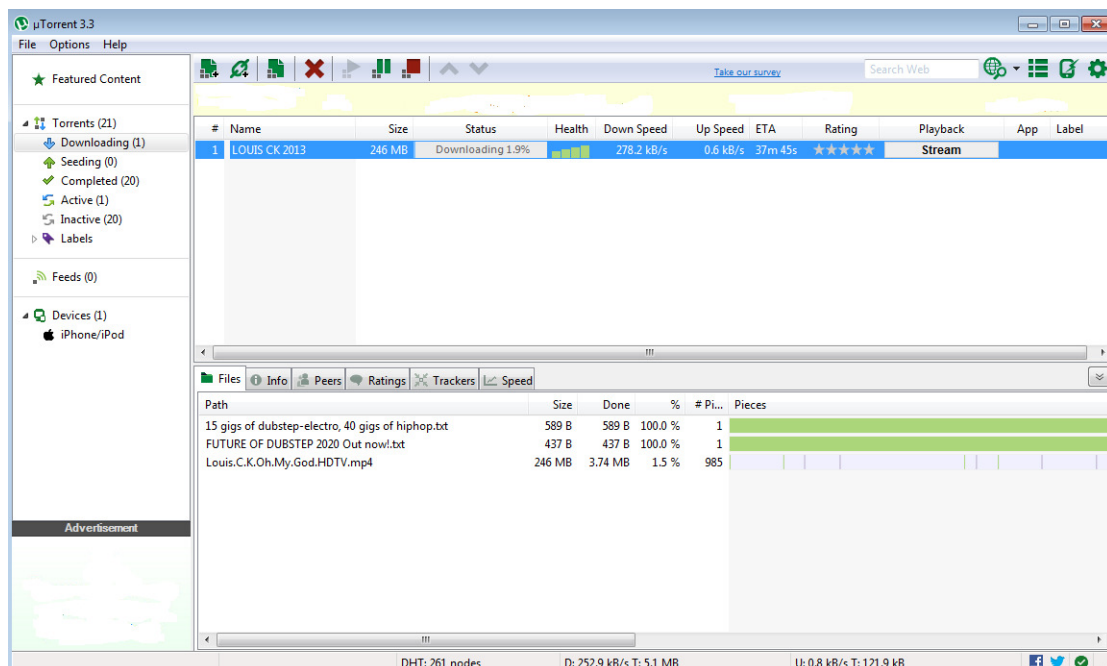
Info Hash:
7F1AF317D92E5FFFEF16ADD836E0AD4A0BF34AE8B

Louis C.K Oh My God HDTV (SeedBox)
fastest downloads on piratebay!
forget ur private site
release info
Louis.C.K.Oh.My.God.HDTV.x264-EVOLVE
MP4 | AAC VBR | 246MB



Το αρχείο θα είναι διαθέσιμο στον χρήστη μόλις ολοκληρωθεί η λήψη του, όμως υπάρχει η δυνατότητα για τα αρχεία ήχου και εικόνας, ο χρήστης να μπορεί να προβάλλει το περιεχόμενο σε μορφή live streaming, προτού ολοκληρωθεί η λήψη τους, πατώντας την επιλογή stream.

Η εφαρμογή μέσω ειδικά διαμορφωμένου αλγορίθμου, κατεβάζει με συγκεκριμένη προτεραιότητα τα τμήματα του αρχείου που είναι απαραίτητα για την ομαλή αναπαραγωγή του, έτσι ώστε ο χρήστης να μπορεί να δει το περιεχόμενο πριν ολοκληρωθεί η λήψη. Αυτό όπως είναι κατανοητό, ισχύει μόνο για αρχεία ήχου ή εικόνας.



Το uTorrent δίνει ακόμη και την δυνατότητα στον χρήστη να δημιουργήσει το δικό του αρχείο .torrent, περιλαμβάνοντας σε αυτό τα αρχεία που θέλει να μοιραστεί με τους

υπόλοιπους χρήστες. Με την επιλογή δημιουργία ενός νέου torrent, ο χρήστης αρκεί να επιλέξει τα αρχεία που θέλει να συμπεριλάβει σε αυτό και να συμπληρώσει τις κατάλληλες διευθύνσεις για τους trackers που θα αναλάβουν τον διαμοιρασμό τους. Ένα αρχείο torrent θα δημιουργηθεί αυτόματα στην προκαθορισμένη από τον χρήστη διεύθυνση αποθήκευσης το οποίο είναι έτοιμο για διαμοιρασμό στο δίκτυο που επιθυμεί.

Create New Torrent

Select Source

C:\Users\Akis\Downloads\series\10-2_legacy_vista32-64_dd_ccc.exe

Add file Add directory

Skip Files:

Torrent Properties

Trackers: udp://tracker.openbittorrent.com:80/announce
udp://tracker.publicbt.com:80/announce

Web Seeds:

Comment:

Piece size: (auto detect)

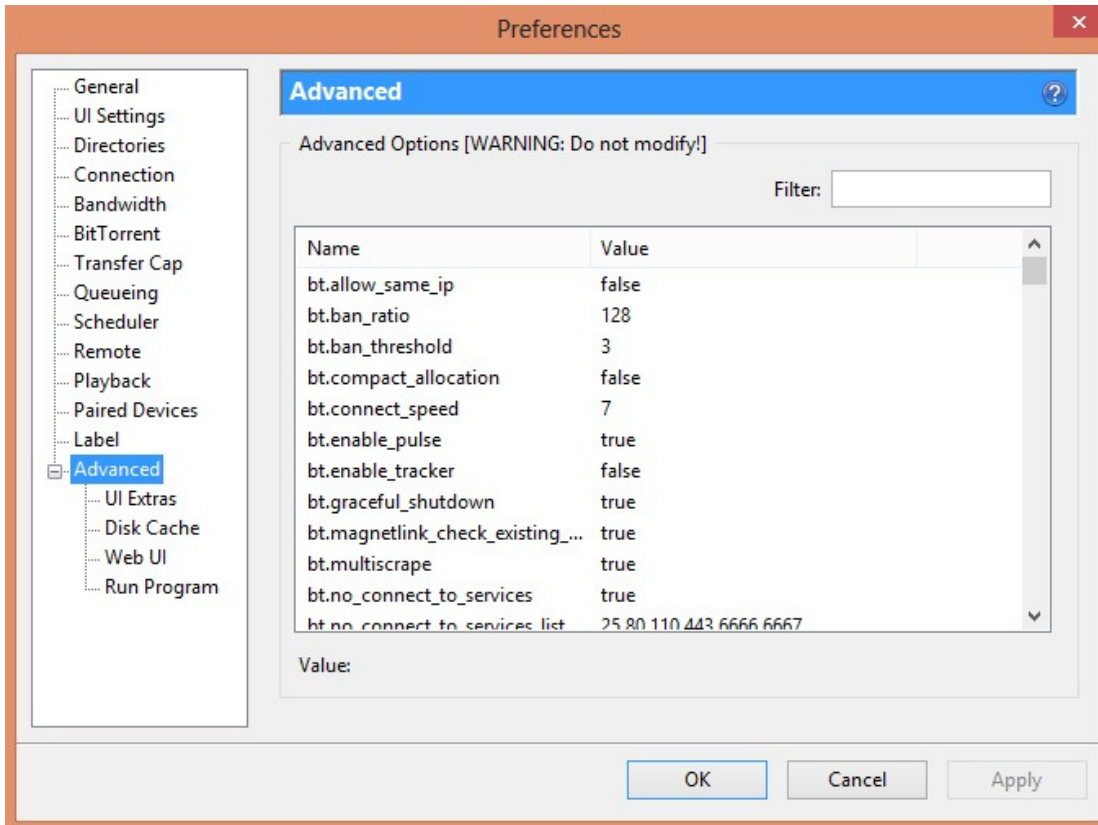
Other

Start seeding Private torrent
 Preserve file order Create Encrypted

Create and save as... Close

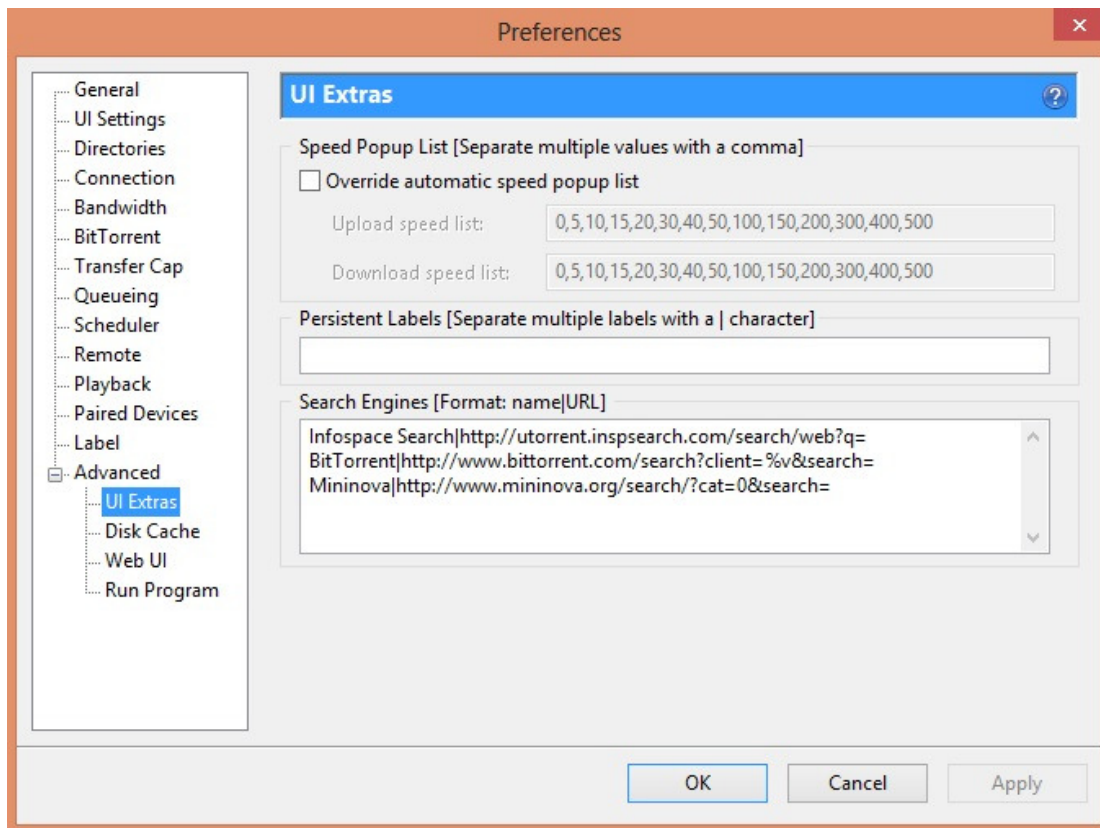
9.1 Προηγμένες Ρυθμίσεις

Παρακάτω θα δούμε κάποιες από τις πολλές προηγμένες ρυθμίσεις που μας παρέχει το uTorrent



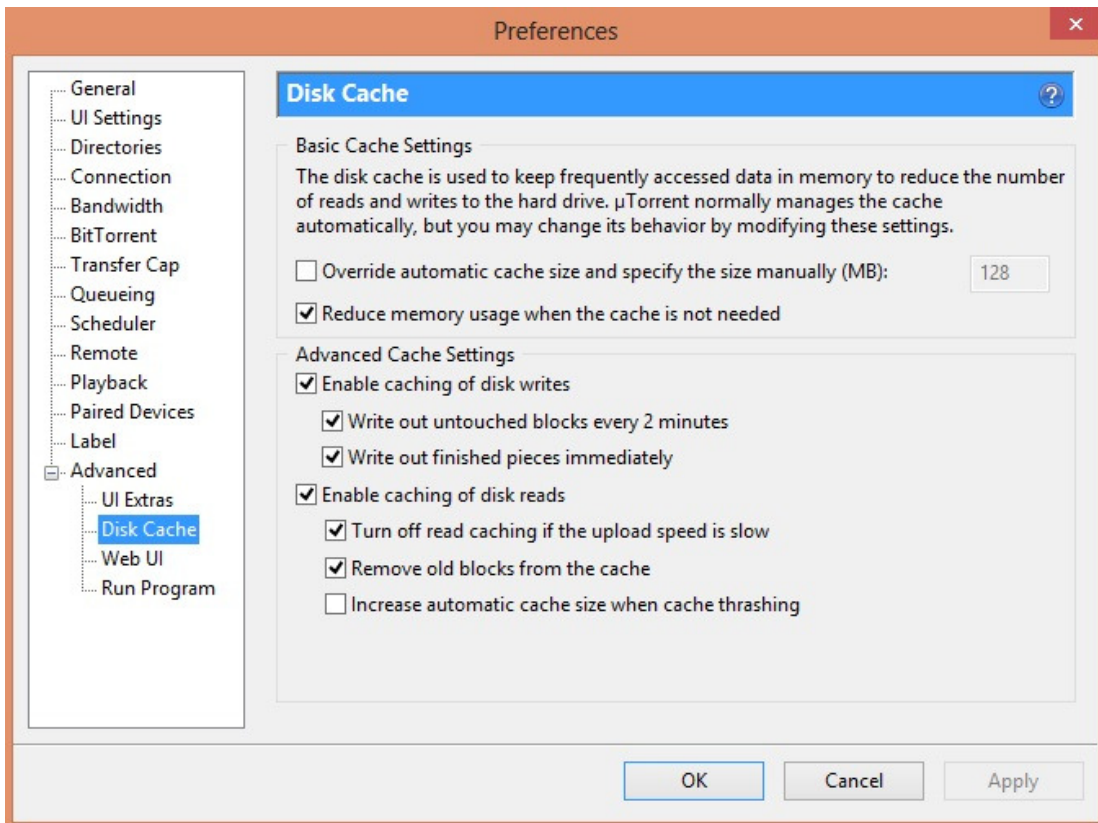
Στην επιλογή Advanced (Προηγμένες) ρυθμίσεις έχουμε ένα πλήθος επιλογών για προχωρημένους χρήστες, όπου μπορούμε να επεμβαίμε σε ρυθμίσεις του κώδικα του προγράμματος μέσα από ένα απλό μενού. Έτσι για παράδειγμα, μπορούμε:

- να ρυθμίσουμε το uTorrent να εκτελείται ταυτόχρονα σε 2 ή περισσότερους υπολογιστές χρησιμοποιώντας την ίδια διεύθυνση IP στο δίκτυο
 - να ορίσουμε το όριο των χρηστών που θα αποκλείει αυτόματα μετά από πολλές αποτυχημένες προσπάθειες
 - να ενεργοποιήσουμε την λειτουργία tracker έτσι ώστε να λειτουργεί σαν ένας αυτόνομος tracker για το P2P δίκτυο
 - να ορίσουμε τον μέγιστο αριθμό συνδέσεων που θέλουμε να έχουμε
- κ.α.



Στην επιλογή UI Extras (Πρόσθετες ρυθμίσεις γραφικού περιβάλλοντος) έχουμε τη δυνατότητα να ορίσουμε τις μονάδες στις οποίες θα αποικονίζεται η γραφική παράσταση της ταχύτητας μεταφοράς, καθώς επίσης και τον ρυθμό με τον οποίο αυτή θα εναλλάσσεται.

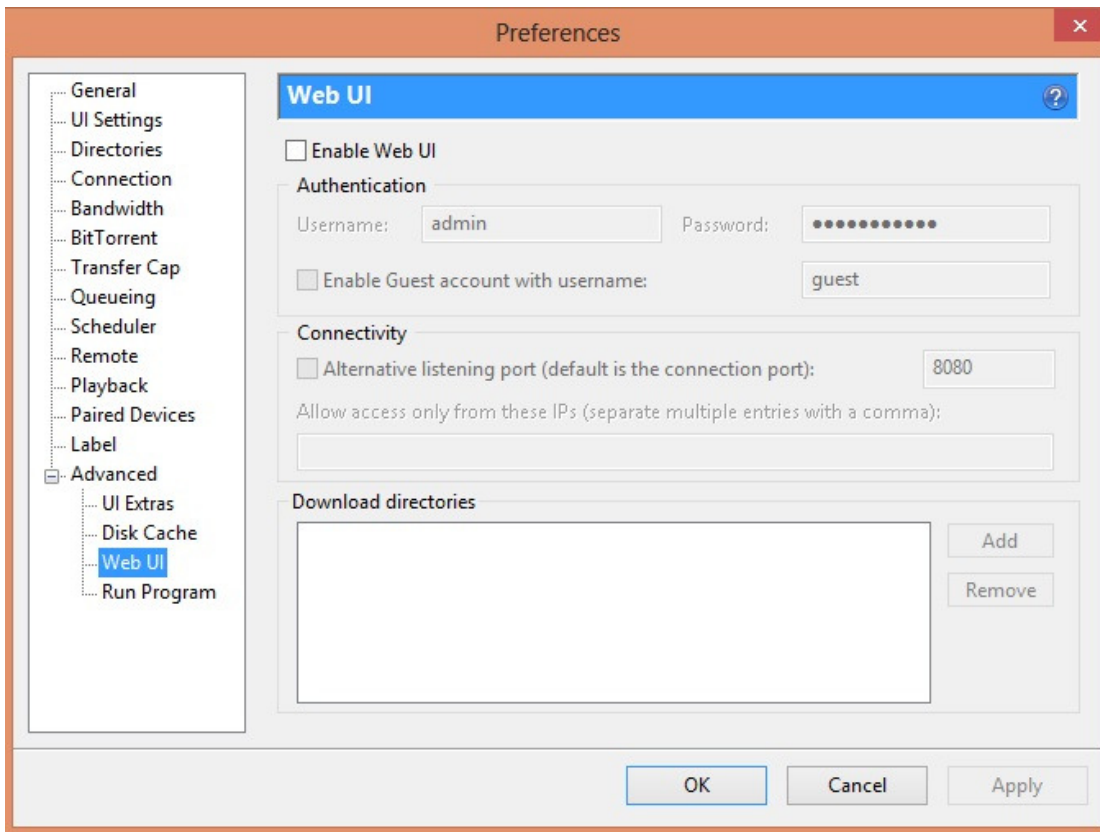
Ακόμη μπορούμε να ορίσουμε ταμπέλες για να την καλύτερη κατηγοριοποίηση των αρχείων που έχουμε στην λίστα, καθώς επίσης και να ορίσουμε τις μηχανές αναζήτησης από τις οποίες θα δεχόμαστε αποτελέσματα.



Στην επιλογή Disk Cache έχουμε τη δυνατότητα να ρυθμίσουμε τον ρυθμό με τον οποίο τα δεδομένα εγγράφονται στον σκληρό δίσκο κατά την αποθήκευση, έτσι ώστε να προσαρμόσουμε το πρόγραμμα στις ανάγκες του κάθε χρήστη, ανάλογα με την εφαρμογή που τον ενδιαφέρει.

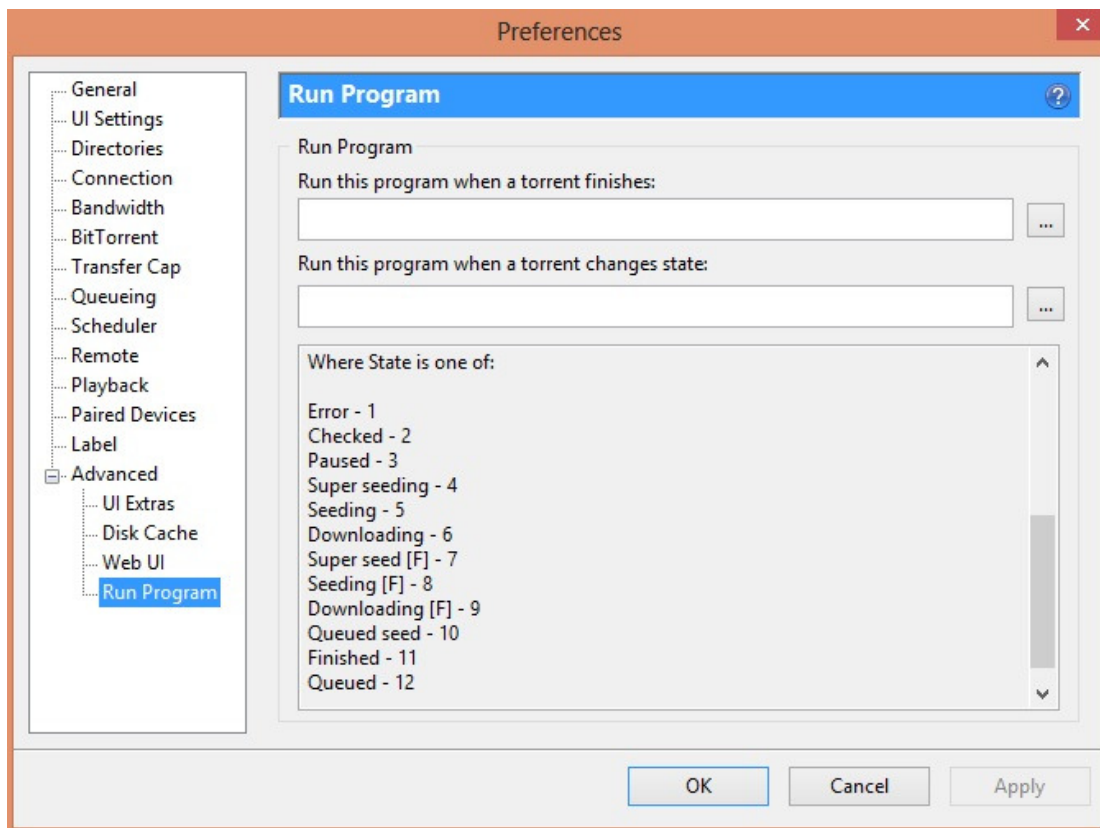
Αναλυτικότερα, μπορούμε:

- να ορίσουμε ένα σταθερό μέγεθος μνήμης cache αντί για το αυτόματο τρόπο που έχει ως προεπιλογή η εφαρμογή
- να ενεργοποιήσουμε την αυτόματη μείωση της μνήμης cache όταν αυτή δεν είναι απαραίτητη
- να ενεργοποιήσουμε ξεχωριστά την μνήμη cache για ανάγνωση ή για εγγραφή καθώς και τον ρυθμό με τον οποίο αυτή θα ανανεώνεται



Μέσα από την επιλογή Web UI (Γραφικό περιβάλλον Web) μπορούμε να ενεργοποιήσουμε την επιλογή που μας δίνει η εφαρμογή για την απομακρυσμένη διαχείριση αυτής από οποιονδήποτε υπολογιστή, μέσα από τον web browser.

Ενεργοποιώντας την επιλογή, ο χρήστης ορίζει το επιθυμητό username και password έτσι ώστε να έχει πρόσβαση στην εφαρμογή, ενεργοποιεί προερατικά την επιλογή guest για περιορισμένη πρόσβαση από άλλους χρήστες, την θύρα (port) στην οποία επιθυμεί να συνδέεται και τέλος ορίζει τους φακέλους που θέλει να είναι προσβάσιμοι. Κάνοντας χρήση της μοναδικής διεύθυνσης ip και της κατάλληλης port, ο χρήστης μπορεί να συνδεθεί από οποιονδήποτε browser.



Τέλος, η επιλογή Run Program (Εκτέλεση Προγράμματος) μας δίνει τη δυνατότητα να επιλέξουμε τις εφαρμογές που θα εκτελέσει το πρόγραμμα όταν τελειώσει τις διαδικασίες που του έχουν ανατεθεί ή όταν αλλάξει μια προεπιλεγμένη κατάσταση.

Μέσα από μια λίστα προεπιλεγμένων εντολών, μπορούμε να διαλέξουμε την κατάσταση στην οποία θα βρίσκεται το πρόγραμμα όταν θέλουμε να εκτελέσει την συγκεκριμένη εφαρμογή και από το πεδίο αναζήτησης δηλώνουμε το πρόγραμμα το οποίο θέλουμε να εκτελεστεί στην εκάστοτε κατάσταση.

9.2 Σύγκριση των δημοφιλέστερων προγραμμάτων

Θα ακολουθήσει μια σύγκριση μεταξύ των δημοφιλέστερων προγραμμάτων ανταλλαγής αρχείων P2P, του uTorrent και του Vuze. Αν και οι βασικές λειτουργίες τους είναι κοινές, δηλ. μπορούν και τα δύο προγράμματα να δημιουργήσουν, να κατεβάσουν και να διαμοιράσουν αρχεία .torrent, υπάρχουν κάποιες μικρές διαφορές που για κάποιους χρήστες μπορεί να είναι αρκετά σημαντικές.

Vuze/Azureus

Αναφορικά με το **Vuze**, το οποίο βασίζεται στο πρώην Azureus, είναι ένα εξελιγμένο πρόγραμμα ανοιχτού κώδικα, το οποίο υποστηρίζει μεγάλο πλήθος τρίτων εφαρμογών, επιτρέποντας έτσι την παραμετροποίησή του από οποιονδήποτε χρήστη έχει γνώσεις προγραμματισμού. Χρησιμοποιείται κυρίως για τον διαμοιρασμό βίντεο υψηλής ποιότητας, υπάρχει και η δυνατότητα όμως της χρήσης του περιβάλλοντος του Azureus παράλληλα.

Είναι γραμμένο σε γλώσσα Java (JRE), χαρακτηριστικό που δίνει το πλεονέκτημα στην εφαρμογή να τρέχει σε οποιοδήποτε λειτουργικό υποστηρίζει γλώσσα java, αλλά παράλληλα και μειονέκτημα, λόγω του ότι η συγκεκριμένη γλώσσα απαιτεί αρκετή υπολογιστική ισχύ.

Αρκετοί χρήστες υποστηρίζουν ότι είναι γρηγορότερο από τον μεγάλο του αντίπαλο, το uTorrent, αλλά οι συγκρίσεις από διάφορα test που έχουν γίνει μέχρι στιγμής, έχουν αμφιλεγόμενο αποτέλεσμα.

uTorrent

Όσον αφορά το **uTorrent**, είναι ένα ελαφρύ από θέμα απαιτούμενης υπολογιστικής ισχύς και παράλληλα αποτελεσματικό πρόγραμμα αρχείων .torrent. Είναι γραμμένο σε κλειστό κώδικα και αυτός είναι ο κύριος λόγος που οι χρήστες του Vuze/Azureus το κατακρίνουν, καθώς οι κατασκευαστές του μπορούν να προσθέσουν οτιδήποτε θελήσουν στο πρόγραμμα, όπως για παράδειγμα προστασία DRM, ή αυτόματες αναφορές για copyrights, χωρίς να έχει προηγηθεί ενημέρωση των χρηστών.

Δεν υποστηρίζει πρόσθετες εφαρμογές από τρίτους, ενώ σαν πλήθος επιλογών έχει λιγότερες από το Vuze, αλλά αρκετά περισσότερες από όσες θα χρειαστεί και θα χρησιμοποιήσει ένας μέσος χρήστης.

Σε αντίθεση με το Vuze, δεν είναι γραμμένο σε γλώσσα Java, γεγονός που του δίνει πλεονέκτημα σχετικά με το μέγεθος, την απόκριση και τις ανάγκες της εφαρμογής για υπολογιστική ισχύ. Αυτό όμως σημαίνει ότι θα πρέπει να αναπτυχθεί μια ξεχωριστή έκδοση της εφαρμογής για κάθε λειτουργικό σύστημα.

Το uTorrent μέχρι στιγμής υποστηρίζει λειτουργικό Windows και OS X, ενώ υπάρχει και έκδοση για Linux, σε μορφή beta, δηλ. βρίσκεται ακόμα στο στάδιο της ανάπτυξης.

Συμπέρασμα

Δεν υπάρχει αμφιβολία ότι το uTorrent είναι γρηγορότερο από το Vuze/Azureus σε επίπεδο προγραμματισμού, από την άλλη όμως τα αμφιλεγόμενα αποτελέσματα των συγκριτικών test που έχουν διεξαχθεί μέχρι σήμερα, δεν μας δίνουν σαφή απάντηση για το ποιά από τις δύο εφαρμογές επιτυγχάνουν μεγαλύτερες ταχύτητες διαμοίρασμού.

10. Βελτιώσεις που χρειάζονται

Παρατηρούμε ότι ενώ ο κεντρικός άξονας της τεχνολογίας p2p είναι σωστά δομημένος και υπάρχει μεγάλο ενδιαφέρον από βιομηχανικής άποψης, υπάρχουν πολλά θέματα που χρειάζονται ακόμα βελτίωση. Ένα από αυτά είναι η επικοινωνία μεταξύ των p2p εφαρμογών, καθώς κάθε μία έχει τις δικές τις βασικές και ειδικές λειτουργίες.

Αξιοσημείωτο είναι επίσης το γεγονός ότι η νέα γενιά προγραμματιστών πάνω σε καινοτόμες εφαρμογές, δεν εκμεταλλεύεται την εμπειρία των προηγούμενων προγραμματιστών p2p, με αποτέλεσμα να ξοδεύουν περιττό χρόνο και ενέργεια αναπτύσσοντας εφαρμογές που έχουν ήδη υλοποιηθεί.

Αντιθέτως, θα ήταν χρήσιμο να ξοδέψουν αυτή την ενέργεια για την υλοποίηση νέων υπηρεσιών και δυνατοτήτων πάνω στις εφαρμογές αυτές. Για να λυθεί αυτό θα πρέπει οι βιομηχανίες να εστιάσουν περισσότερο πάνω στην επικοινωνία μεταξύ των εφαρμογών άλλων εταιριών.

Αυτό μπορεί να γίνει μέσω ενός κοινού συνόλου υπηρεσιών οι οποίες θα προσφέρουν τις δυνατότητες για την ανάπτυξη παρόμοιων εφαρμογών, παράλληλα με τη χρήση υπηρεσιών οι οποίες προσφέρονται από το λειτουργικό σύστημα του peer. Αυτές οι υπηρεσίες αναφέρονται σαν ένα ενδιάμεσο επίπεδο λογισμικού (middleware layer).

Ένα βασικό πλεονέκτημα της χρήσης ενός κοινού middleware είναι το γεγονός ότι οι προγραμματιστές δεν θα είναι απαραίτητο να δημιουργούν τις ίδιες υπηρεσίες επαναλαμβανόμενα. Η επικοινωνία μεταξύ ανομοιογενών συστημάτων (Interoperability) έχει ως σκοπό την επικοινωνία των p2p εφαρμογών μεταξύ διαφορετικού περιβάλλοντος.

Με λίγα λόγια οι peers που τρέχουν σε περιβάλλον Windows, Linux ή οποιοδήποτε άλλο λογισμικό, ακόμα και ασύρματες φορητές συσκευές, μπορούν να χρησιμοποιούν κοινές p2p εφαρμογές. Η επιτυχία του μοντέλου P2P όλα αυτά τα χρόνια, φαίνεται μέσα από τον μεγάλο αριθμό εφαρμογών που έχουν υιοθετήσει το πρωτόκολλο αυτό.

Χρειάζεται όμως αρκετή εξέλιξη ακόμα σε προγραμματιστικό επίπεδο, έτσι ώστε οι προγραμματιστές να αναπτύσσουν τις εφαρμογές με την φιλοσοφία που ακολουθεί το συγκεκριμένο μοντέλο. Ένας τρόπος για να επιτευχθεί ο στόχος αυτός, είναι η συμμετοχή στην ομάδα εργασίας Peer-to-Peer Working Group, μια κοινοπραξία πολλών φορέων της βιομηχανίας ανάπτυξης του P2P Computing.

Κύριως στόχος της ομάδας αυτής είναι η ανάπτυξη και η εξέλιξη υποδομών για την επικοινωνία ανομοιογενών συστημάτων (Interoperability), την διαχείριση της απόδοσης και την ασφάλεια κατά την ανταλλαγή πληροφοριών, έτσι να βοηθήσει στην ανάπτυξη του P2P Computing. Η συμμετοχή στην έρευνα είναι ανοιχτή σε οποιονδήποτε ενδιαφέρεται πάνω στο μοντέλο P2P

11. Βιβλιογραφία

- 1. “JXTA: A Network Programming Environment”,**
L. Gong,
IEEE Internet Computing, vol. 5, no. 3, May/June 2001, pp.
88-95.
- 2. Modeling and Evaluating Peer-to-Peer Storage Architectures,**
Hung-Chang Hsiao, Chung-Ta King,
Department of Computer Science, National Tsing Hua University, Hsinchu, Taiwan
- 3. Modeling and measuring scalable peer-to-peer search networks,**
Brian F. Cooper, Hector Garcia-Molina, Department of Computer Science
Stanford University
- 4. Open Problems in Data-Sharing Peer-to-Peer Systems**
Neil Daswani, Hector Garcia-Molina, Beverly Yang, Department of Computer
Science, Stanford University
- 5. Designing a Distributed Software Development Support System**
Using a Peer-to-Peer Architecture,
Seth Bowen, Frank Maurer, Department of Computer Science, University of
Calgary, Canada
- 6. An Introduction to Peer-to-Peer Computing,**
David Barkai,
Peer-to-Peer Architecture Group, Microcomputer Research Lab Intel Corporation
- 7. Tapestry: An Infrastructure for Fault-tolerant Wide-area Location and Routing,**
Ben Y. Zhao, John Kubiawicz, and Anthony D. Joseph, Computer Science
Division,
University of California, Berkeley
- 8. Routing in Mobile Ad Hoc and Peer-to-Peer Networks.**
A Comparison Rüdiger Schollmeier Ingo Gruber and Michael Finkenzeller
- 9. Report on the dependability properties of P2P architectures** *L.Melville,*
J.Walkerline, I.Sommerville ,Lancaster University
INFORMATION SOCIETIES TECHNOLOGY (IST) PROGRAMME
- 10. Designing a Super-Peer Network,**
Beverly Yang, Hector Garcia-Molina

Computer Science Department, Stanford University

11. Tracing a large-scale Peer to Peer System: an hour in the life of Gnutella ,

Evangelos P. Markatos , Institute of Computer Science (ICS) Foundation for Research & Technology -- Hellas (FORTH)

12. A Measurement Study of Peer-to-Peer File Sharing Systems

Stefan Saroiu, P. Krishna Gummadi, Steven D. Gribble, Dept. of Computer Science and Engineering, Univ. of Washington, Seattle

13. Characteristics of Current P2P File-Sharing Systems power-point presentation

by Stefan Saroiu, P. Krishna Gummadi, Steven Gribble, University of Washington

14. <http://www.openp2p.com/>

Ιστότοπος της εταιρίας O'Reilly που παρουσιάζει διάφορα έργα, κινήσεις και άρθρα πάνω στην τεχνολογία P2P για κάθε τομέα εφαρμογών.

15. <http://pdos.csail.mit.edu/papers/chord:sigcomm01/>

Πληροφορίες για το κατακευματισμένο πρωτόκολλο αναζήτησης δεδομένων των κόμβων που τα περιέχουν σε ένα δίκτυο (Chord).

16. <http://oceanstore.cs.berkeley.edu/info/overview.html>

Ένας ενιαίος αποθηκευτικός χώρος δεδομένων της OceanStore που έχει σχεδιαστεί για την εξυπηρέτηση εκατομμυρίων χρηστών που παρέχει μια ισχυρή και ανθεκτική αποθηκευτική λειτουργία που μπορεί να υλοποιηθεί σε μια υποδομή συνιστάμενη από αναξιόπιστου εξυπηρετητές.

17. <http://www.internetobservatory.net>

Ιστότοπος της εταιρίας iroqme, όπου παρέχονται αρκετές πληροφορίες σχετικά με την στατιστική καταγραφή και ανάλυση των πακέτων που διαμοιράζονται σε ένα δίκτυο, με σκοπό την καλύτερη κατανόηση και έλεγχο της διακίνησης των πληροφοριών μέσα σε αυτό.