

ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ
ΤΜΗΜΑ ΗΛΕΚΤΡΟΝΙΚΩΝ ΜΗΧΑΝΙΚΩΝ &
ΜΗΧΑΝΙΚΩΝ Η/Υ

ΥΠΟΣΤΗΡΙΞΗ ΥΠΗΡΕΣΙΩΝ ΡΟΗΣ ΔΕΔΟΜΕΝΩΝ ΣΕ ΑΝΟΙΧΤΑ ΠΕΡΙΒΑΛΛΟΝΤΑ ΕΤΕΡΟΓΕΝΩΝ ΠΕΛΑΤΩΝ

Νίκος Παππάς

Χανιά 1999

Ευχαριστήρια

Για την ολοκλήρωση της παρούσας μεταπτυχιακής διατριβής ευχαριστώ ιδιαίτερα τον καθηγητή κ. Σταύρο Χριστοδουλάκη για την επίβλεψή του και την ουσιαστική καθοδήγηση κατά την διάρκεια εκπόνησης της εργασίας, αλλά και για την ευκαιρία που μου έδωσε να εργαστώ για μεγάλο χρονικό διάστημα και σαν προπτυχιακός και σαν μεταπτυχιακός φοιτητής στο εργαστήριο Διανεμημένων Πληροφορικών Συστημάτων και Εφαρμογών Πολυμέσων του Πολυτεχνείου Κρήτης.

Ευχαριστώ επίσης ολόκληρη την ομάδα που εργάστηκε για την ολοκλήρωση του ερευνητικού προγράμματος SICMA, Γεωργία Κυριακάκη, Γιάννη Μαραγκουδάκη και Γιάννη Μαυραγάνη για την αποδοτική και μακροχρόνια συνεργασία τους.

Από τα ευχαριστήρια δεν μπορούν να εξαιρεθούν όλοι οι συνάδελφοι του εργαστηρίου για την συμπαράστασή τους κατά την διάρκεια εκπόνησης της διατριβής.

Παππάς Νίκος
Πολυτεχνείο Κρήτης
Νοέμβριος 1999

Περιγραφή

Η μεταπτυχιακή διατριβή περιγράφει τον σχεδιασμό και την ανάπτυξη μιας υπηρεσίας διαχείρισης δεδομένων ροής. Η υπηρεσία αυτή είναι σε θέση να υποστηρίξει τις ανάγκες πολλών πελατών για την αναπαραγωγή αντικειμένων πολυμέσων.

Στα πλαίσια της διατριβής αναπτύχθηκε ένας ολοκληρωμένος μηχανισμός δρομολόγησης πολλαπλών ροών δεδομένων, που συντονίζει κατάλληλα τις εργασίες ανάκτησης, αποταμίευσης και αποστολής των δεδομένων στους πελάτες μέσω δικτύου. Τα αποτελέσματα πειραμάτων που έλαβαν χώρα στα υποσυστήματα μετάδοσης και ανάκτησης, αλλά και σε συστήματα ετερογενών πελατών στο δίκτυο, ελήφθησαν υπόψη στον σχεδιασμό του συστήματος. Ο εξυπηρετητής ροής αξιοποιεί την πολυνηματική τεχνολογία και εφαρμόζει αρχές προγραμματισμού πραγματικού χρόνου.

Οι εργασίες της διατριβής οδήγησαν στην ανάπτυξη ενός εξυπηρετητή δεδομένων ροής μεγάλης κλίμακας, εξοπλισμένο με ένα ολοκληρωμένο μηχανισμό δρομολόγησης εργασιών, ικανό να υποστηρίξει πολλαπλά μέσα αποθήκευσης και να παρέχει υπηρεσίες ροής σε πελάτες αποστέλλοντας τα δεδομένα με τον κατάλληλο ρυθμό και ελεγχόμενο jitter.

Ο εξυπηρετητής, μέσω ενός αυτοπροσαρμοζόμενου μηχανισμού δρομολόγησης, επιτυγχάνει την υποστήριξη των ενεργών πελατών στο δίκτυο. Η πειραματική εφαρμογή συστήματος, παρουσιάζει περιορισμένες απαιτήσεις προσωρινής αποταμίευσης δεδομένων, έτσι δίνει λύση σε ένα από τα βασικά προβλήματα στους εξυπηρετητές πολυμέσων, αξιοποιώντας τις δυνατότητες των σύγχρονων πελατών (επεξεργαστική ισχύς, δικτυακή υποδομή, χώροι αποταμίευσης).

Τα πειράματα που έλαβαν χώρα πάνω από ATM δίκτυο έδειξαν ότι ο εξυπηρετητής ροής που τρέχει σε Ultra One μηχανή είναι σε θέση να εγγυηθεί την εξυπηρέτηση 79 κατάλληλων πελατών που ζητούν την αναπαραγωγή MPEG1 αντικειμένων (1.5Mbps). Από την άλλη πλευρά η εφαρμογή της υπηρεσίας ροής ως εξυπηρετητή ήχου στο ίδιο σύστημα, ανέδειξε πειραματικά την δυνατότητα υποστήριξης 1900 πελατών που, θεωρούμε ότι είναι εξοπλισμένοι με την κατάλληλη δικτυακή υποδομή και ζητούν αντικείμενα ήχου 64Kbps.

Ο σχεδιασμός προϋποθέτει την δυνατότητα επέκτασης της υπηρεσίας ροής, χρησιμοποιώντας στιγμιότυπα εξυπηρετητών σε διανεμημένο επίπεδο, ώστε να είναι δυνατή η υποστήριξη εφαρμογών μεγάλων απαιτήσεων.

Περιεχόμενα

1.1	Γενικά	8
1.2	Αναγκαιότητα	9
1.3	Συνεισφορά διατριβής	10
1.4	Το Πρόγραμμα SICMA	12
1.5	Περιγραφή της διατριβής	15
2.1	Μαγνητικοί δίσκοι	16
2.1.1	Χαρακτηριστικά των σύγχρονων οδηγών δίσκων	16
1.2	Πολιτικές δρομολόγησης δίσκου	18
1.3	Ψηφιακά δεδομένα συνεχόμενης ροής	20
1.3.1	Διαχείριση αποθήκευσης και τοποθέτηση δεδομένων	21
1.3.2	Δρομολογήσεις δίσκου για δεδομένα συνεχόμενης ροής	21
1.1.3	Δρομολόγηση πολλαπλών δίσκων	27
1.4	Επισκόπηση υποσυστήματος υποστήριξης δίσκων	29
1.5	Δρομολόγηση πραγματικού χρόνου	30
1.5.1	Επισκόπηση δρομολόγησης	31
1.1.2	Πολιτικές δρομολόγησης διεργασιών	32
1.1.3	Δρομολόγηση μετάδοσης δεδομένων	35
1.6	Ανάπτυξη εξυπηρετητών δεδομένων ροής	37
1.7	Κλιμάκωση στον εξυπηρετητή αποθήκευσης	39
3.1	Ο Εξυπηρετητής Πολυμέσων Kydonia	48
3.2	Η Αρχιτεκτονική του συστήματος Kydonia	49
3.2.1	Το υποσύστημα υποστήριξης του πελάτη	50
3.2.2	Το υποσύστημα δικτύου και επικοινωνίας	50
3.2.3	Το υποσύστημα διαχείρισης αντικειμένων και συνόδων	50
3.2.4	Το υποσύστημα αποθήκευσης	51
1.1.5	Ο αρχικός σχεδιασμός και ο μηχανισμός δρομολόγησης	53
1.3	Ο εξυπηρετητής πολυμέσων Kydonia-D	56
1.4	Αρχιτεκτονική του μηχανισμού δρομολόγησης & τεχνική επισκόπηση	59
1.5	Αποτελέσματα-Βελτιώσεις	61
1.6	Περίληψη	64
4.1	Ανάγκες της υπηρεσίας διαχείρισης δεδομένων ροής	66
4.2	Αποθήκευση δεδομένων ροής	67
4.2.1	Φυσικό επίπεδο αποθήκευσης	67
4.2.2	Λειτουργίες υποστήριξης των μέσων αποθήκευσης	69
4.2.3	Ο διαχειριστής των μέσων αποθήκευσης	70
1.1.1	Η αποθήκευση δεδομένων πολυμέσων	72
1.1.5	Το σύστημα αποθήκευσης αντικειμένων ροής	73
1.1.6	Διαδικασία αποθήκευσης των αντικειμένων ροής	75

1.3	Διαχείριση προσωρινής μνήμης	77
1.4	Διαχείριση αιτήσεων	79
1.5	Διαχείριση μετάδοσης	80
1.6	Δρομολόγηση εργασιών	82
1.7	Περίληψη	82
5.1	Παράμετροι σχεδιασμού της υπηρεσίας ροής	84
5.2	Η αρχιτεκτονική υλικού	84
1.3	Πειράματα μετάδοσης δεδομένων	85
1.4	Πειράματα λήψης δεδομένων στον πελάτη	88
1.5	Πειράματα απόδοσης στο υποσύστημα ανάκτησης δεδομένων	96
1.6	Περίληψη	101
6.1	Η ροή δεδομένων πολυμέσων και η υπηρεσία διαχείρισης	103
6.2	Η Πολυνηματική Τεχνολογία	104
6.2.1	Γενικά	104
1.1.2	Το μοντέλο νημάτων	106
1.1.3	Υλοποιήσεις νημάτων	107
1.1.4	Διασυνδέσεις νημάτων	108
1.1.5	Εκτέλεση νημάτων	110
1.3	Επισκόπηση αρχιτεκτονικής της υπηρεσίας ροής	111
1.1.1	Η υπηρεσία διαχείρισης δεδομένων συνεχόμενης ροής	114
1.1.2	Το κανάλι επικοινωνίας των εξυπηρετητών δίσκου (DSR-Channel)	120
1.4	Ο μηχανισμός μετάδοσης δεδομένων συνεχόμενης ροής	122
1.4.1	Ο εξυπηρετητής μετάδοσης δεδομένων	122
1.4.2	Πείραμα αποστολής δεδομένων σε τμήματα	124
1.4.3	Πείραμα πολύπλεξης πακέτων πελατών	127
1.1.4	Πείραμα εφαρμογής δρομολόγησης EDF	130
1.1.5	Σχεδιασμός του μηχανισμού δρομολόγησης στον εξυπηρετητή μετάδοσης	132
1.1.6	Πειραματική εφαρμογή του μηχανισμού δρομολόγησης στον εξυπηρετητή μετάδοσης	134
1.5	Συνολική επισκόπηση της υπηρεσίας δεδομένων συνεχόμενης ροής	136
1.6	Απόδοση της υπηρεσίας διαχείρισης δεδομένων ροής	137
1.6.1	Συνολική συμπεριφορά της υπηρεσίας ροής (Πείραμα)	137
1.7	Δρομολόγηση νημάτων με χρήση προτεραιοτήτων	139
1.8	Προγραμματισμός πραγματικού χρόνου	141
1.8.1	Δρομολόγηση πραγματικού χρόνου	141
1.8.2	Κλάσεις δρομολόγησης	144
1.8.3	Αποστολή και εναλλαγή εκτέλεσης (Dispathing & Preemption)	145
1.8.4	Κλείδωμα μνήμης	146
1.9	Αλλαγή της κλάσης δρομολόγησης (Πείραμα)	146
1.10	Σχεδιασμός δρομολόγησης πραγματικού χρόνου	149
1.11	Εφαρμογή του μηχανισμού δρομολόγησης πραγματικού χρόνου της υπηρεσίας ροής	151

1.11.1	Πειραματική εφαρμογή	151
1.1.2	Συνολική ρυθμοαπόδοση του συστήματος (Πείραμα)	154
1.1.3	Αποτελέσματα στην πλευρά των πελατών	155
1.1.4	Συμπεριφορά της υπηρεσίας ροής σε επεξεργαστικά φορτωμένο περιβάλλον (Πείραμα)	156
1.1.5	Πείραμα χρήσης του επεξεργαστή	159
1.12	Δρομολόγηση μεγάλου αριθμού πελατών μικρών απαιτήσεων (Πειράματα)	161
1.13	Περίληψη	167
7.1	Ροές δεδομένων και υλικό υποστήριξης	169
7.2	Ρυθμίσεις συστήματος	170
1.3	Η πολιτική δρομολόγησης για την υποστήριξη ετερογενών πελατών	175
1.4	Αποδοτική διαχείριση μνήμης στον εξυπηρετητή ροής	177
1.4.1	Απαιτήσεις προσωρινής αποταμίευσης δεδομένων	177
1.1.2	Μείωση των απαιτήσεων χώρου σε προσωρινή μνήμη	180
1.5	Ο αυτοπροσαρμοζόμενος μηχανισμός δρομολόγησης	184
1.6	Περίληψη	190
8.1	Κλιμάκωση συστήματος	192
8.2	Κλιμάκωση του συστήματος σε πολυεπεξεργαστικό περιβάλλον διαμοιραζόμενης μνήμης.	193
8.3	Εφαρμογή & Κλιμάκωση του συστήματος σε περιβάλλοντα παράλληλης επεξεργασίας	195
8.4	Κλιμάκωση του συστήματος σε διανεμημένο περιβάλλον	197
1.1.1	Συγχρονισμός Υπολογιστών μέσω δικτύου	199
1.1.2	Πρωτόκολλο SNTP	199
1.1.3	Η επέκταση της υπηρεσίας ροής	200
1.5	Περίληψη	202
9.1	Μελλοντικές επεκτάσεις	206
9.2	Συμπεράσματα – Συνεισφορά	206

➤ Εισαγωγή



Σε αυτό το κεφάλαιο γίνεται εισαγωγή στην περιοχή των εξυπηρετητών αποθήκευσης, παρουσιάζονται η αναγκαιότητα και οι στόχοι της διατριβής, ενώ επίσης περιγράφεται η δομή του κειμένου που ακολουθεί. Γίνεται επίσης μια περιληπτική αναφορά στο ευρωπαϊκό ερευνητικό πρόγραμμα SICMA στα πλαίσια του οποίου σχεδιάστηκε και αναπτύχθηκε το σύστημα της διατριβής.

1.1 Γενικά

Τα σύγχρονα υπολογιστικά συστήματα έχουν ανάγκη υποστήριξης νέων εφαρμογών που συνδυάζουν την ύπαρξη των παραδοσιακών τύπων δεδομένων, αλλά και επιβάλλουν την χρήση πολυμέσων. Η εντυπωσιακή ανάπτυξη των δικτυακών εφαρμογών και η εξάπλωση του διαδικτύου δίνει την δυνατότητα σε πολλούς χρήστες να έχουν πρόσβαση σε ένα μεγάλο όγκο πληροφορίας. Συνεπώς ο σχεδιασμός και η ανάπτυξη συστημάτων εξυπηρετητών αποθήκευσης δεδομένων πολυμέσων, που θα δέχονται αιτήσεις και θα μεταδίδουν δεδομένα μέσω δικτύου, εμφανίζεται ως επιτακτική ανάγκη. Τα χαρακτηριστικά όμως των δεδομένων πολυμέσων, όπως video και ήχος, επιβάλλουν τον προσεκτικό σχεδιασμό και διαχείριση ανάλογων συστημάτων. Κι' αυτό γιατί οι απαιτήσεις πραγματικού χρόνου που υπεισέρχονται στην ανάκτηση και το μεγάλο μέγεθος των δεδομένων πολυμέσων έχουν σαν συνέπεια την δέσμευση σημαντικών πόρων ενός συστήματος στους οποίους περιλαμβάνονται το δίκτυο, η μνήμη, τα μέσα αποθήκευσης και οι επεξεργαστές. Η ανάπτυξη αποδοτικών τεχνικών συμπίεσης έρχεται να βοηθήσει την αντιμετώπιση του προβλήματος διαχείρισης των πολυμέσων με την μείωση των απαιτήσεων αποθηκευτικού χώρου. Στους παρακάτω πίνακες παρουσιάζονται ενδεικτικά μερικά χαρακτηριστικά τύπων πολυμέσων.

Προδιαγραφές	Απαιτήσεις ρυθμού και χώρου
Ήχος ομιλίας	64 kbits/sec
Ήχος ποιότητας CD (2 κανάλια, 16 bit – 44.1kHz)	1.4 Mbits/sec

Video NTSC (640x480 pixels, 8bit/pixel)	8.7 Mbytes/sec
Video HDTV (1024x2000 pixels, 24bit/pixel)	351 Mbytes/sec

Πίνακας 1: Χαρακτηριστικά ψηφιακών δεδομένων πολυμέσων

Ένας εξυπηρετητής πολυμέσων αποθηκεύει συμπιεσμένη πληροφορία σε μέσα αποθήκευσης υψηλής χωρητικότητας, όπως είναι οι μαγνητικοί σκληροί δίσκοι, που παρέχουν δυνατότητα τυχαίας προσπέλασης και σύντομους χρόνους αναζήτησης. Η ιδιαιτερότητα των δεδομένων πολυμέσων διαφοροποιεί τον σχεδιασμό και την ανάπτυξη αυτών των συστημάτων από τις παραδοσιακές προσεγγίσεις. Οι εξυπηρετητές βάσεων δεδομένων, για παράδειγμα, σχεδιάζονται ώστε να βελτιστοποιούν τον αριθμό των αιτήσεων στην μονάδα του χρόνου και να παρέχουν γρήγορους χρόνους απόκρισης στους πελάτες. Σε αντίθεση οι εξυπηρετητές πολυμέσων πρέπει να ικανοποιούν τις απαιτήσεις που προκύπτουν από την «συνεχόμενη» φύση των πολυμέσων και πρέπει να εγγυώνται την παράδοση των δεδομένων με χρονική ακρίβεια, όπως αυτή προσδιορίζεται από τους ρυθμούς αναπαραγωγής των μέσων. Οι βασικοί περιορισμοί σε ένα εξυπηρετητή πολυμέσων είναι το εύρος μετάδοσης των δίσκων, η συνολική χωρητικότητα και το μέγεθος της πρωτεύουσας μνήμης.

Προδιαγραφές	Απαιτήσεις ρυθμού και χώρου
MPEG 1	1.5 Mbits/sec
MPEG 2	2 Mbits/sec
MPEG 2	80 Mbits/sec

Πίνακας 2: Χαρακτηριστικά συμπιεσμένων ψηφιακών πολυμέσων

Στο επόμενο κεφάλαιο γίνεται μια πιο αναλυτική παρουσίαση των ζητημάτων σχετικών με τον σχεδιασμό και την ανάπτυξη εξυπηρετητών και της σχετικής έρευνας στην περιοχή.

1.2 Αναγκαιότητα

Η σύγχρονη έκρηξη της πληροφορικής και της τεχνολογίας των υπολογιστών, έχει δημιουργήσει τις προϋποθέσεις για την εμφάνιση νέων τύπων εφαρμογών που τείνουν να ενοποιήσουν όλους τους τομείς της ηλεκτρονικής τεχνολογίας και όλους τους τύπους πληροφορίας, έχοντας σαν κοινή βάση την χρήση των ψηφιακών υπολογιστών. Έτσι εμφανίζεται η ανάγκη για αποτελεσματική διαχείριση της πληροφορίας πολυμέσων όπως είναι video και ήχος, για την κάλυψη εφαρμογών όπως αλληλεπιδραστική τηλεόραση, ψηφιακές βιβλιοθήκες, εκπαίδευση από απόσταση και πολλές άλλες. Ο μεγάλος όγκος

πληροφορίας, που παρατηρείται σε εφαρμογές πολυμέσων, καθιστά αδύνατη την αποθήκευση της πληροφορίας σε τοπικό επίπεδο με αντίγραφα. Έτσι εκμεταλλευόμαστε την σύγχρονη δικτυακή τεχνολογία και χρησιμοποιώντας το σχήμα πελάτη-εξυπηρετητή επιτρέπουμε σε μεγάλο αριθμό πελατών να έχουν πρόσβαση σε τεράστια ποσά πληροφορίας μέσω δικτύου.

Εύλογη είναι συνεπώς η ανάγκη για την ανάπτυξη τέτοιων εξυπηρετητών που θα αναλαμβάνουν την διαχείριση, την αποθήκευση και την προώθηση των δεδομένων στους πελάτες. Όμως με δεδομένο τις ιδιαιτερότητες των πολυμέσων και την προϋπόθεση εξυπηρέτησης μεγάλου αριθμού πελατών, η ανάπτυξη ενός εξυπηρετητή αποτελεί πραγματικά πρόκληση για την υπάρχουσα τεχνολογία. Κι' αυτό γιατί ένα τέτοιο σύστημα λογισμικού καλείται να διαχειριστεί αποδοτικά πολλούς πόρους, όπως είναι τα μέσα αποθήκευσης, η μνήμη, οι πόροι δικτύου και οι επεξεργαστές, ώστε να αντεπεξέλθει στις υψηλές απαιτήσεις

Η παρούσα διατριβή προσπαθεί να συνεισφέρει στην έρευνα που γίνεται στον τομέα της δρομολόγησης της πρόσβασης στους πόρους και στην εκτέλεση των εργασιών με χρονική ακρίβεια. Η ανάγκη για την υποστήριξη πολλών πελατών οδηγεί στην έρευνα για τους τρόπους καλύτερης αξιοποίησης των διαθέσιμων πόρων. Ειδικά συστήματα λογισμικού πρέπει να αναπτυχθούν με στόχο την σωστή δρομολόγηση των διεργασιών και της πρόσβασης στους πόρους ώστε να επιτευχθούν:

- ικανοποίηση των περιορισμών πραγματικού χρόνου στην παρουσίαση των μέσων
- η μεγιστοποίηση του αριθμού των πελατών που ζητούν την αναπαραγωγή πολυμέσων και εξυπηρετούνται ταυτόχρονα
- η ελαχιστοποίηση της μέσης καθυστέρησης απόκρισης των αιτήσεων.

Επιπλέον υπάρχει ανάγκη για μια ενιαία αντιμετώπιση της δρομολόγησης όλων των πόρων, που θα καταλήξει στον σχεδιασμό ενός ολοκληρωμένου συστήματος χωρίς να αγνοούνται κάποιοι τομείς λειτουργίας. Καθώς όμως οι απαιτήσεις ολοένα και αυξάνονται χρειάζεται και ένας μηχανισμός κλιμάκωσης του συστήματος ενός εξυπηρετητή, ο οποίος θα επιτρέπει να αντεπεξέλθει ικανοποιητικά το σύστημα σε αύξηση του φόρτου (π.χ. ανάγκη για κάλυψη περισσότερων πελατών) με πρόσθεση επιπλέον πόρων.

1.3 Συνεισφορά διατριβής

Η μεταπτυχιακή διατριβή ασχολείται με ζητήματα ανάπτυξης ενός εξυπηρετητή δεδομένων ροής και τους μηχανισμούς δρομολόγησης που πρέπει να εφαρμόσει, ώστε να έχουμε τα καλύτερα δυνατά αποτελέσματα απόδοσης άλλα και αξιοποίησης των πόρων του συστήματος. Προσθετικά η αρχιτεκτονική του συστήματος προβλέπει την κλιμάκωση του εξυπηρετητή, καθώς το σύστημα λογισμικού έχει την ικανότητα επέκτασης σε περιπτώσεις αύξησης των πόρων για την κάλυψη μεγάλου αριθμού πελατών.

Η παρούσα διατριβή παρουσιάζει μια ολοκληρωμένη αρχιτεκτονική ενός υποσυστήματος εξυπηρέτησης δεδομένων συνεχόμενης ροής και την υλοποίησή του με τις κατάλληλες τεχνικές ανάπτυξης λογισμικού (δρομολόγησης διεργασιών και νημάτων, συγχρονισμός εργασιών, υπηρεσίες πραγματικού χρόνου, κλείδωμα μνήμης, ανάλυση script συμβολοσειρών, ενδοεπικοινωνία διεργασιών και διαχείριση μνήμης).

Ο προτεινόμενος σχεδιασμός του μηχανισμού δρομολόγησης των αιτήσεων και της ροής δεδομένων στο σύστημα προϋποθέτει πολυνηματική ανάπτυξη και εφαρμογή προγραμματισμού πραγματικού χρόνου. Η εφαρμογή αυτού του μηχανισμού δρομολόγησης έχει να επιδείξει καλή αξιοποίηση της χρήσης του επεξεργαστή (CPU utilization) και επίτευξη καλής ακρίβειας στην περιοδική ανάκτηση δεδομένων διατηρώντας σταθερούς κύκλους εξυπηρέτησης.

Όπως αναφέρεται και στην σχετική έρευνα η πλειονότητα των δρομολογήσεων ανάκτησης δεδομένων ακολουθεί εξυπηρέτηση σε κύκλους, η οποία γίνεται είτε με ανάθεση μιας διεργασίας ανά αντικείμενο ροής (η οποία περιοδικά στέλνει αιτήσεις στο δίσκο και στα μεσοδιαστήματα παραμένει αδρανής), είτε με μια διεργασία ανά δίσκο (και κάποια διεργασία δρομολογητής στέλνει περιοδικά αιτήσεις στο δίσκο). Αυτές όμως οι απλές αντιμετωπίσεις δεν έχουν θετικά αποτελέσματα, εφόσον οι παραδοσιακά δρομολογούμενες διεργασίες έχουν ως άμεση απόρροια την εμφάνιση αποκλίσεων από την εκτέλεση των εργασιών με χρονικά σωστό τρόπο, ιδιαίτερα σε περιπτώσεις αντιμετώπισης υψηλού φόρτου εργασίας. Επίσης δεν υπάρχουν σοβαρές προτάσεις πολυνηματικού σχεδιασμού. Στην διατριβή προτείνεται ένας πολυνηματικός σχεδιασμός ανάπτυξης του δρομολογητή που προσφέρει αξιοπιστία.

Επιπρόσθετα στο σύστημα εφαρμόζεται δρομολόγηση μετάδοσης με ικανότητα περιορισμού των προβλημάτων διακυμάνσεων του ρυθμού μετάδοσης, καθυστέρησης αποστολής πακέτων και της νευρικότητας της μετάδοσης (transmission rate, delay, jitter). Επίσης παρουσιάζεται ένας ολοκληρωμένος μηχανισμός ελέγχου και διαχείρισης των πολλαπλών ροών δεδομένων από τα μέσα αποθήκευσης, στους αποταμιευτές προσωρινής μνήμης και το τοπικό δίκτυο και όλα αυτά μέσα από ένα αυτοπροσαρμοζόμενο σύστημα δρομολόγησης, που σχεδιάστηκε ειδικά για την υποστήριξη υπηρεσιών ροής σε ανοιχτά περιβάλλοντα ετερογενών πελατών. Η χρήση του μηχανισμού αυτού έχει να επιδείξει δραστική μείωση των απαιτήσεων αποταμίευσης δεδομένων στον εξυπηρετητή, προσπαθώντας να εκμεταλλευτεί τις δυνατότητες των πελατών για αποθήκευση και λήψη δεδομένων.

Ο σχεδιασμός και ανάπτυξη του συστήματος έγινε με δυνατότητα κλιμακωτής επέκτασης εφαρμόζοντας ένα μηχανισμό πολλών συγχρονισμένων τοπικών δρομολογητών για την εξυπηρέτηση των αιτήσεων αναπαραγωγής, επιτυγχάνοντας μείωση των απαιτήσεων σε εύρος για την ενδοεπικοινωνία των κόμβων του συστήματος και αποφυγή μετάδοσης δεδομένων περιεχομένου μεταξύ των κόμβων. Η κλιμάκωση συστημάτων εξυπηρετητών με χρήση συνηθισμένου και οικονομικού υλικού γίνεται με την επέκταση πολλών υπολογιστών-εξυπηρετητών σε δίκτυο. Η τοποθέτηση των δεδομένων στους εξυπηρετητές γίνεται με δύο τρόπους. Ο πρώτος είναι η αποθήκευση αντιγράφων αντικειμένων σε διαφορετικούς

κόμβους (replication), ενώ ο δεύτερος προτείνει την διασπορά των δεδομένων σε πολλούς κόμβους (striping). Πολλές αναφορές στην βιβλιογραφία που υιοθετούν την δεύτερη λύση δεν αναφέρουν σαφώς πως γίνεται ο συντονισμός των κόμβων του διανεμημένου εξυπηρετητή για την ικανοποίηση των αιτήσεων αναπαραγωγής των πελατών. Άλλες αναφορές περιγράφουν τον συντονισμό των εργασιών ως υπευθυνότητα των χρηστών. Άλλες προτείνουν ένα σχήμα σύμφωνα με το οποίο ένας κόμβος αναλαμβάνει την ευθύνη διεκπεραίωσης μιας αίτησης, δέχεται δεδομένα από τους εμπλεκόμενους κόμβους και τα μεταδίδει με την σωστή σειρά στον πελάτη. Ενώ τέλος άλλες αναφορές εμφανίζουν ως λύση συντονισμού την συνεχή μετάδοση μηνυμάτων (που περιγράφουν την κατάσταση εξυπηρέτησης των αιτήσεων) μεταξύ των κόμβων. Οι προσεγγίσεις αυτές εμφανίζουν μειονεκτήματα λόγω της πολυπλοκότητας του λογισμικού στην πλευρά του χρήστη ή λόγω της ανάγκης για υψηλό εύρος μετάδοσης μεταξύ των κόμβων. Στην διατριβή περιγράφεται ένα σχήμα το οποίο μετά από ένα αρχικό συντονισμό των τοπικών δρομολογητών και βασισμένο στην δρομολόγηση πραγματικού χρόνου η οποία έχει επιτευχθεί σε εξυπηρετητές stand-alone, επιτρέπει την ανεξάρτητη λειτουργία των κόμβων για την εξυπηρέτηση αιτήσεων με μειωμένο ενδο-επικοινωνιακό κόστος. Η βαθύτερη μελέτη και ο πειραματισμός πάνω στο προτεινόμενο σχήμα επέκτασης αποτελεί στόχο μελλοντικών εργασιών.

1.4 Το Πρόγραμμα SICMA

Τα τελευταία χρόνια υπάρχει παγκόσμιο ενδιαφέρον για την κατασκευή μιας υποδομής εξυπηρετητή που θα επιτρέπει την κατανομή της πληροφορίας σε ένα σύνολο από εξυπηρετητές που θα την παρέχουν κατ' απαίτηση σε χρήστες μέσω δικτύων υψηλής χωρητικότητας. Οι τελικοί χρήστες θα είναι σπίτια εφοδιασμένα με συσκευές ανάκτησης ή Set Top Units (STUs) και η πληροφορία στην οποία θα έχουν πρόσβαση θα είναι πληροφορία πολυμέσων (κείμενο, εικόνες, video και ήχος).

Οι εξυπηρετητές πολυμέσων αποτελούν βασικό στοιχείο αυτού του είδους υποδομής εφόσον προσφέρουν αποθήκευση τεράστιου ποσού δεδομένων πολυμέσων σε ένα ευρύ πεδίο εφαρμογών όπως: video κατ' απαίτηση, τηλεαγορές, νέα κατ' απαίτηση. Ο τεχνικός στόχος του προγράμματος SICMA ήταν η ανάπτυξη ενός μεγάλης κλίμακας πληροφοριακού εξυπηρετητή πολυμέσων που θα έχει τη δυνατότητα να αποθηκεύει, να διαχειρίζεται και να μεταδίδει δεδομένα πολυμέσων σε ένα μεγάλο αριθμό πελατών.

Ο εξυπηρετητής του SICMA είναι συμβατός με διεθνή πρότυπα όπως το DAVIC (Digital Audio Visual Council), έτσι ώστε να επιτυγχάνει τη μέγιστη δυνατή διαλειτουργικότητα με άλλα στοιχεία της υποδομής, καθώς και με συστήματα μετάδοσης πληροφορίας και για την υποστήριξη ενός μεγάλου φάσματος εφαρμογών. Το DAVIC είναι ένας μη κερδοσκοπικός οργανισμός που περιλαμβάνει μεγάλους οργανισμούς στα πεδία των τηλεπικοινωνιών, της καλωδιακής τηλεόρασης, στους εξυπηρετητές video, και τους κατασκευαστές συσκευών. Το πρότυπο αυτό περιγράφει τη λειτουργικότητα και τα πρωτόκολλα που πρέπει να υποστηρίζονται από κάθε τμήμα της υποδομής ώστε να

επιτυγχάνεται διαλειτουργικότητα. Ο εξυπηρετητής πρέπει να είναι μεγάλης κλίμακας, ικανός να ανταποκρίνεται σε μεγάλο αριθμό αιτήσεων χωρίς να μειώνεται η απόδοσή του και να διαχειρίζεται το περιεχόμενο πολυμέσων με αποδοτικό τρόπο, ώστε να υποστηρίζει τις απαιτήσεις πραγματικού χρόνου των δεδομένων πολυμέσων, όπως την ευαισθησία σε καθυστερήσεις του video.

Βασισμένος στο πρότυπο DAVIC, ο εξυπηρετητής του SICMA έχει αντικειμενοστραφή αρχιτεκτονική και αποτελείται από ένα σύνολο από υπηρεσίες που υποστηρίζουν τις εφαρμογές πολυμέσων. Οι κύριες υπηρεσίες που παρέχονται από το σύστημα είναι :

1. η υπηρεσία εγκατάστασης περιεχομένου που χρησιμοποιείται από τους παροχείς/διοκτήτες της πληροφορίας για να εγκαταστήσουν νέες εφαρμογές στον εξυπηρετητή και να ενημερώσουν τις ήδη υπάρχουσες
2. η υπηρεσία επιλογής υπηρεσιών που επιτρέπει σε εξωτερικούς πελάτες να εντοπίσουν και να επιλέξουν κάποια από τις διαθέσιμες υπηρεσίες
3. η υπηρεσία διαχείρισης των ροών πληροφορίας που μεταδίδει και διαχειρίζεται τις συνεχείς ροές πληροφορίας από τον εξυπηρετητή στον τελικό χρήστη/συσκευή
4. η υπηρεσία εγκατάστασης νέων υπηρεσιών/εφαρμογών επί πληρωμή

Ο εξυπηρετητής του SICMA μπορεί επίσης να μεταφέρει δεδομένα video και ήχου κωδικοποιημένα σε MPEG-2 μορφή και να μεταδώσει ροές πληροφορίας πάνω από ATM (Asynchronous Transfer Mode) δίκτυα υψηλών ταχυτήτων. Άλλα χαρακτηριστικά/υπηρεσίες που παρέχονται είναι η μεταφορά ηλεκτρονικών αρχείων και εκτελέσιμων προγραμμάτων από τον εξυπηρετητή στη συσκευή τελικό αποδέκτη.

Το πρότυπο DAVIC έχει να κάνει κυρίως με την ικανότητα των συστημάτων να αλληλεπιδρούν το ένα με το άλλο και να ανταλλάσσουν δεδομένα με μια προσυμφωνημένη μέθοδο χωρίς να καθορίσει πως θα είναι η εσωτερική διαχείριση των δεδομένων πολυμέσων στον εξυπηρετητή.

Ο εξυπηρετητής του προγράμματος SICMA χρησιμοποιεί το σύστημα KYDONIA - μια αντικειμενοστραφή βάση δεδομένων που έχει αναπτυχθεί από το Εργαστήριο Καταμεμημένων Πληροφορικών Συστημάτων του Πολυτεχνείου Κρήτης για τη διαχείριση των δεδομένων πολυμέσων. Το σύστημα αυτό έχει επεκταθεί στα πλαίσια του SICMA, έτσι ώστε να υποστηρίξει πλήρη λειτουργικότητα πολυμέσων. Ο ρόλος του συστήματος KYDONIA αφορά

- στην διαχείριση της αποθήκευσης αντικειμένων πολυμέσων
 - στην παροχή συνεχών ροών πραγματικού χρόνου μέσω μηχανισμών κατάλληλης δρομολόγησης των αιτήσεων των πελατών
 - στην υποστήριξη λειτουργικότητας βάσεων δεδομένων πολυμέσων, όπως δημιουργία αντικειμένων πολυμέσων και αντικειμένων που αναπαριστούν το περιεχόμενο των πολυμέσων

- επαρκή διαχείριση MPEG-2 video και HTML εγγράφων
- υποστήριξη μεθόδων προσπέλασης για δεδομένα βίντεο και κειμένου
- υποστήριξη ανάκτησης πολυμέσων βάσει του περιεχομένου τους

Ο εξυπηρετητής του SICMA υποστηρίζει ένα μεγάλο αριθμό πελατών χρησιμοποιώντας μια αρχιτεκτονική από επεξεργαστικά τμήματα, συσκευές αποθήκευσης και εξωτερικά τμήματα διεπικοινωνίας.

Το πρόγραμμα SICMA τελείωσε τον Νοέμβριο 1998 με ιδιαίτερη επιτυχία, καθώς οδήγησε στην ανάπτυξη του εξυπηρετητή πολυμέσων SICMA, που συμβαδίζει με το πρότυπο DAVIC. Ο εξυπηρετητής αυτός εγκαταστάθηκε και χρησιμοποιήθηκε σε δύο δημόσιες πειραματικές εφαρμογές (trials) μέσα στα πλαίσια του προγράμματος. Η πρώτη πειραματική εφαρμογή έλαβε χώρα στο μουσείο φυσικής ιστορίας του Λονδίνου την περίοδο Ιούνιος-Σεπτέμβριος 1997. Ο εξυπηρετητής SICMA εγκαταστάθηκε σε μια παράλληλη μηχανή και μέσω του τοπικού δικτύου ATM, υποστήριζε -με την παροχή πληροφορίας πολυμέσων- μια εφαρμογή εικονικής πραγματικότητας που αναπαριστούσε τα ταξίδια του γνωστού πλοίαρχου Cook. Ο εξυπηρετητής παρείχε υπηρεσίες σε μερικούς πελάτες ταυτόχρονα, που κυρίως είχαν να κάνουν με την παρουσίαση video υψηλής ποιότητας (MPEG2-Transport Stream). Την ίδια περίοδο λάμβανε χώρα η ίδια πειραματική εφαρμογή σε μικρότερη κλίμακα σε ένα άλλο μουσείο στην πόλη Tongeren του Βελγίου. Αξιοσημείωτο είναι το γεγονός ότι ορισμένες χρονικές περιόδους ένας πελάτης του Λονδίνου δεχόταν πληροφορία από το σύστημα το Βελγίου, μέσω ενός μηχανισμού ενδοεπικοινωνίας μεταξύ των δυο εξυπηρετητών (server to server communication), ο οποίος ήταν διαφανής ως προς τους χρήστες. Η επικοινωνία των δύο εξυπηρετητών γινόταν πάνω από πανευρωπαϊκό δίκτυο υψηλών ταχυτήτων ATM, στο οποίο δεσμεύτηκε ένα συγκεκριμένο εύρος μετάδοσης με την συνεργασία ευρωπαϊκών τηλεπικοινωνιακών οργανισμών ειδικά για της ανάγκες αυτής της πειραματικής εφαρμογής.

Η δεύτερη πειραματική εφαρμογή διεξήχθη στην πόλη Hasselt του Βελγίου και έλαβε χώρα την περίοδο Σεπτέμβριος-Οκτώβριος 1998. Σε αυτή την περίπτωση ο εξυπηρετητής SICMA εγκαταστάθηκε πάλι σε μια παράλληλη μηχανή ευρείας κλίμακας, με στόχο να υποστηρίξει δεκάδες χρήστες για τις ανάγκες μιας εφαρμογής τηλε-εκπαίδευσης. Οι χρήστες αποτελούνταν από φοιτητές του Πανεπιστημίου Limburg του Βελγίου, οι οποίοι χρειάζονταν πρόσβαση σε υλικό πολυμέσων για την εκπλήρωση των απαιτήσεων ορισμένων μαθημάτων. Η πρόσβαση ήταν εφικτή με την χρήση ειδικού λογισμικού το οποίο αναπτύχθηκε να αποτελέσει τον πελάτη και εγκαταστάθηκε σε προσωπικούς υπολογιστές. Οι φοιτητές μπορούσαν να χρησιμοποιήσουν υπολογιστές του Πανεπιστημίου διασυνδεδεμένους σε δίκτυο με τον εξυπηρετητή, αλλά και είχαν την δυνατότητα πρόσβασης από τον υπολογιστή του σπιτιού τους μέσω ADSL modems.

Τα αποτελέσματα και των δύο πειραματικών εφαρμογών ήταν πολύ θετικά από πλευράς απόδοσης των συστημάτων και παρουσιάζονται στα άρθρα [ECM96] [ECM97] [ECM98]. Το ουσιαστικότερο όμως κέρδος ίσως αποτελούν οι πολύτιμες εμπειρίες που προέκυψαν από την ανάπτυξη και εγκατάσταση του συστήματος σε πραγματικό περιβάλλον

με πραγματικά δεδομένα και απαιτήσεις. Στον εξυπηρετητή που εγκαταστάθηκε στο Βέλγιο εφαρμόστηκαν με επιτυχία τα περισσότερα ζητήματα που αναπτύσσονται στην παρούσα διατριβή και περιγράφονται στην επόμενη ενότητα.

1.5 Περιγραφή της διατριβής

Το κεφάλαιο αυτό αποτελεί μια εισαγωγή στα ζητήματα που διαπραγματεύονται στην παρούσα διατριβή. Έγινε μια εισαγωγή στα δεδομένα πολυμέσων, τους εξυπηρετητές αποθήκευσης, ενώ υπήρξε επίσης αναφορά στα σημεία εκείνα που συνιστούν την συνεισφορά της διατριβής. Τέλος στο πρώτο κεφάλαιο περικλείεται μια συνοπτική περιγραφή του ευρωπαϊκού προγράμματος SICMA/ACTS στα πλαίσια των εργασιών του οποίου σχεδιάστηκε και αναπτύχθηκε η μεταπτυχιακή διατριβή.

Στο επόμενο κεφάλαιο παρουσιάζεται η σχετική έρευνα σε ζητήματα δρομολόγησης και ανάπτυξης εξυπηρετητών αποθήκευσης πολυμέσων που υπάρχει στην βιβλιογραφία. Γίνεται αναφορά στις συσκευές μαγνητικών δίσκων, στις πολιτικές δρομολόγησης δίσκου, στα χαρακτηριστικά των ψηφιακών δεδομένων πολυμέσων και στην ανάπτυξη υποσυστημάτων πολλαπλών δίσκων για την υποστήριξη των αντικειμένων ροής. Στην συνέχεια παραθέτονται ζητήματα και αρχές δρομολόγησης διεργασιών και μετάδοσης δεδομένων. Περιγράφονται γνωστοί στην βιβλιογραφία μηχανισμοί ανάπτυξης εξυπηρετητών αποθήκευσης, ενώ επίσης γίνεται εκτενής αναφορά προς το τέλος του κεφαλαίου σε θέματα κλιμακωτής επέκτασης των συστημάτων εξυπηρετητών πολυμέσων.

Στο 3ο κεφάλαιο παρουσιάζεται σε γενικές γραμμές ο σχεδιασμός του εξυπηρετητή αποθήκευσης. Περιγράφεται το σύστημα Kydonia πάνω στο οποίο στηρίχθηκε η διατριβή, εξετάζονται τα προβλήματα που εμφανίστηκαν σε προηγούμενους σχεδιασμούς και παρουσιάζεται η ενσωμάτωση μηχανισμών συμβατών με το πρότυπο DAVIC μέσα από μια συνολική τεχνική επισκόπηση του συστήματος.

Στο 4ο κεφάλαιο περιγράφονται αναλυτικά όλα τα θέματα ανάπτυξης της υπηρεσίας διαχείρισης δεδομένων ροής, καθώς επίσης και οι δομές υποστήριξης που χρησιμοποιούνται.

Στο 5ο κεφάλαιο λαμβάνουν χώρα μια σειρά από πειράματα στα υποσυστήματα της υπηρεσίας διαχείρισης δεδομένων ροής με στόχο την ανάλυση των απαιτήσεων αλλά και των χαρακτηριστικών των επιμέρους εργασιών που αναλαμβάνει να οδηγήσει ο μηχανισμός δρομολόγησης της υπηρεσίας ροής, ο οποίος σχεδιάζεται στο 6ο κεφάλαιο.

Ζητήματα κλιμάκωσης της υπηρεσίας ροής εξετάζονται στο 8ο κεφάλαιο, ενώ τέλος στο 9ο κεφάλαιο γίνεται η απαραίτητη ανακεφαλαίωση και οι στόχοι μελλοντικών επεκτάσεων.

Στο τέλος της παρούσας εργασίας παραθέτονται τρία παραρτήματα τεχνικής περιγραφής μερικών εφαρμογών που αναπτύχθηκαν και ένα παράρτημα με τις αναφορές στην σχετική βιβλιογραφία.

➤ Εξυπηρετητές αποθήκευσης δεδομένων και σχετική έρευνα

2

Ένας εξυπηρετητής αποθήκευσης δεδομένων ευαίσθητων στις καθυστερήσεις είναι ένα υπολογιστικό σύστημα που αποτελείται από ένα ή περισσότερους επεξεργαστές, μνήμη τυχαίας προσπέλασης και αρκετά αποθηκευτικά μέσα για την μόνιμη αποθήκευση των δεδομένων. Ο εξυπηρετητής πρέπει να έχει μια ολοκληρωμένη λύση στο πρόβλημα δρομολόγησης και διαχείρισης των πόρων. Στο κείμενο αυτό γίνεται μια προσπάθεια παρουσίασης της σχετικής έρευνας στην περιοχή. Περιγράφονται οι απαιτήσεις και τα χαρακτηριστικά των ψηφιακών δεδομένων πολυμέσων και παρουσιάζονται πολιτικές δρομολόγησης της πρόσβασης σε μαγνητικούς δίσκους. Στην συνέχεια αναλύεται το πρόβλημα υποστήριξης πολλών πελατών και γίνεται αναφορά σε αλγορίθμους δρομολόγησης διεργασιών και δρομολόγησης της μετάδοσης δεδομένων. Τέλος παρουσιάζονται τεχνικές ανάπτυξης πληροφορικών συστημάτων με δυνατότητα κλιμάκωσης ώστε να είναι εφικτή η αντιμετώπιση αναγκών εξυπηρέτησης μεγαλύτερου φόρτου αιτήσεων/πελατών με την προσθετική αύξηση των πόρων στο σύστημα.

2.1 Μαγνητικοί δίσκοι

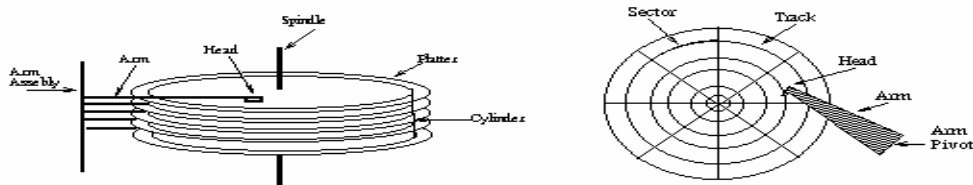
Η σύγχρονη τεχνολογία των επεξεργαστών αναπτύσσεται με γρήγορους ρυθμούς με ποσοστά που αγγίζουν το 40–60% κάθε χρόνο [Reum94]. Αν και η τεχνολογία των μαγνητικών μέσων αποθήκευσης εξελίσσεται εντυπωσιακά, ωστόσο η απόδοση των υποσυστημάτων αποθήκευσης παραμένει ένας κρίσιμος παράγοντας που επηρεάζει την συνολική απόδοση των υπολογιστικών συστημάτων. Πολλές έρευνες έχουν κατά καιρούς δημοσιευτεί με κύριο στόχο την βελτίωση της απόδοσης της δευτερεύουσας μνήμης, οι οποίες κυρίως αναφέρονται στους δίσκους. Οι έρευνες αυτές γίνονται συνήθως με αναλυτικά μοντέλα ή διάφορες προσομοιώσεις που χρησιμοποιούνται για την σύγκριση διαφορετικών προσεγγίσεων.

2.1.1 Χαρακτηριστικά των σύγχρονων οδηγών δίσκων

Πριν γίνει αναφορά στις πολιτικές δρομολόγησης σχετικές με δίσκους, χρήσιμη είναι μια περιγραφή της σύγχρονης τεχνολογίας οδηγών μαγνητικών δίσκων. Ο μηχανισμός των μαγνητικών δίσκων περιέχει τα τμήματα εγγραφής και τα τμήματα τοποθέτησης. Στα τμήματα εγγραφής περιλαμβάνονται οι περιστρεφόμενοι δίσκοι και οι κεφαλές που χρησιμοποιούνται για την πρόσβαση αυτών, ενώ στα τμήματα τοποθέτησης εντάσσονται ολόκληρος ο μηχανισμός του βραχίονα, που μετακινεί τις κεφαλές στην σωστή θέση, καθώς και τα συστήματα που οδηγούν την κεφαλή στα μονοπάτια του δίσκου (track following systems). Ο ελεγκτής δίσκου (disk controller) είναι επίσης ένα ακόμα εξάρτημα των οδηγών δίσκων. Ο ελεγκτής περιέχει ένα μικρο-επεξεργαστή, μνήμη αποθήκευσης ταχείας προσπέλασης και ένα σύστημα ζεύξης και επαφής με το υπόλοιπο σύστημα (μέσω του

δίσκου). Ο ελεγκτής διαχειρίζεται την αποθήκευση και ανάκτηση των δεδομένων στα και από τα τμήματα αποθήκευσης και δημιουργεί τις απεικονίσεις ανάμεσα στα λογικά τμήματα του δίσκου και τα φυσικά.

Τα τμήματα εγγραφής αποτελούνται από ταχέως πολλαπλούς περιστρεφόμενους δίσκους (platters) γύρω από κοινό άξονα. Τα δεδομένα εγγράφονται σε καθορισμένα μονοπάτια (tracks) πάνω στους δίσκους, τα οποία διαιρούνται σε ένα αριθμό από τομείς (sectors). Ένας κύλινδρος είναι ένα σύνολο από μονοπάτια (ένα από κάθε δίσκο) που απέχουν ίση απόσταση από τον κεντρικό άξονα. Συνεπώς η φυσική θέση εγγραφής ενός τμήματος δεδομένων καθορίζεται από τον κύλινδρο την επιφάνεια και το τομέα. Ο βραχίονας εγγραφής/ανάγνωσης φέρει ένα αριθμό από κεφαλές ίσο με τον αριθμό των διαθέσιμων δίσκων. Οι κεφαλές είναι διασυνδεδεμένες με τρόπο ώστε δεδομένης μια θέσης του βραχίονα όλες οι κεφαλές να βρίσκονται πάνω σε ένα κύλινδρο. Στο Σχήμα 1 φαίνονται σχηματικά τα χαρακτηριστικά των μαγνητικών δίσκων.



Σχήμα 1: Τα μηχανικά μέρη των οδηγών μαγνητικών δίσκων

Οι αιτήσεις ανάκτησης ή εγγραφής δεδομένων δέχονται καθυστερήσεις μηχανικού χαρακτήρα. Για παράδειγμα για την ανάκτηση ενός συγκεκριμένου τμήματος δεδομένων απαιτείται μετακίνηση του βραχίονα στον κατάλληλο κύλινδρο οπότε εμφανίζεται η καθυστέρηση αναζήτησης (seek delay). Στη συνέχεια η κεφαλή χρειάζεται να περιμένει ώστε τα κατάλληλα δεδομένα του μονοπατιού να περιστραφούν μέχρι το σημείο κάτω από την θέση της κεφαλής, οπότε έχουμε και την καθυστέρηση περιστροφής. Μετά από τις παραπάνω καθυστερήσεις η μετάδοση των δεδομένων από τον οδηγό του δίσκου μπορεί να αρχίσει.

Οι περισσότεροι δίσκοι κατέχουν ικανό υλικό λογικής και υπολογισμού για να προσφέρουν εύκολη ζεύξη με το υπόλοιπο σύστημα με πρωτόκολλα όπως το Small Computer System Interface (SCSI). Ο SCSI ελεγκτής του συστήματος στέλνει αιτήσεις στον οδηγό δίσκου με την μορφή αρχής λογικού τμήματος (logical block number LBN) και μεγέθους δεδομένων ανάκτησης/εγγραφής. Οι λεπτομέρειες της φυσικής πρόσβασης των δεδομένων είναι συνήθως αδιαφανείς για το υπόλοιπο σύστημα εκτός του οδηγού. Έτσι για παράδειγμα ένας SCSI δίσκος αντιμετωπίζεται σαν εάν γραμμικό διάνυσμα από τμήματα με διευθύνσεις μεγέθους από 256-1024 bytes. Διαχωρίζοντας σε λογικό και φυσικό επίπεδο τον τρόπο προσπέλασης των δεδομένων στον δίσκο επιτρέπεται στον οδηγό να αναλαμβάνει χαμηλού επιπέδου βελτιώσεις απόδοσης και να αποκρύπτει από το ανώτερο επίπεδο φυσικούς τομείς που έχουν υποστεί ζημιά αλλάζοντας την αντιστοίχιση των λογικών και φυσικών τμημάτων.

2.2 Πολιτικές δρομολόγησης δίσκου

Όταν ένα σύστημα χαρακτηρίζεται από την ανάγκη για την ικανοποίηση πολλών και συνεχόμενων αιτήσεων πρόσβασης της δευτερεύουσας μνήμης, τα μέσα αποθήκευσης αναδεικνύονται σε σοβαρό εμπόδιο απόδοσης στο σύστημα. Το μεγαλύτερο μέρος από την καθυστέρηση απόκρισης, την οποία γνωρίζει μια αίτηση προσπέλασης του δίσκου, οφείλεται στις μηχανικές μετακινήσεις που εξαρτώνται σε μεγάλο βαθμό από τις σχετικές θέσεις των διαφόρων δεδομένων προς εξυπηρέτηση και της τρέχουσας θέσης του βραχίονα κεφαλής. Η ύπαρξη συνεπώς ενός δρομολογητή ο οποίος θα καθοδηγεί την πρόσβαση στον δίσκο, καθορίζοντας την σειρά εκτέλεσης των αιτήσεων, μπορεί να βελτιώσει σημαντικά την απόδοση του μέσου.

Από τους πρώτους αλγόριθμους που εμφανίστηκαν στην βιβλιογραφία [Denn67] είναι η εξυπηρέτηση των αιτήσεων με κριτήριο το μικρότερο χρόνο αναζήτησης (Shortest Seek Time First) SSTF. Σύμφωνα με την πολιτική αυτή ο δρομολογητής επιλέγει για εξυπηρέτηση την αίτηση εκείνη η οποία θα εμφανίσει την μικρότερη καθυστέρηση αναζήτησης με δεδομένο την τρέχουσα θέση του βραχίονα κεφαλής. Συνήθως είναι δύσκολο να υπολογίσουμε τον ακριβή χρόνο της καθυστέρησης μεταξύ κυλίνδρων αλλά μπορούμε να έχουμε ένα αποτέλεσμα καλής προσέγγισης χρησιμοποιώντας τις αποστάσεις μεταξύ δύο διαφορετικών αναζητήσεων. Η εφαρμογή του SSTF για μεγάλο χρονικό διάστημα έχει σαν αποτέλεσμα την μείωση του μέσου χρόνου απόκρισης αλλά οδηγεί σε αύξηση της διαφοροποίησης των χρόνων απόκρισης καθώς κάτω από συνθήκες μεγάλου φόρτου εργασίας τείνει να ευνοεί την εξυπηρέτηση ενός υποσυνόλου αιτήσεων, που βρίσκονται σε μια περιοχή, αφήνοντας χωρίς εξυπηρέτηση αιτήσεις εκτός της περιοχής αυτής για μεγάλο χρονικό διάστημα.

Μια άλλη πολιτική η οποία μειώνει την διαφοροποίηση στους χρόνους απόκρισης είναι η SCAN, ενώ ταυτόχρονα εμφανίζει οριακή αύξηση στον μέσο χρόνο απόκρισης. Ο δρομολογητής, που ακολουθεί την πολιτική αυτή κατευθύνει τον βραχίονα μπροστά και πίσω σε ολόκληρη την περιοχή των κυλίνδρων από τον εξωτερικό μέχρι τον εσωτερικό και αντίθετα. Καθώς κινείται ο βραχίονας ο δρομολογητής εξυπηρετεί τις αιτήσεις που χρειάζονται προσπέλαση στην κατεύθυνση που ακολουθεί κάθε φορά. Ευνόητο είναι ότι ακολουθώντας αυτόν τον αλγόριθμο η κεφαλή επισκέπτεται τους μεσαίους κυλίνδρους πιο συχνά από ότι αυτούς στις άκρες.

Αρκετές παραλλαγές του SCAN αλγόριθμου έχουν προταθεί. Ο κυκλικός SCAN αλγόριθμος (C_SCAN) αντικαθιστά τον SCAN αλγόριθμο ακολουθώντας μια κατεύθυνση κατά την κίνηση του βραχίονα κεφαλής αντί για δύο. Έτσι όταν ο βραχίονας έρχεται πάνω από τον τελευταίο κύλινδρο, μετακινείται αμέσως στον πρώτο χωρίς να εξυπηρετήσει αιτήσεις κατά την μετακίνηση αυτή. Ο C-SCAN αντιμετωπίζει με δίκαιο τρόπο τον κάθε κύλινδρο αντίθετα με τον απλό SCAN που ευνοεί την πρόσβαση στους μεσαίους κυλίνδρους. Μια άλλη προσέγγιση αποτελεί ο LOOK αλγόριθμος ο οποίος αλλάζει την φορά κίνησης του βραχίονα κάθε φορά που δεν εκκρεμεί αίτηση προσπέλασης σε κυλίνδρους που βρίσκονται προς αυτήν κατεύθυνση κίνησης. Ο συνδυασμός του C-SCAN και LOOK οδηγεί σε μια άλλη παραλλαγή του C-LOOK αλγόριθμου.

Στο [Geis87] προτείνεται ο VSCAN(R) αλγόριθμος ο οποίος δημιουργεί μια συνέχεια ανάμεσα στον SSTF και LOOK. Η παράμετρος R δηλώνει πόσο τείνει η δρομολόγηση να διατηρήσει την τρέχουσα φορά κίνησης του βραχίονα. Ο VSCAN(0,0) είναι ισοδύναμος με τον SSTF και ο VSCAN(1.0) καταλήγει στον LOOK. Στο [Geis87] προτείνεται ότι ο VSCAN(0.2) προσφέρει μια καλή ισορροπία ανάμεσα στον μέσο χρόνο απόκρισης του SSTF και την αντίσταση στην αύξηση της διαφοροποίησης των χρόνων απόκρισης του LOOK.

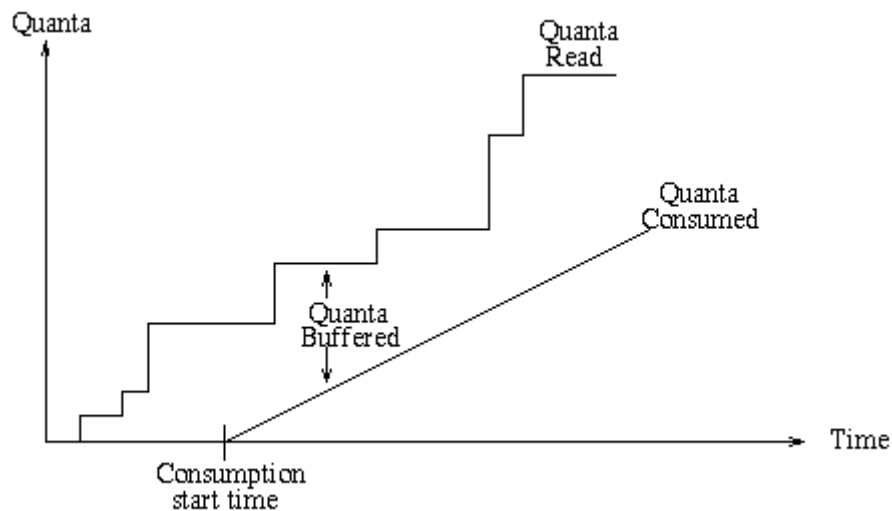
Η προσπάθεια για την μείωση της καθυστέρησης αναζήτησης προϋποθέτει μόνο γνώση των σχετικών αποστάσεων μεταξύ των τμημάτων δεδομένων που πρέπει να προσπελαστούν. Θεμιτός είναι όμως και ο περιορισμός της καθυστέρησης περιστροφής. Συνεπώς περισσότερη πληροφορία σχετικά με την ακριβή τοποθέτηση των δεδομένων στην επιφάνεια του δίσκου είναι απαραίτητη προκειμένου ο δρομολογητής να αποφασίζει την σειρά εκτέλεσης των αιτήσεων. Η πολιτική Shortest Access Time First (SATF) υποδεικνύει ότι ο δρομολογητής εξυπηρετεί την επόμενη αίτηση με βάση το μικρότερο χρόνο τοποθέτησης της κεφαλής, που προκύπτει από τον συνδυασμό της καθυστέρησης αναζήτησης και περιστροφής [Jacob91].

Όπως έχει αναφερθεί όλοι οι αλγόριθμοι που αποσκοπούν στην μείωση του μέσου χρόνου απόκρισης των διεργασιών (προσπαθώντας να δρομολογήσουν με κριτήριο τον μικρότερο χρόνο επεξεργασίας, οδηγούν) τελικά μετά από μεγάλο χρονικό διάστημα στην αύξηση της διαφοροποίησης των χρόνων απόκρισης και σε φαινόμενα ασιτίας (starvation). Διάφορες τεχνικές έχουν χρησιμοποιηθεί σε συνδυασμό με αυτούς τους αλγόριθμους για την αντιμετώπιση αυτού του προβλήματος. Οι πιο δημοφιλείς είναι οι τεχνικές batching και Aging. Η τεχνική batching επιτρέπει την εφαρμογή του αλγορίθμου σε υποσύνολα αιτήσεων με βάση διάφορα κριτήρια, ενώ η τεχνική aging προσθέτει μια παράμετρο βάρους σε κάθε αίτηση η οποία αυξάνεται συνεχώς όσο η αίτηση δεν τυγχάνει εξυπηρέτησης. Αναθέτοντας στην συνέχεια προτεραιότητες στις αιτήσεις οι οποίες προκύπτουν από την αρχική πολιτική συνδυασμένες με το τρέχων βάρος, γίνεται προσπάθεια να προσεγγιστεί η κατάλληλη δρομολόγηση.

Στο [Worth94] γίνεται μια μελέτη απόδοσης των περισσότερων αλγορίθμων δρομολόγησης δίσκου τόσο με τυχαία όσο και με πραγματικά σύνολα αιτήσεων ανάκτησης/εγγραφής δεδομένων. Τα αποτελέσματα δείχνουν ότι οι αλγόριθμοι που στοχεύουν στην μείωση του χρόνου αναζήτησης αλλά και του χρόνου περιστροφής, παρέχουν υψηλότερα επίπεδα απόδοσης. Ένα ακόμα θετικό στοιχείο που προκύπτει από την έρευνα [Worth94] είναι ότι το κέρδος που επιτυγχάνεται υιοθετώντας ένα αλγόριθμο μείωσης του χρόνου πρόσβασης εμπλουτισμένο με την γνώση της πλήρους αντιστοιχίας λογικών και φυσικών διευθύνσεων είναι μικρό και κυμαίνεται κάτω του 2%. Το αποτέλεσμα αυτό είναι ιδιαίτερα θετικό για την ανάπτυξη δρομολογητών καθώς μειώνεται η πολυπλοκότητά τους αλλά κυρίως γιατί συνήθως είναι δύσκολη η εύρεση της πλήρους αντιστοιχίας των φυσικών και λογικών επιπέδων.

2.3 Ψηφιακά δεδομένα συνεχόμενης ροής

Οι σύγχρονες απαιτήσεις των εφαρμογών υπολογιστών δεν μπορούν να περιοριστούν στις παραδοσιακές μορφές ψηφιακής πληροφορίας όπως είναι τα δεδομένα κειμένου και τα αριθμητικά. Η παρουσίαση πληροφορίας με video και ήχο είναι πλέον αναγκαία. Κύριο χαρακτηριστικό της ψηφιακής αναπαράστασης αυτών των τύπων δεδομένων αποτελούν οι μεγάλες απαιτήσεις που παρουσιάζουν ως προς τον χώρο αποθήκευσης. Συνήθως τα ψηφιακά δεδομένα συγκροτούνται από μια σειρά τμημάτων δεδομένων για την καλύτερη διαχείρισή τους, τα οποία αποκτούν σημασία μόνο αν αυτή η σειρά των τμημάτων παρουσιαστεί συνεχόμενα ως προς τον χρόνο. Για τον λόγο αυτό ονομάζονται και δεδομένα συνεχόμενης ροής (streams). Άμεση απόρροια των χαρακτηριστικών αυτών είναι η δυσκολία διαχείρισης τους από την σύγχρονη τεχνολογία αποθηκευτικών μέσων, κυρίως για τις υψηλές ανάγκες τους σε ρυθμούς ανάκτησης δεδομένων.



Σχήμα 2: Ανάκτηση και αναπαραγωγή του μέσου από τον δίσκο

Η αναπαραγωγή ενός μέσου συνεχόμενης ροής μπορεί να θεωρηθεί ως μια σειρά σειριακών διεργασιών με προθεσμίες, όπου οι διεργασίες αντιστοιχούν σε ανακτήσεις τμημάτων του μέσου από τον δίσκο και οι προθεσμίες αντιστοιχούν στους δρομολογημένους χρόνους αναπαραγωγής [Gem94a].

Λόγω των διαφορών στους ρυθμούς ανάκτησης, μετάδοσης και κατανάλωσης, τα δεδομένα πρέπει να αποταμιεύονται προσωρινά (buffering), για να επιτυγχάνεται η κατάλληλη μετατροπή των ρυθμών μεταξύ των διαφορετικών επιπέδων λειτουργίας. Έτσι η πρόκληση των δρομολογητών στους εξυπηρετητές αποθήκευσης είναι να καταφέρουν να εφοδιάζουν τους αποταμιευτές δεδομένων των πελατών με την κατάλληλη ποσότητα δεδομένων συνεχώς, ώστε ποτέ να μην παρατηρηθεί έλλειψη δεδομένων κατά την κατανάλωση. Το γεγονός αυτό φαίνεται γραφικά από το Σχήμα 2 [Gem94a]. Στην γραφική παράσταση η γραμμή σε σχήμα σκάλας αντιπροσωπεύει το μέγεθος των δεδομένων που ανακτώνται από τον δίσκο ως προς τον χρόνο. Η αναπαραγωγή του μέσου αρχίζει λίγο αργότερα με σταθερό ρυθμό κατανάλωσης δεδομένων. Για να είναι επιτυχημένη η

αναπαραγωγή θα πρέπει πάντα η γραμμή ανάκτησης να βρίσκεται πάνω από την γραμμή κατανάλωσης.

2.3.1 Διαχείριση αποθήκευσης και τοποθέτηση δεδομένων

Κάθε μέσο που αντιπροσωπεύει video ή ήχο συνήθως διαιρείται από τον εξυπηρετητή αποθήκευσης σε τμήματα δεδομένων καθώς αποθηκεύεται σε ένα δίσκο. Κάθε τμήμα του μέσου μπορεί να περιλαμβάνει αρκετούς φυσικούς τομείς. Η αποθήκευση των δεδομένων ενός αντικειμένου ροής μπορεί να είναι συνεχόμενη ή διασκορπισμένη ως προς το μέσο αποθήκευσης. Η πολιτική συνεχόμενης τοποθέτησης είναι εύκολη στην υλοποίηση αλλά εμφανίζει προβλήματα κατακερματισμού του διαθέσιμου χώρου και μεγάλο κόστος κατά τις ενημερώσεις (εφόσον απαιτείται αντιγραφή μεγάλων συνήθως μεγεθών δεδομένων). Αντίθετα η πολιτική της διασκορπισμένης τοποθέτησης δεν εμφανίζει ανάλογα μειονεκτήματα. Όσον όμως αφορά την αποθήκευση αντικειμένων ροής με την πολιτική συνεχόμενης τοποθέτησης, κατά την ανάκτηση απαιτείται μια ενέργεια αναζήτησης που θα τοποθετήσει την κεφαλή στην κατάλληλη θέση και στην συνέχεια όλα τα δεδομένα ανακτώνται σειριακά. Από την άλλη πλευρά η πολιτική διασκορπισμένης τοποθέτησης εμφανίζει πολλές καθυστερήσεις αναζήτησης, καθώς απαιτούνται επανατοποθετήσεις τις κεφαλής ανάμεσα στα τμήματα δεδομένων. Μια λογική προσέγγιση για την μείωση του κόστους είναι η επιλογή του κατάλληλου μεγέθους του τμήματος δεδομένων (το οποίο αποθηκεύεται σειριακά), ώστε περιοδικά όταν χρειάζεται ανάκτηση δεδομένων για την αναπαραγωγή ενός μέσου να απαιτείται μια μόνο ενέργεια αναζήτησης. Επίσης το μέγεθος του τμήματος που θα επιλεγεί για την αποθήκευση ενός μέσου ροής έχει άμεσες επιπτώσεις στον εξυπηρετητή αποθήκευσης, καθώς επηρεάζει ανάλογα το μέγεθος των δομών διεύθυνσης (indexes) και κατά συνέπεια την δυνατότητα του εξυπηρετητή να κρατά τις δομές αυτές στην κύρια μνήμη κατά την διάρκεια της αναπαραγωγής.

2.3.2 Δρομολογήσεις δίσκου για δεδομένα συνεχόμενης ροής

Η μηχανική φύση των μαγνητικών δίσκων εισάγει σημαντικές καθυστερήσεις στην ανάκτηση πληροφορίας. Η χρήσιμη εργασία ενός δίσκου λαμβάνει χώρα κατά το διάστημα μεταφοράς δεδομένων, ενώ κανένα κέρδος δεν επιφέρουν τα διαστήματα μηχανικών μετακινήσεων. Η πολύπλεξη των αιτήσεων ανάκτησης διαφορετικών αντικειμένων σε ένα δίσκο έχει σαν αποτέλεσμα ένα ποσοστό του χρόνου να καταναλώνεται σε άχρηστες επανατοποθετήσεις της κεφαλής. Ένας τρόπος για την μείωση αυτού του ποσοστού (και για την καλύτερη αξιοποίηση του μέσου) είναι να αυξήσουμε σημαντικά το μέγεθος των τμημάτων που ανακτώνται, ενέργεια όμως που έχει αντίκτυπο στις απαιτήσεις μνήμης. Εναλλακτικά [Ghan95] μπορούμε να μειώσουμε αυτό το ποσοστό και ταυτόχρονα τις απαιτήσεις σε αποταμιευτικό χώρο (buffering), μειώνοντας τις καθυστερήσεις αναζήτησης. Αυτό μπορεί να επιτευχθεί είτε ελέγχοντας τον τρόπο τοποθέτησης δεδομένων, είτε εφαρμόζοντας την κατάλληλη δρομολόγηση δίσκου, είτε επιβάλλοντας ένα συνδυασμό των προηγούμενων.

Στο τμήμα 2.2 έχουν περιγραφεί οι παραδοσιακές δρομολογήσεις δίσκου που είχαν σαν στόχο την αύξηση της απόδοσης, την δίκαιη αντιμετώπιση και την μείωση των μέσων χρόνων των καθυστερήσεων. Η προσθήκη όμως των χρονικών περιορισμών που

υπεισέρχονται στα δεδομένα συνεχόμενης ροής καθιστούν τους αλγόριθμους αυτούς ακατάλληλους. Πολλές προσπάθειες έχουν παρουσιαστεί στην βιβλιογραφία για συνδυασμό των παραδοσιακών αλγορίθμων δρομολόγησης δίσκου και των αλγορίθμων δρομολόγησης διεργασιών πραγματικού χρόνου.

Ο πιο δημοφιλής αλγόριθμος για την δρομολόγηση διεργασιών πραγματικού χρόνου ο Earliest Deadline first – EDF μπορεί να εφαρμοστεί με επιτυχία για την δρομολόγηση των αιτήσεων δίσκου. Όμως μια δρομολόγηση βασισμένη μόνο στον EDF είναι πιθανόν να οδηγήσει σε μεγάλες καθυστερήσεις αναζήτησης και περιστροφής και συνεπώς σε μη σωστή αξιοποίηση των αποθηκευτικών πόρων. Από την άλλη πλευρά η εφαρμογή του EDF αλγορίθμου προϋποθέτει την γνώση των προθεσμιών προκαταβολικά, ενώ ο χρόνος εξυπηρέτησης των αιτήσεων δίσκου ποικίλει λόγω των διαφορετικών σχετικών θέσεων πάνω στον δίσκο. Επίσης εφαρμόζεται σε διαδικασίες οι οποίες μπορούν να διακοπούν για χάρη μιας άλλης με μικρότερη προθεσμία και να συνεχίσουν αργότερα (preemptive), ιδιότητα που δεν μπορεί να εφαρμοστεί στους σύγχρονους δίσκους. Για τους παραπάνω λόγους μόνο παραλλαγές του πρωτότυπου EDF αναπτύσσονται για την υποστήριξη της δρομολόγησης των αιτήσεων πρόσβασης στον δίσκο.

Μια άλλη παραλλαγή συνδυάζει τον SCAN και τον EDF δίνοντας τον EDF-SCAN. Ο αλγόριθμος αυτός δρομολογεί με βάση τις προθεσμίες, αλλά για αιτήσεις που φέρουν τις ίδιες προθεσμίες εφαρμόζει την πολιτική SCAN. Συνεπώς η αποτελεσματικότητα του EDF-SCAN εξαρτάται από το ποσοστό των αιτήσεων που έχουν τις ίδιες προθεσμίες. Έτσι ο EDF-SCAN μπορεί να συνδυαστεί με μία τεχνική ανάθεσης περιοδικών προθεσμιών, ώστε όλες οι αιτήσεις σε μια περίοδο να έχουν τις ίδιες προθεσμίες [Red94].

Μια ενδιαφέρουσα τεχνική υλοποίησης του EDF-SCAN έχει προταθεί [Red94], χρησιμοποιώντας στον EDF διαφοροποιημένες προθεσμίες. Αν υποθέσουμε ότι D_i είναι οι προθεσμίες των αιτήσεων ανάκτησης και N_i είναι οι θέσεις των μονοπατιών (tracks) των δεδομένων προς ανάκτηση, τότε διαφοροποιούμε τις αρχικές προθεσμίες σύμφωνα με τον τύπο : $D_i - f(N_i)$, όπου $f()$ είναι μια συνάρτηση η οποία μετατρέπει τους αριθμούς των μονοπατιών σε μικρές παρέλξεις των προθεσμιών. Οι παρέλξεις αυτές είναι αρκετά μικρές ώστε να ισχύει ότι $D_i + f(N_i) > D_j + f(N_j)$ αν $D_i > D_j$. Συνεπώς εφαρμόζοντας στις αιτήσεις με τις διαφοροποιημένες προθεσμίες την δρομολόγηση EDF, έχουμε στην ουσία τα ίδια αποτελέσματα με την EDF-SCAN.

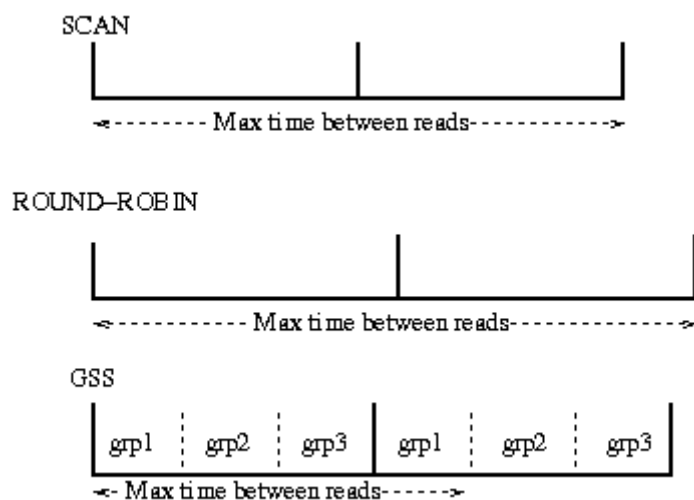
Ο T-SCAN αλγόριθμος έχει προταθεί και αυτός για την αντιμετώπιση της δρομολόγησης δεδομένων συνεχούς ροής με διαφορετικά χαρακτηριστικά. Οι διαφορετικές απαιτήσεις σε ρυθμό ανάκτησης δεδομένων έχει σαν συνέπεια οι αιτήσεις ανάκτησης να έχουν διαφορετική περίοδο εμφάνισης. Ο αλγόριθμος T-SCAN μετασχηματίζει όλες αυτές τις περιόδους σε μια ενιαία περίοδο και καθορίζει το ποσό πληροφορίας που ανακτάται σε κάθε μια νέα περίοδο. Στην συνέχεια εφόσον όλες οι αιτήσεις αποκτούν την ίδια περίοδο εμφάνισης εφαρμόζει την SCAN δρομολόγηση δίσκου.

Οι περισσότερες προσεγγίσεις ανάκτησης δεδομένων ροής σε συστήματα στηρίζονται στην εξυπηρέτηση βασισμένη σε κύκλους. Η αντιμετώπιση αυτή συμβαδίζει με την φύση των δεδομένων ροής που είναι περιοδική [Gem94a]. Έτσι σε κάθε κύκλο ένας αριθμός

τμημάτων ανακτάται για κάθε αντικείμενο ροής. Για την αντιμετώπιση της ανάκτησης αντικειμένων με διαφορετικούς ρυθμούς αναπαραγωγής και για την καλύτερη δυνατή αξιοποίηση των αποθηκευτικών μέσων, η οποία εκφράζεται με την μεγιστοποίηση των αντικειμένων ροής που μπορεί να υποστηριχθούν ταυτόχρονα, έχει αποδειχθεί ότι ο αριθμός των τμημάτων που ανακτώνται σε κάθε κύκλο πρέπει να είναι ανάλογος του ρυθμού αναπαραγωγής αυτών [Vin93].

Η πιο απλή προσέγγιση του προβλήματος βασίζεται στην τεχνική round-robin σύμφωνα με την οποία η εξυπηρέτηση των αιτήσεων ανάκτησης γίνεται σε κύκλους και η σειρά με την οποία εξυπηρετούνται οι αιτήσεις είναι σταθερή. Η τεχνική όμως round-robin όπως και η EDF δεν εκμεταλλεύεται τις σχετικές θέσεις των τμημάτων που ανακτώνται σε κάθε κύκλο. Για τον λόγο αυτό μπορεί να συνδυαστεί με την εφαρμογή πολιτικών δρομολόγησης δίσκου που στοχεύουν στην ελαχιστοποίηση των χρόνων προσπέλασης. Ορισμένες συνδυασμένες πολιτικές έχουν προταθεί, ενώ επίσης για τον ίδιο σκοπό έχουν αναπτυχθεί και τεχνικές τοποθέτησης δεδομένων στον δίσκο με περιορισμούς (constrained placement).

Η εφαρμογή του SCAN αλγορίθμου σε κάθε κύκλο μπορεί να επιφέρει καλή αξιοποίηση του δίσκου και ελαχιστοποίηση των καθυστερήσεων αναζήτησης. Η γνώση των αιτήσεων του επόμενου κύκλου μπορεί να βελτιώσει τον SCAN σε pre-seeking SCAN επιτρέποντας μια ενέργεια αναζήτησης (seek) στο τέλος του γύρου ώστε η κεφαλή να είναι στην κατάλληλη θέση στην αρχή του επόμενου κύκλου [Gem94b].



Σχήμα 3: Κυκλική επεξεργασία με εφαρμογή δρομολογήσεων δίσκου

Η εφαρμογή του SCAN κατά την επεξεργασία κύκλων αντίθετα με τον round-robin μπορεί να οδηγήσει μια αίτηση να εξυπηρετηθεί στην αρχή ενός κύκλου και στο τέλος του επόμενου κύκλου, ανάλογα με τις σχετικές θέσεις των τμημάτων δημιουργώντας με αυτόν τον τρόπο μια διαφορά δύο κύκλων μεταξύ διαδοχικών εξυπηρετήσεων στην χειρότερη περίπτωση. Το γεγονός αυτό έχει άμεσες επιπτώσεις στον χρόνο απόκρισης και στον χώρο των αποταμιευτών του συστήματος. Για την αντιμετώπιση αυτών των μειονεκτημάτων

προτάθηκε ο Grouped-Sweeping Scheme – GSS ο οποίος διαιρεί τον κύκλο σε ένα αριθμό ομάδων [Yu93]. Κάθε αίτηση ανάκτησης αντικειμένου ροής ανατίθεται σε συγκεκριμένη ομάδα.

Οι ομάδες εξυπηρετούνται με σταθερή σειρά σε κάθε κύκλο ενώ οι αιτήσεις που ανήκουν σε μια ομάδα δέχονται την SCAN δρομολόγηση. Με τον τρόπο αυτό και με την κατάλληλη επιλογή του αριθμού των ομάδων μειώνεται η αρχική καθυστέρηση και οι απαιτήσεις χώρου στους αποταμιευτές του συστήματος σε σχέση με την απλή εφαρμογή του SCAN, με το τίμημα όμως μιας ελεγχόμενης αύξησης της μέσης καθυστέρησης αναζήτησης.

Στο Σχήμα 3 αναπαριστώνται γραφικά οι πολιτικές round-robin, SCAN και GSS , ενώ επίσης διακρίνονται οι αρχικοί χρόνοι απόκρισης και αναλογικά οι απαιτήσεις σε χώρο αποταμιευτών.

Η πολιτική τοποθέτησης REBECA προσεγγίζει τον στόχο ελάττωσης του χρόνου αναζήτησης θέτοντας όρια στην απόσταση μεταξύ δύο τμημάτων τα οποία ανακτώνται σειριακά [Ghan95]. Η REBECA τμηματοποιεί τον χώρο του δίσκου σε R περιοχές. Στην συνέχεια τα διαδοχικά τμήματα ενός αντικειμένου αναθέτονται στις περιοχές ακολουθώντας zigzag μορφή. Με τον τρόπο αυτό ελαχιστοποιούνται οι αποστάσεις που διανύει η κεφαλή καθώς 1) η κεφαλή κινείται κάθε φορά προς μια κατεύθυνση, 2) σε κάθε χρονική περίοδο εξυπηρετούνται εκείνα τα αντικείμενα που αντιστοιχούν σε μια περιοχή (ενεργή περιοχή) και 3) σε κάθε επόμενη χρονική περίοδο εξυπηρετούνται οι περιοχές που προηγούνται ή έπονται της ενεργής ανάλογα προς την κατεύθυνση κίνησης της κεφαλής. Βέβαια η τεχνική αυτή αναδεικνύει ένα αμοιβαίο τίμημα (trade off) μεταξύ της απόδοσης (throughput) και της αρχικής καθυστέρησης (start up latency). Εναλλακτική πρόταση αποτελεί μια παραλλαγή της REBECA η πολιτική OREO, σύμφωνα με την οποία η ανάθεση των τμημάτων γίνεται κυκλικά στις περιοχές. Με την OREO επιτυγχάνεται μείωση της μέγιστης αρχικής καθυστέρησης κατά 50%. Ενώ η OREO-GSS πολιτική συνδυάζει την OREO εφαρμόζοντας σε κάθε ομάδα την GSS.

Η επιλογή μια ολοκληρωμένης πολιτικής για την ανάκτηση δεδομένων συνεχόμενης ροής έχει άμεσες επιπτώσεις στον απαιτούμενο χώρο αποταμιευτών που πρέπει να υπάρχει διαθέσιμος. Αποδεικνύεται ότι δεδομένου ενός αριθμού N αντικειμένων ροής που πρέπει να εξυπηρετηθούν το μέγεθος του τμήματος που πρέπει να χρησιμοποιηθεί είναι διαφορετικό και ανάλογο του συνολικού χρόνου αναζήτησης που παρατηρείται με την κάθε τεχνική. Γεγονός που μεταφράζεται στο ότι όσο μεγαλύτερη είναι η μείωση που επιτυγχάνεται στις καθυστερήσεις αναζήτησης, τόσο μικρότερη είναι και η απαίτηση σε μέγεθος των αποταμιευτών.

Επίσης σημαντικές διαφοροποιήσεις στον απαιτούμενο αποταμιευτικό χώρο επέρχονται από διάφορες στρατηγικές διαχείρισης μνήμης που μπορεί να εφαρμοστούν από τον δρομολογητή ανάκτησης των δεδομένων ροής. Η πιο απλή προσέγγιση του προβλήματος διαχείρισης της προσωρινής μνήμης είναι η δέσμευση για κάθε διαφορετικό αντικείμενο ροής, για το οποίο έχει αιτηθεί ανάκτηση, χώρου ίσου με $2*B$ όπου B το μέγεθος του τμήματος που πρέπει να ανακτηθεί σε κάθε κύκλο. Το διπλάσιο μέγεθος που

δεσμεύεται, προκύπτει από την εφαρμογή σχήματος διπλού αποταμιευτή (double-buffering), κατά το οποίο ο μισός χώρος χρησιμοποιείται για την ανάκτηση των δεδομένων από τον δίσκο και ο άλλος μισός για την αποστολή των δεδομένων στο δίκτυο. Αν υποθέσουμε ότι πρέπει να εξυπηρετηθούν N ισοδύναμα αντικείμενα ροής τότε η συνολική απαίτηση σε χώρο προσωρινής μνήμης ανέρχεται σε $2 \cdot N \cdot B$ [Ozd94].

Στο [Ghan95] παρουσιάζονται δυο εναλλακτικές μέθοδοι οι οποίες επιτυγχάνουν μείωση στον μέγιστο απαιτούμενο χώρο, επιτρέποντας σε τμήματα των αποταμιευτών να διαμοιράζονται από διαφορετικά αντικείμενα ροής. Η τεχνική Coarse-Grain sharing διατηρεί ένα σύνολο από θέσεις μνήμης που χρησιμεύουν για την προσωρινή αποθήκευση των τμημάτων δεδομένων. Κάθε διεργασία (είτε ανάκτησης, είτε αποστολής δεδομένων) δεσμεύει θέσεις τμημάτων στον διαμοιραζόμενο αποταμιευτή κατά απαίτηση. Μόλις η διεργασία επεξεργαστεί τα δεδομένα του τμήματος που δέσμευσε, ελευθερώνει τη θέση. Αν η τεχνική αυτή εφαρμοστεί στην GSS δρομολόγηση οδηγεί σε περιορισμό της απαίτησης σε μνήμη σε σύγκριση με την προηγούμενη απλή προσέγγιση όπως φαίνεται από τον ακόλουθο τύπο, όπου g ο αριθμός των υπο-ομάδων που χρησιμοποιούνται στο σχήμα GSS:

$$M_{coarse} = \left(N + \left\lceil \frac{N}{g} \right\rceil \right) \times B$$

Η Δέσμευση των τμημάτων λαμβάνει χώρα στην αρχή εξυπηρέτησης κάθε υποομάδας (υπο-κύκλου), ενώ ταυτόχρονα ελευθερώνονται οι θέσεις εκείνες οι οποίες είχαν δεσμευθεί πριν από ένα κύκλο και ένα υπο-κύκλο.

Από την άλλη πλευρά η Fine-Grain sharing τεχνική προσπαθεί να εκμεταλλευτεί το γεγονός ότι καθώς οι διεργασίες επεξεργάζονται τα δεδομένα στα τμήματα ένα ορισμένο ποσοστό των δεδομένων στον αποταμιευτή δεν χρειάζεται πια. Για παράδειγμα η διεργασία αποστολής δεδομένων αποστέλλει με ορισμένο ρυθμό τα δεδομένα ενός τμήματος στους πελάτες με αποτέλεσμα σταδιακά τα δεδομένα που βρίσκονται στην αρχή του τμήματος να δεσμεύουν χώρο παρά το γεγονός ότι έχουν καταναλωθεί. Συνεπώς η FCS προσπαθεί να αυξήσει το ποσοστό της διαμοιραζόμενης μνήμης, εισάγοντας μεγαλύτερη ακρίβεια στον τρόπο δέσμευσης του χώρου του αποταμιευτή. Έτσι η μονάδα δέσμευσης μειώνεται σε σελίδα μνήμης (memory page) αντί για ολόκληρο τμήμα όπως στη CGS τεχνική. Αν και ο ακριβής υπολογισμός της μνήμης που απαιτείται εφαρμόζοντας την FCS είναι δύσκολος, εντούτοις στο [Ghan95] γίνεται μία προσπάθεια υπολογισμού μια ενδεικτικής τιμής η οποία δείχνει ότι η FCS τεχνική επιτυγχάνει μεγαλύτερη μείωση στον απαιτούμενο χώρο από ότι η CGS.

Στο [Ozd94] παρουσιάζεται μια άλλη τεχνική τοποθέτησης δεδομένων με περιορισμούς και ονομάζεται τοποθέτηση βασισμένη σε πίνακα (Matrix-based). Η τεχνική αυτή φιλοδοξεί να εξουδετερώσει εντελώς τις επιπτώσεις των καθυστερήσεων του δίσκου και συνεπώς να κρατήσει τις απαιτήσεις σε αποταμιευτικό χώρο χαμηλές. Η εφαρμογή της όμως καταλήγει σε αύξηση του χειρότερου χρόνου αρχικής απόκρισης. Σύμφωνα με [Ozd94] όλα τα video αποθηκεύονται στον δίσκο συνεχόμενα και σειριακά το ένα μετά το άλλο, έτσι ώστε να

δημιουργήσουν ένα υπέρ-video. Υποθέτουμε ότι τα δεδομένα ανακτώνται από τον δίσκο σε τμήματα μεγέθους d , ότι όλα τα video έχουν τον ίδιο ρυθμό αναπαραγωγής r_{med} και ότι η μέγιστη χρονική διάρκεια του υπέρ-video είναι l . Ο μέγιστος θεωρητικά αριθμός video που μπορούν να εξυπηρετηθούν ταυτόχρονα είναι $p = r_{disk}/r_{med}$. Το υπέρ-video χωρίζεται σε p τμήματα χρονικής διάρκειας $tc = l/p$ και κάθε τμήμα σε $n = (tc \times r_{med})/d$. Μετά από αυτήν την επεξεργασία το υπέρ-video μπορεί να θεωρηθεί ως ένας πίνακας $p \times n$ με γραμμές τα δημιουργημένα τμήματα και στήλες τα κομμάτια των τμημάτων. Αν στον δίσκο αποθηκευτούν σειριακά οι στήλες αυτού του πίνακα τότε είναι δυνατόν να υποστηριχθούν τα p video με σειριακή ανάκτηση των δεδομένων από τον δίσκο χωρίς μετακινήσεις αναζήτησης και με μέγιστο χρόνο αρχικής καθυστέρησης tc , εφόσον η ανάκτηση μιας στήλης του πίνακα διαρκεί λιγότερο ή ίσο της αναπαραγωγής ενός τμήματος, δηλαδή $(p \times d)/r_{disk} \leq d/r_{med}$.

Το σχήμα διαχείρισης της προσωρινής μνήμης που προτείνεται με αυτή την τεχνική αποτελείται από p αποταμιευτές-γραμμές. Κάθε φορά που μια στήλη ανακτάται από τον δίσκο το κάθε στοιχείο (τμήμα) αποθηκεύεται στην αντίστοιχη γραμμή-αποταμιευτή. Σύμφωνα με την τεχνική νέα δεδομένα γράφονται στους αποταμιευτές γραμμές με ρυθμό r_{med} .

Αν και η τεχνική matrix-based εκμηδενίζει τις καθυστερήσεις αναζήτησης καταλήγοντας σε σειριακή ανάκτηση εκμεταλλευόμενη τον μέγιστο ρυθμό μεταφοράς δεδομένων του δίσκου, ωστόσο έχει πολλά μειονεκτήματα που ακολουθούν και οι υπόλοιπες τεχνικές τοποθέτησης με περιορισμούς που περιγράφηκαν. Οι πολλές υποθέσεις και οι περιορισμοί όπως η ισοδυναμία των αντικειμένων ροής και ο τρόπος τοποθέτησης των αντικειμένων (που αναδεικνύουν απαγορευτικές τις ενημερώσεις, νέες εισαγωγές και διαγραφές αντικειμένων, καθώς είναι αναγκαία η μαζική αναδιοργάνωση της αποθήκευσης), μειώνουν την αποτελεσματικότητα της εφαρμογής αυτών των πολιτικών σε ένα πραγματικό σύστημα εξυπηρετητή αποθήκευσης. Επίσης σημαντική είναι και η επίπτωση της αύξησης του μέγιστου χρόνου αρχικής απόκρισης που επιφέρουν αυτές οι πολιτικές.

Όπως προαναφέρθηκε η πιο απλή προσέγγιση της διαχείρισης προσωρινής μνήμης είναι το σχήμα διπλού αποταμιευτή που οδηγεί σε δέσμευση $2 \times B \times N$ χώρου. Όταν μια αίτηση υποβάλλεται προς εκτέλεση αποκτά μια προθεσμία η οποία είναι ίση με τον χρόνο υποβολής συν τον χρόνο μίας περιόδου. Η προθεσμία αυτή καθορίζει και τον χρόνο αρχικής απόκρισης. Αν όμως διαφοροποιήσουμε την προθεσμία [Red94] προσθέτοντας μια ακόμα περίοδο δηλαδή τον χρόνο υποβολής συν $2 \times p$, τότε η διεργασία αποστολής δεδομένων εργάζεται $2p$ χρόνο πίσω από την διεργασία ανάκτησης. Το γεγονός αυτό δίνει την δυνατότητα στον δρομολογητή δίσκου να επιτελέσει περισσότερες βελτιώσεις μειώνοντας του χρόνου αναζήτησης, από ότι θα μπορούσε διαφορετικά εφόσον σε αυτή την περίπτωση έχει περισσότερο χρόνο να εκτελέσει μια αίτηση. Όταν χρησιμοποιούμε ανάκτηση μεγάλων τμημάτων δεδομένων ή διαφοροποιημένες προθεσμίες προκαλούμε αύξηση στην αρχική απόκριση και απαιτείται μεγαλύτερος αποταμιευτικός χώρος. Από την άλλη μεριά όμως επιτυγχάνουμε αύξηση της απόδοσης και μεγιστοποιούμε τον αριθμό των μέσων ροής που μπορούν να εξυπηρετηθούν ταυτόχρονα. Αυτά τα αμοιβαία τμήματα πρέπει να αξιολογηθούν και να εκτιμηθούν κατά την ανάπτυξη ενός εξυπηρετητή αποθήκευσης.

Πειράματα έδειξαν [Red94] ότι η εφαρμογή των διαφοροποιημένων προθεσμιών βελτιώνει σε σημαντικό βαθμό τον μέγιστο αριθμό μέσων που μπορούν να αναπαραχθούν ταυτόχρονα για όλες τις πολιτικές δρομολόγησης δίσκου (CSCAN, EDF, SCAN-EDF), οδηγώντας όμως σε υψηλές απαιτήσεις σε διαθέσιμο χώρο προσωρινής μνήμης. Αν η δέσμευση του απαραίτητου χώρου δεν είναι εφικτή τότε μπορεί να χρησιμοποιηθούν μικρότερα τμήματα δεδομένων εφόσον πειράματα έδωσαν αποτελέσματα που ευνοούν την εφαρμογή διαφοροποιημένων προθεσμιών παρά την αύξηση των τμημάτων ανάκτησης.

2.3.3 Δρομολόγηση πολλαπλών δίσκων

Στο προηγούμενο υποκεφάλαιο αναλύθηκε η αποθήκευση δεδομένων ροής σε ένα δίσκο. Ο αριθμός όμως των ανακτήσεων αντικειμένων ροής που μπορούν να υποστηριχθούν ταυτόχρονα περιορίζεται από την απόδοση του δίσκου. Η πιο απλή προσέγγιση για την αύξηση του αριθμού αυτού είναι η χρήση πολλαπλών δίσκων οι οποίοι λειτουργούν ως αντίγραφα. Η τοποθέτηση νέων δίσκων αντιγράφων σε ένα σύστημα επιτρέπει την αύξηση του συνολικού αριθμού πελατών που είναι δυνατόν να εξυπηρετηθούν. Μεγάλο τίμημα όμως αποτελεί η ανάγκη για πολλαπλάσιο χώρο από αυτόν που χρειάζονται τα μέσα για αποθήκευση. Ο περιορισμός αυτός οδηγεί στην μείωση του αριθμού των διαφορετικών μέσων ροής που ο εξυπηρετητής δύναται να αποθηκεύσει. Για την μείωση αυτού του κόστους υπάρχουν πολιτικές που διαφοροποιούν τον αριθμό των αντιγράφων με κριτήριο πόσο δημοφιλής είναι η αναπαραγωγή ενός μέσου με στόχο την εξοικονόμηση αποθηκευτικού χώρου.

Μια πιο αποδοτική τεχνική είναι η διασπορά των δεδομένων των αντικειμένων ροής σε πολλούς δίσκους και κυρίως εφαρμόζεται με δύο τεχνικές: την data striping και την data interleaving [Chun96][Gem94a]. Η χρήση της πρώτης τεχνικής λαμβάνει χώρα συνήθως σε σχήματα πολλών δίσκων σε σχήμα RAID. Έτσι τα δεδομένα ενός αντικειμένου ροής διαιρούνται και αποθηκεύονται σε κάθε δίσκο. Η προσπέλαση των δεδομένων σε ένα τέτοιο σχήμα γίνεται παράλληλα και οι αντίστοιχοι φυσικοί τομείς από κάθε δίσκο προσπελούνται παράλληλα σαν μεγάλοι λογικοί τομείς. Οι δίσκοι στο σχήμα RAID είναι συγχρονισμένοι ως προς την περιστροφή και εργάζονται παράλληλα με τον ίδιο τρόπο. Άμεσο αποτέλεσμα της πολιτικής data striping είναι η μεγάλη αύξηση του ρυθμού μεταφοράς δεδομένων αλλά δεν προσφέρει βελτιώσεις όσον αφορά την μείωση των χρόνων αναζήτησης. Από την άλλη πλευρά το μεγάλο συνήθως μέγεθος των λογικών τμημάτων οδηγεί σε μεγάλη καθυστέρηση αρχικής απόκρισης και σε αυξημένες απαιτήσεις αποθηκευτικού χώρου.

Ακολουθώντας την data interleaving τεχνική τα δεδομένα ενός μέσου αποθηκεύονται σε πολλούς δίσκους με ένα σχήμα ώστε διαδοχικά τμήματα δεδομένων να αναθέτονται σε διαφορετικούς δίσκους. Το πιο απλό σχήμα αναθέτει τα τμήματα δεδομένων κυκλικά σε ένα σύνολο δίσκων. Αντίθετα με το σχήμα RAID σε αυτήν την περίπτωση οι δίσκοι είναι δεν είναι συγχρονισμένοι και λειτουργούν ανεξάρτητα. Η δρομολόγηση της ανάκτησης σύμφωνα με αυτή την τοποθέτηση μπορεί να ακολουθήσει το σχήμα RAID δηλαδή σε κάθε κύκλο εξυπηρέτησης ανακτάται ένα τμήμα από κάθε δίσκο για την αναπαραγωγή ενός μέσου. Αυτός ο τρόπος ανάκτησης οδηγεί σε ισορροπία ως προς τον φόρτο εργασίας των δίσκων

αλλά με τα μειονεκτήματα που έχουν αναφερθεί. Μια εναλλακτική λύση αποτελεί η πολιτική ανάκτησης ενός μόνο τμήματος από ένα δίσκο σε κάθε κύκλο εξυπηρέτησης για κάθε αντικείμενο ροής. Κατά συνέπεια η αναπαραγωγή λαμβάνει χώρα με την ανάκτηση τμημάτων κυκλικά από κάθε δίσκο σε ένα σύνολο N δίσκων. Για την μεγιστοποίηση της απόδοσης είναι απαραίτητη η επίτευξη ισορροπίας φόρτου εργασίας στο σύνολο των δίσκων που μπορεί να εγγυηθεί με διάφορες τεχνικές. Αρκετή έρευνα υπάρχει στην βιβλιογραφία για διάφορες τεχνικές striping όπως [Ber94] [Ozd95] [Tob93].

Μια διαφορετική προσέγγιση στο ζήτημα υποστήριξης πολλαπλών δίσκων είναι η εφαρμογή υβριδικών τεχνικών που συνδυάζουν τις πολιτικές striping, interleaving, replication [Chun96]. Από την πλευρά της δρομολόγησης το κέρδος από την εφαρμογή υβριδικών συστημάτων είναι αμφίβολο, αλλά έχοντας ένα σύστημα που μπορεί να υποστηρίξει διάφορες στρατηγικές είναι θετικό.

Ένα πρόβλημα στην υποστήριξη πολλαπλών ασύγχρονων δίσκων είναι η επιλογή της μεθοδολογίας δρομολόγησης. Στην δρομολόγηση βασισμένη σε κύκλους ο δρομολογητής είναι υπεύθυνος να συντονίζει την πρόσβαση στα αποθηκευτικά μέσα αλλά και την λειτουργία τους, διατηρώντας τους κύκλους εξυπηρέτησης. Στο [Vin94] προτείνονται δύο διαφορετικοί τρόποι αντιμετώπισης, είτε με μια συντηρητική πολιτική είτε με μια επιθετική (conservative/aggressive servicing policy). Στην συντηρητική στρατηγική ο δρομολογητής προσπαθεί να ανακτήσει όλα τα τμήματα δεδομένων για ένα κύκλο από όλους τους δίσκους πριν την έναρξη του επόμενου κύκλου. Αυτό σημαίνει ότι η διάρκεια του κύκλου εξυπηρέτησης καθορίζεται από το μεγαλύτερο χρόνο, από τους χρόνους που χρειάστηκαν να διανύσουν όλοι οι δίσκοι προκειμένου να εξυπηρετήσουν όλες τις αιτήσεις του κύκλου. Άμεση συνέπεια αυτής της πολιτικής είναι ότι οι δίσκοι που διεκπεραιώνουν γρηγορότερα τις αιτήσεις του κύκλου παραμένουν αδρανείς μέχρι να τελειώσει ο πιο φορτωμένος δίσκος. Ευνόητο είναι ότι το γεγονός αυτό προκαλεί μερική αξιοποίηση των αποθηκευτικών μέσων. Για την αντιμετώπιση της μειωμένης αξιοποίησης μπορεί να εφαρμοστεί η επιθετική πολιτική κατά την οποία ο δρομολογητής επιτρέπει την έναρξη της επεξεργασίας αιτήσεων του $(r+1)$ κύκλου μόλις ο δίσκος εκπληρώσει τις αιτήσεις ανάκτησης του r κύκλου και τελικά συγχρονίσει τους δίσκους μετά από $k \geq 1$ κύκλους. Οι προσομοιώσεις έδειξαν ότι η επιθετική πολιτική με διάφορους συνδυασμούς, τόσο με το αριθμό των πελατών όσο και με το μέγεθος των τμημάτων, επιτυγχάνει μεγαλύτερο ποσοστό αξιοποίησης σε σχέση με την συντηρητική πολιτική αλλά όπως είναι φυσικό εις βάρος του διαθέσιμου χώρου προσωρινής μνήμης. Ενδιαφέρον είναι το αποτέλεσμα ότι αν ένας δρομολογητής έχει στην διάθεσή του τον χώρο που χρειάζεται για την εφαρμογή της επιθετικής πολιτικής, η οποία συγχρονίζει τους δίσκους μετά από D κύκλους, τότε η εφαρμογή της συντηρητικής πολιτικής με διάρκεια κύκλου $r \times D$ εμφανίζει μεγαλύτερη απόδοση (throughput), εφόσον έχει την δυνατότητα να ανακτήσει όλα τα τμήματα διασχίζοντας τον δίσκο σύμφωνα με τον SCAN. Άμεσο κόστος αποτελεί ο μεγάλος χρόνος αρχικής απόκρισης.

Η υποστήριξη πολλαπλών δίσκων από ένα εξυπηρετητή αποθήκευσης πολυμέσων καθιστά πιθανή την ύπαρξη ετερογενών αποθηκευτικών μέσων. Δυο είναι κυρίως οι τρόποι αντιμετώπισης του προβλήματος. Μια κατηγορία τεχνικών διαχωρίζουν το σύνολο των δίσκων σε υποομάδες με βάση το μοντέλο τους οι οποίες διαχειρίζονται ανεξάρτητα. Για την

αποφυγή φαινομένων συμφόρησης (bottleneck) γίνονται αντίγραφα των συχνά ζητούμενων μέσων ανάμεσα στις υποομάδες (BSR, DASD dancing, Heat-Based τεχνικές). Οι τεχνικές όμως που δεν επιβάλλουν διαχωρισμό των μέσων αποθήκευσης φαίνεται ότι επιτυγχάνουν καλύτερα αποτελέσματα. Σε αυτή την κατηγορία ανήκουν οι τεχνικές disk grouping, staggered grouping, disk merging. Οι τρεις τεχνικές βασίζονται στην αντιστοίχιση ομάδων φυσικών δίσκων σε λογικούς με ενιαία χαρακτηριστικά. Στην τεχνική disk grouping τα τμήματα δεδομένων των πολυμέσων αποθηκεύονται κυκλικά στους λογικούς δίσκους και κάθε τμήμα χωρίζεται σε μικρότερα κομμάτια που ονομάζονται fragments και αποθηκεύονται στους φυσικούς δίσκους. Το μέγεθος των κομματιών αυτών ποικίλει έτσι ώστε ο χρόνος ανάκτησης αυτών να είναι περίπου ίδιος για όλους τους δίσκους. Η staggered grouping αποτελεί μια βελτίωση της προηγούμενης με μικρότερες απαιτήσεις σε προσωρινή μνήμη, ενώ στην disk merging πολιτική το υποσύστημα των μέσων αποθήκευσης μοντελοποιείται διαχωρίζοντας εντελώς την έννοια των λογικών δίσκων από τους φυσικούς επιτρέποντας την δημιουργία λογικών δίσκων από κλάσματα των διαθέσιμων εύρους μεταφοράς και χωρητικότητας των φυσικών δίσκων.

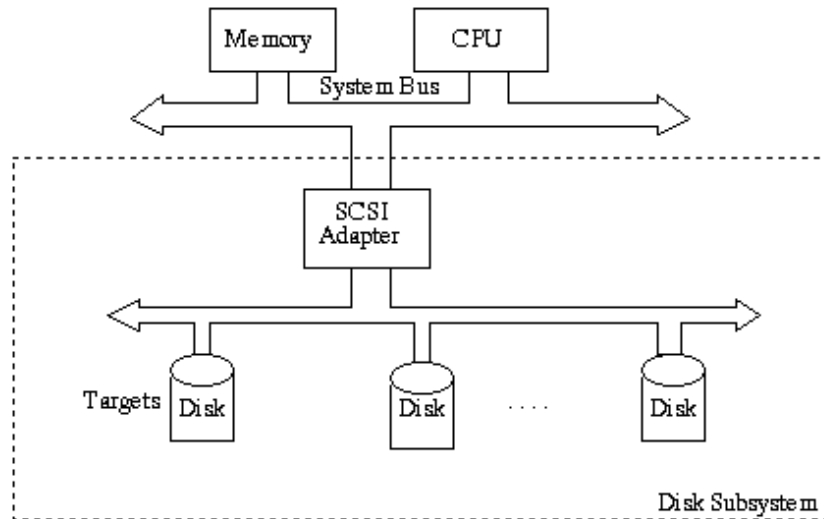
2.4 Επισκόπηση υποσυστήματος υποστήριξης δίσκων

Στο Σχήμα 4 φαίνεται η αρχιτεκτονική ενός υπολογιστικού συστήματος που υποστηρίζει πολλαπλά μέσα αποθήκευσης. Ο δίαυλος του συστήματος (system bus) διασυνδέει επικοινωνιακά με πολύ μεγάλες ταχύτητες μέρη του συστήματος όπως ο επεξεργαστής και η κύρια μνήμη. Εκεί βρίσκεται επίσης και ο προσαρμογέας οδηγών δίσκου, ο οποίος είναι υπεύθυνος για την επικοινωνία μεταξύ του συστήματος και των μέσων αποθήκευσης. Τα πιο συνήθη περιφερειακά buses που χρησιμοποιούνται είναι το SCSI (Small Computer Interface) και το IPI (Intelligent Peripheral Interface) ενώ πολλά άλλα έχουν εφαρμοστεί στους προσωπικούς υπολογιστές.

Το πρότυπο SCSI αποτελεί ταυτόχρονα προσδιορισμό του δίαυλου και ένα σύνολο εντολών που χρησιμοποιούν τον δίαυλο (bus). Τα αποθηκευτικά μέσα προσκολλώνται στο SCSI bus και ονομάζονται στόχοι (targets). Ο σκοπός του SCSI προτύπου είναι η παροχή μια αφαίρεσης ανεξάρτητης των τεχνικών χαρακτηριστικών διαφορετικών αποθηκευτικών μέσων. Κάθε στόχος έχει σταθερή προτεραιότητα και αντιστοιχεί στο αναγνωριστικό του SCSI (ID). Το χαρακτηριστικό αυτό δημιουργεί προβλήματα ιδιαίτερα στην υποστήριξη ενός εξυπηρετητή πολυμέσων που έχει ανάγκη από υψηλούς ρυθμούς μετάδοσης, καθώς όταν περισσότεροι από ένας δίσκος προσπαθούν να μεταφέρουν δεδομένα τότε αυτός ο δίσκος που έχει την μεγαλύτερη προτεραιότητα αποκτά την άδεια να χρησιμοποιήσει το bus, οδηγώντας σε μείωση της αξιοποίησης του άλλου δίσκου. Αυτή η πολυπλεξία χρήσης πολλαπλών δίσκων που επιχειρείται μέσω ενός σειριακού μέσου, εμφανίζει φαινόμενα συμφόρησης σε περιπτώσεις υψηλού φόρτου εργασίας παρά το γεγονός ότι το εύρος μεταφοράς δεδομένων του SCSI bus μπορεί θεωρητικά να καλύπτει ένα αριθμό δίσκων.

Επίσης μια άλλη παράμετρος που πρέπει να λαμβάνεται υπόψη κατά τον σχεδιασμό ενός εξυπηρετητή δεδομένων ροής, σε σχέση με την αποφυγή φαινομένων συμφόρησης

στο SCSI bus, είναι το μέγεθος του τμήματος που ανακτάται από τον δίσκο, καθώς μεγάλα τμήματα μπορεί μεν να βελτιώνουν την απόδοση των δίσκων αλλά οδηγούν σε μεγαλύτερους χρόνους αναμονής για τους δίσκους με μικρότερη προτεραιότητα. Ένα γεγονός που μπορεί να καταλήξει σε παραβίαση των προθεσμιών ανάκτησης [Red94].



Σχήμα 4: Υποσύστημα υποστήριξης δίσκων

Πειράματα που έχουν λάβει χώρα για την δυνατότητα κλιμάκωσης του SCSI bus ως προς τον αριθμό των υποστηριζόμενων δίσκων [Gan95] έδειξαν ότι όσο αυξάνουν οι δίσκοι τόσο μειώνεται ο αθροιστικός ρυθμός μεταφοράς δεδομένων που μπορεί να επιτευχθεί σε σχέση με το άθροισμα των ρυθμών μετάδοσης που προκύπτουν από την χρήση καθενός δίσκου χωριστά στο SCSI bus.

Συμπερασματικά θα μπορούσε κανείς να αναφέρει ότι πρέπει να προσαρμόζονται λιγότεροι δίσκοι σε ένα υποσύστημα SCSI από αυτούς που θεωρητικά μπορεί να υποστηρίξει ώστε να έχουμε την μέγιστη αξιοποίηση των δίσκων.

2.5 Δρομολόγηση πραγματικού χρόνου

Ένας εξυπηρετητής αποθήκευσης ψηφιακών δεδομένων συνεχόμενης ροής εκτός από την διαχείριση των μέσων αποθήκευσης και του διαθέσιμου χώρου καλείται να καλύψει και τις απαιτήσεις πραγματικού χρόνου που συμπεριλαμβάνονται σε δεδομένα όπως video και ήχος. Το γεγονός αυτό αναδεικνύει την ανάγκη ύπαρξης μιας πολιτικής δρομολόγησης που θα διαχειρίζεται και θα συγχρονίζει όλους τους πόρους του συστήματος. Τα συστήματα υπολογιστών έχουν συνήθως πόρους που πρέπει να μοιράζονται σειριακά κυρίως τους επεξεργαστές αλλά και άλλους που είτε είναι φυσικοί (πόροι δικτύου) είτε είναι λογικοί (συγχρονιστές).

2.5.1 Επισκόπηση δρομολόγησης

Η δρομολόγηση είναι ο ορισμός της σειράς με την οποία ανταγωνιστικές διεργασίες προσπελαίνουν ένα σειριακά διαμοιραζόμενο πόρο. Ο δρομολογητής αναθέτει κάθε φορά την άδεια στην πιο κατάλληλη διεργασία –σύμφωνα με την πολιτική δρομολόγησης- να αποκτήσει πρόσβαση στον διαμοιραζόμενο πόρο (dispatching). Η δρομολόγηση μπορεί να είναι διακοπτόμενη (preemptive), δηλαδή σε κάποια χρονική στιγμή η πιο κατάλληλη διεργασία μπορεί να πάρει την θέση της τρέχουσας διακόπτοντας την εκτέλεσή της ή non-preemptive κατά την οποία μια διεργασία χρησιμοποιεί τον πόρο για όσο χρόνο χρειάζεται ή μέχρι να σταματήσει για κάποιο λόγο.

Άλλη κατηγοριοποίηση των πολιτικών δρομολόγησης περιλαμβάνει αλγορίθμους στατικούς και δυναμικούς. Ο στόχος της δρομολόγησης δεν είναι απλώς ο καθορισμός της σειριακής πρόσβασης σε ένα πόρο αλλά προσπαθεί να λύσει και το πρόβλημα της συμφόρησης των πόρων (resource contention), σύμφωνα με ένα κριτήριο βελτιστοποίησης που είναι συνήθως σχετικό των εφαρμογών. Για παράδειγμα στα συστήματα πραγματικού χρόνου το χρονικό κριτήριο περιλαμβάνει την εγγύηση ότι όλες οι αυστηρές προθεσμίες (hard deadlines) που έχουν ανατεθεί πρέπει να ικανοποιηθούν, ενώ στα συστήματα πραγματικού χρόνου με μειωμένες απαιτήσεις (soft real-time systems) το χρονικό κριτήριο είναι η ελαχιστοποίηση των αναμενόμενων περιπτώσεων παραβίασης προθεσμιών [Cor96].

Συνήθως η δρομολόγηση εφαρμόζεται σε περιπτώσεις ύπαρξης πολλών διεργασιών ή νημάτων διεργασιών (processes, threads), που τρέχουν σε ένα επεξεργαστή, χωρίς βέβαια να αποκλείονται και άλλοι πόροι του συστήματος όπως δίκτυο, αποταμιευτές, αντικείμενα συγχρονισμού κτλ .

Η δρομολόγηση μπορεί να λάβει χώρα στατικά σαν μέρος του σχεδιασμού ενός συστήματος ή της ρύθμισής του (design & configuration). Σε αρκετά συστήματα πραγματικού χρόνου γίνεται στατική δρομολόγηση με τον κατακερματισμό του χρόνου σε τμήματα (frame-based), χρησιμοποιώντας κυκλική σειρά εκτέλεσης των διεργασιών ή με την ανάθεση σταθερών προτεραιοτήτων στις εργασίες οι οποίες δεν αλλάζουν κατά την εκτέλεση. Οι καθορισμένες προτεραιότητες συνήθως εξαρτώνται από το είδος της εφαρμογής αλλά υπάρχουν και μεθοδολογίες όπως η rate-monotonic ή deadline-monotonic ανάλυση που αναθέτουν σταθερές προτεραιότητες σε περιοδικές εργασίες με αυστηρές προθεσμίες. Οι υπάρχουσες δυναμικές δρομολογήσεις ασχολούνται με την διαχείριση και ανάθεση προτεραιοτήτων σε εργασίες. Η ανάθεση δυναμικών προτεραιοτήτων γίνεται συνήθως από τις εφαρμογές παρά από το λειτουργικό σύστημα. Αρκετοί αναλυτικά βασισμένοι αλγόριθμοι έχουν προταθεί στην περιοχή όπως ο EDF για την ανάθεση προτεραιοτήτων σε περιοδικές και απεριοδικές διεργασίες.

Οι δρομολογήσεις βασίζονται σε ένα κριτήριο βελτιστοποίησης (optimization factor) που τυπικά περιλαμβάνει παραμέτρους λογικής ορθότητας όπως εξαρτήσεις μεταξύ των διεργασιών, που υποδηλώνουν ποιες πρέπει να προηγούνται ως πιο σημαντικές, ή προκύπτουν από τον έλεγχο ταυτοχρονισμού (concurrency control). Για ένα σύνολο διεργασιών που η κάθε μία έχει περιορισμούς στον χρόνο εκτέλεσης (deadline), το κριτήριο βελτιστοποίησης περιλαμβάνει επίσης μια παράμετρο χρονικής ορθότητας που καθορίζει ότι

το σύνολο των διεργασιών εκτελείται σε χρόνους αποδεκτούς για το σύστημα. Για hard real-time συστήματα η παράμετρος αυτή σημαίνει ότι πάντα όλες οι προθεσμίες πρέπει να ικανοποιούνται, ενώ στα soft real-time συστήματα η χρονική παράμετρος περιλαμβάνει την ελαχιστοποίηση του αριθμού των προθεσμιών που παραβιάζονται, όπως έχει αναφερθεί. Συχνά η λογικές και χρονικές παράμετροι σε ένα κριτήριο βελτιστοποίησης βρίσκονται σε αντίθεση. Για παράδειγμα μια δρομολόγηση με βάση τον EDF μπορεί να οδηγήσει σε μια εκτέλεση διεργασιών που δεν είναι σειριακά σωστή (serializable). Συνήθως το κριτήριο βελτιστοποίησης σε μια δρομολόγηση οδηγεί στην επιλογή ενός αλγορίθμου δρομολόγησης. Η σχέση όμως αυτή δεν είναι ένα προς ένα. Χρειάζεται μια πιο πολύπλοκη προσέγγιση δρομολόγησης που πρόκειται να εφαρμοστεί σε ένα σύστημα, καθώς πολλά διαφορετικά κριτήρια δεν μπορούν να ικανοποιηθούν από την επιλογή ενός αλγορίθμου.

Σε ένα σύστημα υπάρχουν πολλοί πόροι που διαχειρίζονται από επιμέρους δρομολογητές. Για παράδειγμα πολλές υπηρεσίες μπορεί να έχουν συνδεθεί με ουρές αναμονής που επιβάλλουν FIFO δρομολόγηση στην πρόσβαση της υπηρεσίας. Αν όμως οι διαφορετικοί πόροι σε ένα σύστημα δρομολογούνται με διαφορετικούς επιμέρους δρομολογητές τελικά η συνολική συμπεριφορά του συστήματος ενδέχεται να μην συμφωνεί με το κριτήριο βελτιστοποίησης ολόκληρου του συστήματος. Ανάλογα, σε ένα διανεμημένο σύστημα που περιέχει διεργασίες που τρέχουν σε διαφορετικούς κόμβους και συνθέτουν μια εργασία με χρονικούς περιορισμούς, η σωστή λειτουργία προϋποθέτει ότι κάθε επιμέρους υπολογιστική ή επικοινωνιακή διεργασία χρειάζεται να δρομολογηθεί με τρόπο ώστε να επιτυγχάνεται μια συνοχή σε συνολικό επίπεδο. Έτσι συμπερασματικά δύο είναι τα ζητήματα σε περιπτώσεις δρομολόγησης συστημάτων ευρείας κλίμακας σε διανεμημένο περιβάλλον: 1) η συνοχή στην δρομολόγηση των επιμέρους υπολογιστικών διεργασιών που συνθέτουν τις διεργασίες υψηλού επιπέδου σε ένα κόμβο και 2) η συνοχή στην δρομολόγηση στα υπολογιστικά και επικοινωνιακά επιμέρους κομμάτια των λογικών διεργασιών που τρέχουν σε πολλούς κόμβους [Cor96].

2.5.2 Πολιτικές δρομολόγησης διεργασιών

Οι εφαρμογές πολυμέσων έχουν ανάγκη από πρόσβαση, επεξεργασία και μετάδοση ψηφιακών δεδομένων. Εφόσον τα δεδομένα συνεχόμενης ροής, όπως έχει αναφερθεί, αποκτούν την σωστή σημασία μόνο όταν παρουσιάζονται συνεχόμενα στον χρόνο και με καθορισμένο ρυθμό, είναι επόμενο εφαρμογές που χρησιμοποιούν δεδομένα ροής να θέτουν περιορισμούς πραγματικού χρόνου στα υποκείμενα συστήματα επεξεργασίας, αποθήκευσης και μετάδοσης. Ένα από τα σημαντικότερα θέματα είναι η δρομολόγηση στην κεντρική μονάδα επεξεργασίας. Για την επιλογή της κατάλληλης δρομολόγησης χρειαζόμαστε να γνωρίζουμε τα χαρακτηριστικά των εφαρμογών ή διεργασιών που πρέπει να δρομολογηθούν.

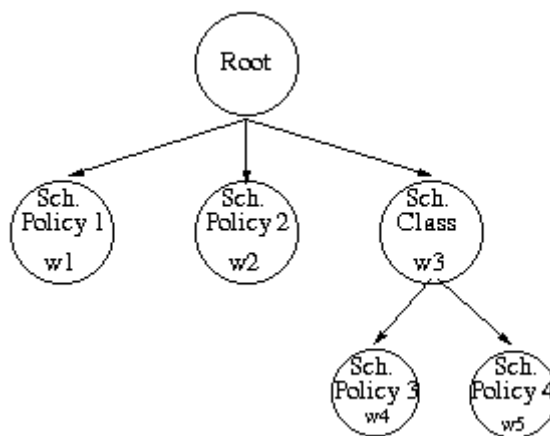
Οι διεργασίες που καλούνται να δρομολογηθούν χωρίζονται στις εξής κατηγορίες:

- 1) περιοδικές διεργασίες οι οποίες πρέπει να εκτελούνται σε σταθερά διαστήματα,
- 2) απεριοδικές με ευέλικτες προθεσμίες που εκτελούνται σε μη σταθερά διαστήματα

- 3) σποραδικές που αποτελούν απεριοδικές διεργασίες με αυστηρές προθεσμίες και ελάχιστη χρονική διάρκεια επανεμφάνισης.

Οι παραδοσιακές μέθοδοι δρομολόγησης διεργασιών στα λειτουργικά συστήματα έχουν σαν στόχο την αύξηση της απόδοσης, την μείωση της απόκρισης και την δίκαια αντιμετώπιση (πολλές έχουν μελετηθεί και εφαρμοστεί όπως round-robin, priority scheduling, multiple queues, Shortest Job First, Policy Driven scheduling [Tan87]).

Οι εφαρμογές με αυστηρές απαιτήσεις πραγματικού χρόνου χρειάζονται ντετερμινιστικές εγγυήσεις για την καθυστέρηση που μπορεί να έχουν ορισμένες διεργασίες. Συνηθισμένοι αλγόριθμοι όπως EDF και RMA θεωρούνται ικανοποιητικοί για ανάλογες εφαρμογές. Ο RMA υποθέτει ότι όλες οι διεργασίες είναι περιοδικές, έχουν προθεσμία ίση με την περίοδο, είναι ανεξάρτητες και διακοπτόμενες και τέλος έχουν σταθερό χρόνο εκτέλεσης. Ο RMA δρομολογεί με βάση προτεραιότητες και αναθέτει την μεγαλύτερη προτεραιότητα στην διεργασία με την μικρότερη περίοδο. Οι εφαρμογές με περιορισμένες απαιτήσεις πραγματικού χρόνου απαιτούν στατιστική εγγύηση ποιότητας εξυπηρέτησης με παραμέτρους όπως μέγιστη καθυστέρηση και απόδοση. Εφαρμογές που διαχειρίζονται δεδομένα video και ήχου μπορούν να τοποθετηθούν σε αυτή την κατηγορία. Τέλος οι εφαρμογές χωρίς χρονικούς περιορισμούς δεν έχουν ανάγκη αυστηρών εγγυήσεων απόδοσης αλλά αιτούν εκείνη την εξυπηρέτηση που θα οδηγήσει σε μικρό μέσο χρόνο απόκρισης και υψηλή απόδοση. Οι πολιτικές διαμοιραζόμενου χρόνου (Time sharing) είναι ο συνήθης τρόπος αντιμετώπισης τέτοιων διεργασιών.



Σχήμα 5: Δέντρο δρομολόγησης

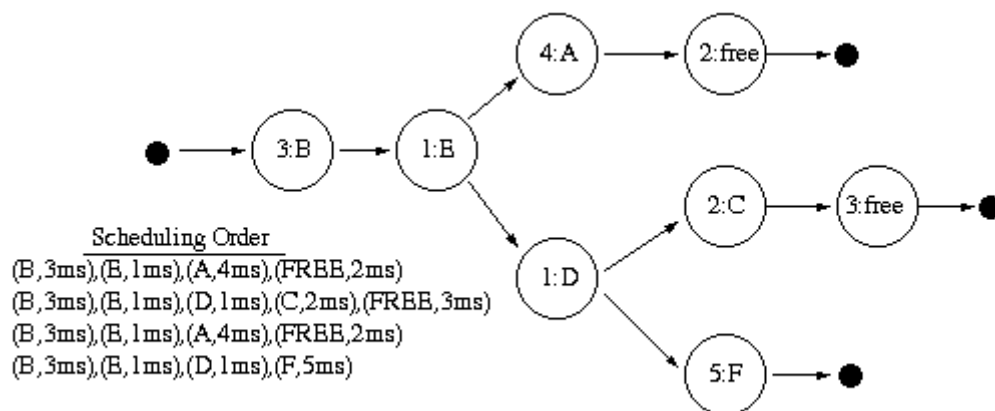
Η υποστήριξη εφαρμογών ή διεργασιών που ανήκουν σε διαφορετικές κατηγορίες χρειάζεται μια διαφοροποιημένη δρομολόγηση. Η ιεραρχική τμηματοποίηση του εύρους της CPU (hierarchical partitioning) προτείνεται στο [Goy96]. Ο δρομολογητής του επεξεργαστή στηρίζεται σε ένα δέντρο. Κάθε διεργασία στο σύστημα ανήκει σε ένα φύλλο του δέντρου και έτσι κάθε κόμβος στο δέντρο αντιπροσωπεύει είτε κλάση εφαρμογών είτε μια συνάθροιση κλάσεων εφαρμογών. Έτσι καθώς τα φύλλα δρομολογούνται ανάλογα με τον αλγόριθμο δρομολόγησης της κλάσης στην οποία ανήκουν, οι κόμβοι πρέπει να δρομολογηθούν με τέτοιο τρόπο ώστε να επιτυγχάνεται η κατάλληλη τμηματοποίηση της χρήσης του

επεξεργαστή. Για αυτήν την διαφοροποίηση χρησιμοποιούνται βάρη που αναθέτονται στους κόμβους. Το βάρος καθορίζει το ποσοστό χρήσης του επεξεργαστή που πρέπει να έχει ο κόμβος από το ποσοστό χρήσης του κόμβου πατέρα.

Ο Start-time Fair Queuing αλγόριθμος εγγυάται δικαιοσύνη στην εξυπηρέτηση διεργασιών οδηγώντας στον αναλογικό καταμερισμό του εύρους του επεξεργαστή. Με αυτόν τον στόχο γίνεται ανάθεση σε κάθε διεργασία μιας ετικέτας. Κάθε ετικέτα αρχικοποιείται στο χρόνο υποβολής της αίτησης εκτέλεσης και στην συνέχεια αυξάνεται σε κάθε τμήμα εξυπηρέτησης του επεξεργαστή (CPU quantum), με μέγεθος διαφορετικό για κάθε διεργασία και αντιστρόφως ανάλογο του βάρους της. Στην συνέχεια η δρομολόγηση των διεργασιών γίνεται με βάση την αύξουσα σειρά των ετικετών.

Ο SFQ εφαρμόζεται στην δρομολόγηση του επεξεργαστή και προκύπτει ως παραλλαγή των αλγορίθμων από την περιοχή των δικτύων όπως ο Weighted Fair Queuing, ο Fair queuing και ο Self-Clocked Fair Queuing.

Μια άλλη προσέγγιση δρομολόγησης των διεργασιών γίνεται στο [Jon97]. Ο μηχανισμός δρομολόγησης που προτείνεται βασίζεται στην λειτουργία κρατήσεων στον επεξεργαστή (CPU reservations). Οι κρατήσεις αυτές έχουν την μορφή «κράτηση Χ μονάδες χρόνου για κάθε Υ μονάδες» και εξαρτώνται από την κάθε εφαρμογή. Χρονικοί περιορισμοί μπορούν ωστόσο να εισαχθούν ώστε να υπάρχει δρομολόγηση με βάση τις προθεσμίες. Η δρομολόγηση στηρίζεται σε ένα προϋπολογισμένο γράφο κρατήσεων που έχει την μορφή που φαίνεται στο Σχήμα 6.



Σχήμα 6: Γράφος δρομολόγησης

Ο γράφος του σχήματος προέρχεται από τις κρατήσεις των διεργασιών A-F, με A(4,20), B(3,10), C(2,40), D(1,20), E(1,10), F(5,40)-διεργασία(χρόνος_δέσμευσης, περίοδος). Κάθε κόμβος του γράφου αναπαριστά ένα περιοδικό διάστημα χρόνου που είτε είναι δεσμευμένο στην εκτέλεση κάποιας διεργασίας είτε είναι ελεύθερο. Κάθε μονοπάτι από αριστερά προς τα δεξιά έχει σταθερό μήκος (για το σχήμα 10ms). Αυτό το μήκος είναι η βασική περίοδος του γράφου δρομολόγησης. Ο δρομολογητής επαναληπτικά διασχίζει τον γράφο από αριστερά προς τα δεξιά, ακολουθώντας εναλλακτικά μονοπάτια κάθε φορά που συναντά ένα κόμβο. Επόμενο είναι οι κόμβοι που ακολουθούν τους κόμβους διακλάδωσης

να δρομολογούνται τις μισές φορές από αυτούς. Το γεγονός αυτό επιτρέπει την δρομολόγηση κρατήσεων με διαφορετικές περιόδους χρησιμοποιώντας τον ίδιο γράφο με την υπόθεση ότι κάθε περίοδος είναι πολλαπλάσια της βασικής περιόδου ως προς δύναμη του δύο. Αν αυτό δεν συμβαίνει τότε μετατρέπουμε την κράτηση ώστε να έχει την κατάλληλη περίοδο. Με αυτήν την δρομολόγηση χρησιμοποιώντας τον γράφο δεν έχουμε μεγάλο επιπρόσθετο φόρτο δρομολόγησης, ενώ η ανάλυση για τυχόν περιορισμούς χρόνου μπορεί εύκολα να γίνει, εφόσον είναι γνωστό πότε μια διεργασία πρόκειται να εκτελεστεί και πότε υπάρχει ελεύθερος χρόνος. Η ικανοποίηση τυχόν χρονικών περιορισμών μπορεί να γίνει με επιβολή του EDF αλγορίθμου για τις διεργασίες καθώς διασχίζεται ο γράφος.

2.5.3 Δρομολόγηση μετάδοσης δεδομένων

Ένας εξυπηρετητής αποθήκευσης δεδομένων ροής είναι υπεύθυνος για την διαχείριση των αποθηκευτικών μέσων, την ανάκτηση και μετάδοση των δεδομένων στους πελάτες μέσω δικτύου. Ο μηχανισμός δρομολόγησης της ροής δεδομένων πρέπει να αντιμετωπίζει με συνοχή (όπως αναφέρθηκε στο προηγούμενο τμήμα) όλες τις διεργασίες στο σύστημα από την δρομολόγηση των αιτήσεων δίσκου, την πρόσβαση στους αποταμιευτές δεδομένων, μέχρι την αποστολή δεδομένων στο δίκτυο.

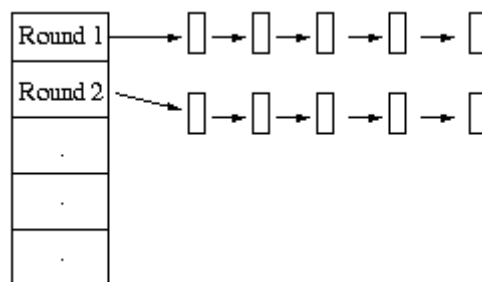
Αρκετές είναι οι πολιτικές δρομολόγησης που είναι γνωστές στην βιβλιογραφία, οι οποίες περιγράφουν τεχνικές ανάθεσης εύρους (bandwidth allocation) και δρομολόγηση μετάδοσης σε διακόπτες (switches). Οι πολιτικές αυτές παρουσιάζουν ενδιαφέρον αν αντιστοιχίσουμε τον εξυπηρετητή με ένα διακόπτη. Σαν είσοδο ο εξυπηρετητής δέχεται δεδομένα από τον δίσκο, όπως ο διακόπτης δέχεται πακέτα από την σύνδεση εισόδου, τα αποταμιεύει προσωρινά και τα μεταδίδει με την μορφή πακέτων δεδομένων στην έξοδο με κατάλληλο τρόπο (store&forward switch).

Η δρομολόγηση FIFO είναι ίσως η πιο εύκολη πολιτική για υλοποίηση αλλά δεν προσφέρει διαφοροποίηση μεταξύ ανεξάρτητων συνόδων (sessions) που χρειάζονται ντετερμινιστικές εγγυήσεις εύρους μετάδοσης. Γενικά οι δρομολογητές μπορούν να χαρακτηριστούν ως work-conserving και non work-conserving. Ένας δρομολογητής ανήκει στην πρώτη κατηγορία όταν δεν παραμένει αδρανής εφόσον υπάρχει ένα τουλάχιστον πακέτο αποταμιευμένο στον εξυπηρετητή μετάδοσης. Αντίθετα σύμφωνα με την δεύτερη κατηγορία ο δρομολογητής μπορεί να παραμένει αδρανής παρά το γεγονός ότι υπάρχουν πακέτα διαθέσιμα προς αποστολή. Αυτό για παράδειγμα μπορεί να συμβεί στην περίπτωση που η μετάδοση ενός πακέτου καθυστερεί, ενώ αναμένεται ένα μεγαλύτερης προτεραιότητας πακέτο να αφιχθεί σύντομα. Επίσης non work-conserving αλγόριθμοι χρησιμοποιούνται για τον έλεγχο του delay και του jitter, καθυστερώντας τα πακέτα που φτάνουν νωρίτερα. Παραδείγματα work-conserving δρομολογητών αποτελούν οι αλγόριθμοι : Generalized processor sharing GPS, Weighted fair queuing, Virtual clock, Delay-earliest-due-date, Weighted round-robin, Deficit round-robin. Από την άλλη πλευρά οι δρομολογήσεις HRR, Rate-controlled, Stop-and-go queuing, Jitter-earliest-due-date, αποτελούν παραδείγματα non work-conserving προσεγγίσεων.

Άλλη κατηγοριοποίηση βασίζεται στην εσωτερική δομή των δρομολογητών. Σύμφωνα με αυτή την κατηγοριοποίηση υπάρχουν δύο κύριες αρχιτεκτονικές. Η μια στηρίζεται στην

ανάθεση προτεραιοτήτων και στην δρομολόγηση με βάση τις προτεραιότητες αυτές, ενώ η άλλη χωρίζει τον χρόνο σε λογικά τμήματα εξυπηρέτησης (frames). Σε ένα δρομολογητή ο οποίος ανήκει στην πρώτη κατηγορία υπάρχει μια μεταβλητή που συνήθως αναφέρεται ως εικονικός χρόνος και σχετίζεται με κάθε σύνδεση. Κάθε φορά που ένα πακέτο έρχεται, εξυπηρετείται και η μεταβλητή αυτή ανανεώνεται. Ένα χρονικό στίγμα (timestamp) συνδέεται με κάθε πακέτο στο σύστημα και υπολογίζεται ως συνάρτηση του εικονικού χρόνου. Στην συνέχεια τα πακέτα δρομολογούνται ανάλογα με την σειρά των χρονικών στιγμάτων. Πολιτικές όπως Virtual clock, WFQ και Delay-EDD ακολουθούν αυτήν την αρχιτεκτονική. Εναλλακτικά στην αρχιτεκτονική βασισμένη σε frames ο χρόνος κατακερματίζεται σε τμήματα σταθερού ή μεταβλητού μήκους. Έτσι γίνονται κρατήσεις από τις συνόδους με την μορφή του μέγιστου όγκου δεδομένων που επιτρέπεται να μεταδοθεί από κάθε σύνοδο σε κάθε τμήμα του χρόνου.

Η μετάδοση από ένα εξυπηρετητή δεδομένων ροής πραγματικού χρόνου απαιτεί υψηλό εύρος μετάδοσης, χαμηλές καθυστερήσεις και μικρές διαφοροποιήσεις (high bandwidth, low latency, low jitter). Ένα παράδειγμα δρομολόγησης που ελέγχει τον ρυθμό μετάδοσης είναι ο Rate-based αλγόριθμος [Kalm90]. Για την υλοποίησή του χρησιμοποιείται μια δεντρική δομή για την αποθήκευση πακέτων που περιμένουν την σειρά τους για μετάδοση.



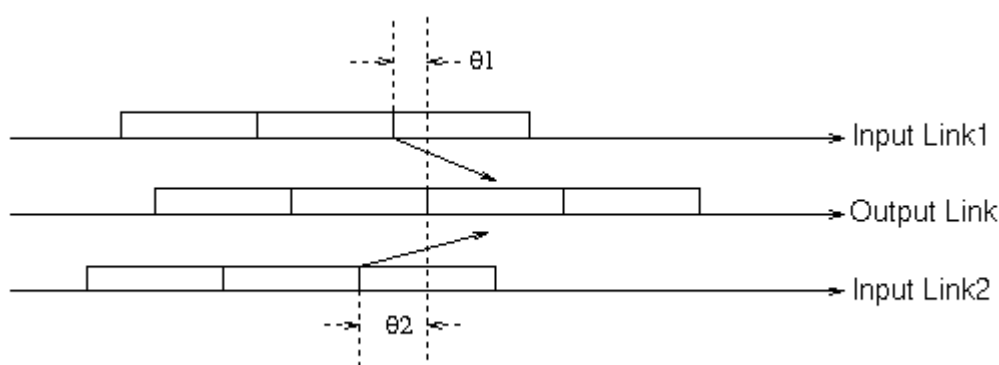
Σχήμα 7

Η μετάδοση λαμβάνει χώρα σε κύκλους σταθερού μήκους (π.χ. Σταθερό αριθμό μετάδοσης bytes). Τα πακέτα κάθε κύκλου βρίσκονται σε μια διασυνδεδεμένη λίστα και μόλις μεταδοθούν όλα ο δρομολογητής αρχίζει να εξυπηρετεί τον επόμενο κύκλο. Ο έλεγχος του ρυθμού επιτυγχάνεται με την κατάλληλη εισαγωγή πακέτων σε αυτή την δομή. Κάθε σύνδεση συσχετίζεται με τις παρακάτω παραμέτρους: το τμήμα εξυπηρέτησης, τον αριθμό τελευταίου κύκλου, τον τρέχοντα αριθμό πακέτου και το διάστημα. Το τμήμα εξυπηρέτησης αναπαριστά τον αριθμό των πακέτων που μπορούν να μεταδοθούν ανά κύκλο για κάθε σύνδεση, η δεύτερη παράμετρος κρατά τον αριθμό κύκλου στον οποίο προστέθηκαν πακέτα για τελευταία φορά, ο τρέχων αριθμός είναι ο αριθμός των πακέτων που έχουν προστεθεί στον τελευταίο κύκλο, και το διάστημα είναι ένας αριθμός που προστίθεται στον αριθμό τελευταίου κύκλου, όταν ο τρέχων αριθμός γίνει ίσος με το τμήμα εξυπηρέτησης. Τα πακέτα τοποθετούνται κάθε φορά στον κύκλο που καθορίζεται από την αντίστοιχη παράμετρο ανεξάρτητα αν υπάρχει χώρος σε προηγούμενους κύκλους. Με τον τρόπο αυτό επιτυγχάνεται έλεγχος του ρυθμού αποστολής πακέτων και περιορίζουμε το jitter. Μια άλλη

υλοποίηση της ίδιας ιδέας αποτελεί ο Hierarchical Round-robin HRR σχεδιασμένος για ATM δίκτυα.

Η δρομολόγηση Stop-and-Go ακολουθεί την αρχιτεκτονική που βασίζεται σε τμήματα εξυπηρέτησης. Ο αλγόριθμος ορίζει frames αναχώρησης και άφιξης για κάθε κόμβο στο δίκτυο. Σε κάθε διακόπτη το frame της άφιξης αντιστοιχεί στο frame αναχώρησης εισάγοντας μια καθυστέρηση θ . Σύμφωνα με την πολιτική αυτή η μετάδοση ενός πακέτου που έρχεται σε ένα κόμβο, κατά την διάρκεια ενός τμήματος f , καθυστερεί μέχρι την έναρξη του επόμενου frame (Σχήμα 8).

Για τον έλεγχο του ρυθμού επιτρέπεται μόνο ένα κλάσμα του χρόνου του frame να χρησιμοποιηθεί για μια σύνδεση. Η καθυστέρηση περιορίζεται από την μεταβλητή θ , ενώ ο μηχανισμός που εφαρμόζεται (framing) ομαλοποιεί το jitter στην έξοδο του διακόπτη.



Σχήμα 8: Stop-and-Go δρομολόγηση

2.6 Ανάπτυξη εξυπηρετητών δεδομένων ροής

Οι εξυπηρετητές αποθήκευσης βασίζονται σε δύο κυρίως αρχιτεκτονικές όσον αφορά δεδομένα ευαίσθητα στις καθυστερήσεις. Έτσι είτε επικεντρώνονται σε ένα σύστημα αρχείων (file system) είτε ακολουθούν το μοντέλο ροής δεδομένων. Στην πρώτη περίπτωση ο πελάτης αντιμετωπίζει τα αντικείμενα πολυμέσων σαν ένα μεγάλο αρχείο και εφαρμόζει τις λειτουργίες άνοιγμα κλείσιμο και διάβασμα αρχείου για να τα προσπελάσει. Η ανάκτηση γίνεται με περιοδικές αιτήσεις του πελάτη και αποτελεί το λεγόμενο μοντέλο έλξης (pull model). Διάφορες τεχνικές μπορούν να εφαρμοστούν σε ένα τέτοιο μοντέλο για καλύτερα αποτελέσματα, όπως για παράδειγμα η περιοδική προ-ανάκτηση δεδομένων στους προσωρινούς αποταμιευτές από τον δρομολογητή του εξυπηρετητή, η οποία μειώνει την απόκριση των λειτουργιών διαβάσματος. Αντίθετα στην δεύτερη αρχιτεκτονική ο εξυπηρετητής με τον κατάλληλο μηχανισμό δρομολόγησης αποστέλλει δεδομένα στον πελάτη περιοδικά και συνεχόμενα. Ο πελάτης στέλνει αιτήσεις της μορφής play, pause, resume και στην συνέχεια η όλη διαδικασία ανάκτησης ελέγχεται από τον εξυπηρετητή.

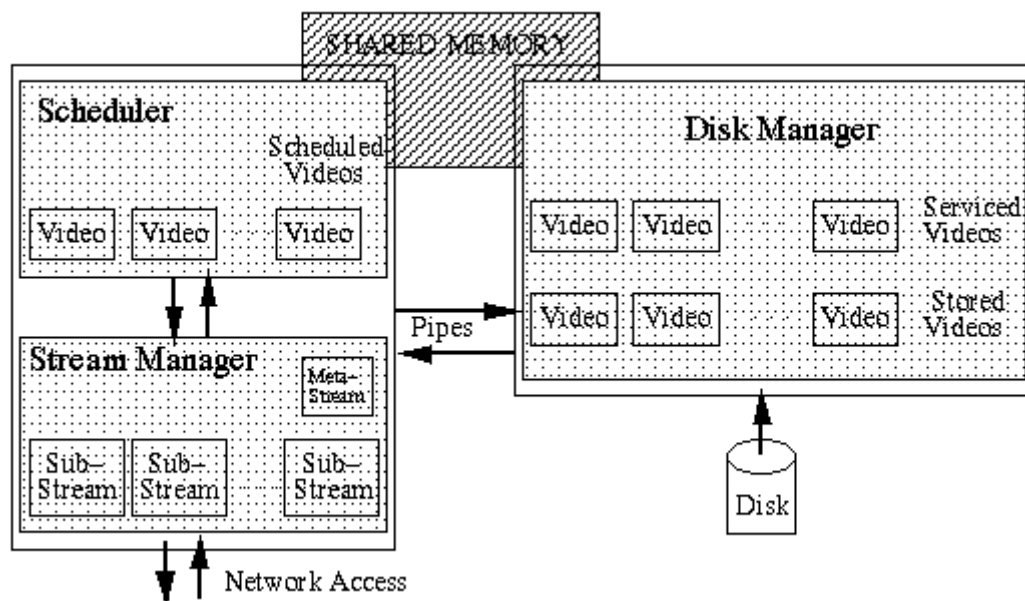
Κάθε αίτηση ανάκτησης από τους πελάτες σχετίζεται και με ένα ρυθμό μετάδοσης, ο οποίος είναι απαραίτητος για την ομαλή αναπαραγωγή. Όταν η αίτηση αφίχθαι στον

εξυπηρετητή χρησιμοποιείται ένας αλγόριθμος ελέγχου υποβολής αιτήσεων (admission control) που αποφασίζει αν θα κάνει δεκτή την καινούργια αίτηση. Ο έλεγχος αυτός βασίζεται στην διαθεσιμότητα των πόρων και τις απαιτήσεις σε πόρους της αίτησης. Κάθε αίτηση μπορεί να αναλυθεί σε πολλά στιγμιότυπα υπο-αιτήσεων που ζητούν την ανάκτηση τμημάτων δεδομένων του συνολικού αντικειμένου. Για την ικανοποίηση της αίτησης και των απαιτήσεων πραγματικού χρόνου που επιβάλλει είναι υπεύθυνος ο δρομολογητής του συστήματος, εφαρμόζοντας τον κατάλληλο συντονισμό στις υποβολές των υπο-αιτήσεων.

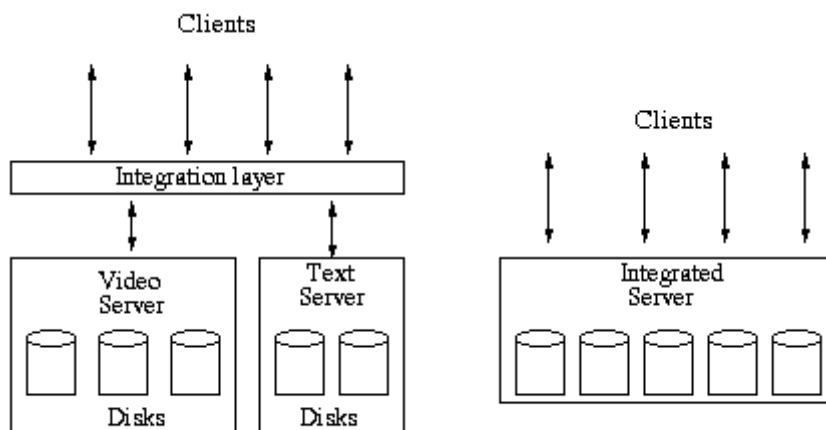
Ένα παράδειγμα υλοποίησης του μηχανισμού δρομολόγησης ανάκτησης μπορεί να γίνει με την δημιουργία μιας διεργασίας για κάθε αίτηση ανάκτησης [His94]. Η διεργασία αυτή προσπαθεί να υποβάλλει υπο-αιτήσεις ανάκτησης στο υποσύστημα αποθήκευσης με τέτοιο τρόπο ώστε να εξασφαλίζεται ο ρυθμός ροής δεδομένων που επιβάλλει η αίτηση. Έτσι περιοδικά υποβάλλονται υπο-αιτήσεις ανάκτησης δεδομένων από την διεργασία η οποία παραμένει αδρανής μέχρι την έναρξη της επόμενης περιόδου, όταν ενεργοποιείται από ένα σήμα (signal). Ο μηχανισμός αυτός δρομολόγησης ιδιαίτερα σε μεγάλο φόρτο εργασίας είναι φυσικό να εμφανίζει προβλήματα στην αναπαραγωγή των πολυμέσων, καθώς οι χρόνοι έναρξης και παύσης των διεργασιών όπως και η σειρά εκτέλεσης τους δεν μπορεί να καθοριστεί, καθιστώντας πιθανή την παραβίαση διάφορων προθεσμιών.

Μια άλλη προσέγγιση του μηχανισμού δρομολόγησης ανάκτησης [Ma97] περιλαμβάνει την ύπαρξη πολλών δρομολογητών που καλούνται α-δρομολογητές (r-schedulers). Οι α-δρομολογητές οργανώνουν τις υπο-αιτήσεις σε ομάδες που μοιράζονται την πρόσβαση στους δίσκους και διαχειρίζονται την λειτουργία των δίσκων. Οι υπο-αιτήσεις σε μια ομάδα α-δρομολογητή ανήκουν σε μια αίτηση. Οι α-δρομολογητές είναι τόσοι όσοι και οι διαθέσιμοι δίσκοι και οι υπο-αιτήσεις αναθέτονται στους δρομολογητές με τέτοιο τρόπο ώστε να ζητούν πρόσβαση στον ίδιο δίσκο σε κάθε περίοδο δρομολόγησης. Η προσέγγιση αυτή είναι καλύτερη από την προηγούμενη καθώς τμήμα του λογισμικού ασχολείται με την διαχείριση του κάθε δίσκου χωριστά οδηγώντας στην καλύτερη αξιοποίηση αυτών.

Στο [Bern95] περιγράφεται η ανάπτυξη ενός πρωτότυπου εξυπηρετητή δεδομένων video. Οι δύο κύριες εργασίες του εξυπηρετητή αποθήκευσης είναι η διαδικασία της ανάκτησης δεδομένων από τους δίσκους και η αποστολή τους μέσω δικτύου στους πελάτες. Αυτές οι εργασίες υλοποιούνται με δύο διεργασίες τον διαχειριστή δίσκου και τον διαχειριστή ροής, όπως φαίνονται στο Σχήμα 9. Σε αυτό το παράδειγμα ο διαχειριστής δίσκου εργάζεται κάτω από τον έλεγχο του διαχειριστή ροής. Και οι δύο διεργασίες επικοινωνούν μέσω διαμοιραζόμενης μνήμης για μεταφορά δεδομένων και μέσω μιας ουράς μηνυμάτων για την μετάδοση εντολών. Για κάθε νέα αίτηση ανάκτησης αντικειμένου ροής δεσμεύεται διπλάσιος χώρος του τμήματος του αντικειμένου στην διαμοιραζόμενη μνήμη. Με τη βοήθεια του δρομολογητή (ο οποίος λειτουργεί σε σταθερούς κύκλους και υπολογίζει το ποσό των δεδομένων που πρέπει να επεξεργαστεί για κάθε αντικείμενο ροής) ο διαχειριστής ροής αποστέλλει τα δεδομένα στους πελάτες χρησιμοποιώντας το κατάλληλο λογισμικό επαφής με το δίκτυο.



Σχήμα 9: Υλοποίηση πρωτότυπου εξυπηρετητή



Σχήμα 10: Εξυπηρετητές αποθήκευσης δεδομένων πολυμέσων

Δύο μεθοδολογίες υπάρχουν για τον σχεδιασμό ολοκληρωμένων εξυπηρετητών αποθήκευσης πολυμέσων. Η πρώτη μεθοδολογία περιλαμβάνει συστήματα που αποτελούνται από πολλαπλούς εξυπηρετητές κάθε ένας σχεδιασμένος για την βέλτιστη υποστήριξη συγκεκριμένων τύπων δεδομένων, ενώ η δεύτερη περιλαμβάνει συστήματα που υποστηρίζουν όλους τους τύπους δεδομένων σε ένα εξυπηρετητή όπως φαίνεται στο Σχήμα 10. Κάθε μεθοδολογία έχει τα δικά της πλεονεκτήματα και μειονεκτήματα.

2.7 Κλιμάκωση στον εξυπηρετητή αποθήκευσης

Η αυξημένη χρήση της πληροφορίας πολυμέσων είναι αποτέλεσμα της συνεχούς βελτίωσης του υλικού των υπολογιστών και της τεχνολογίας δικτύων. Στις μέρες μας

υπάρχει μεγάλη ανάγκη από νέες εφαρμογές όπως video κατά απαίτηση και απομακρυσμένη εκπαίδευση. Πολλά εμπορικά και ερευνητικά πρωτότυπα προϊόντα έχουν αναπτυχθεί με στόχο την εξυπηρέτηση πολλών πελατών με την μετάδοση ψηφιακού video μέσω δικτύου.

Η ανάπτυξη ενός υπολογιστικού συστήματος που θα μπορεί από μόνο του να αποθηκεύσει πολλά μέσα συνεχόμενης ροής και θα είναι ικανό να εξυπηρετεί ένα μεγάλο αριθμό πελατών είναι πολύ δύσκολη εργασία αν αναλογιστεί κανείς τις απαιτήσεις σε εύρος μετάδοσης, αποθηκευτικού χώρου και πραγματικού χρόνου που συνοδεύουν την πληροφορία πολυμέσων. Έτσι το πρόβλημα που εμφανίζεται είναι ότι εφόσον έχουμε συστήματα τα οποία μπορούν να διαχειριστούν ένα μικρό αριθμό από αντικείμενα ροής και να εξυπηρετήσουν μερικούς πελάτες, πώς μπορούμε να τοποθετήσουμε νέους πόρους ώστε κλιμακώνοντας το σύστημα να το καταστήσουμε ικανό να διαχειριστεί μεγαλύτερο ποσό δεδομένων και αριθμό πελατών.

Η κλιμάκωση στην πληροφορική σημαίνει ότι όχι μόνο όλα τα τμήματα του υπολογιστικού συστήματος (υλικό και λογισμικό) συνεχίζουν να λειτουργούν σωστά καθώς γίνεται η κλιμάκωση, αλλά συγχρόνως εκμεταλλεύονται πλήρως αυτήν την ανάπτυξη. Έτσι ένα σύστημα εξυπηρέτησης πολυμέσων με δυνατότητα κλιμάκωσης μπορεί να διαχειριστεί μια αύξηση στον αριθμό των χρηστών με την κατάλληλη προσθήκη πόρων χωρίς να εμφανίζονται προβλήματα και φαινόμενα συμφόρησης. Το σύστημα πρέπει να αντιμετωπίζει την αύξηση σε απαιτήσεις χωρίς αλλαγές στην δομή του και στους εφαρμοζόμενους αλγορίθμους [Kor95].

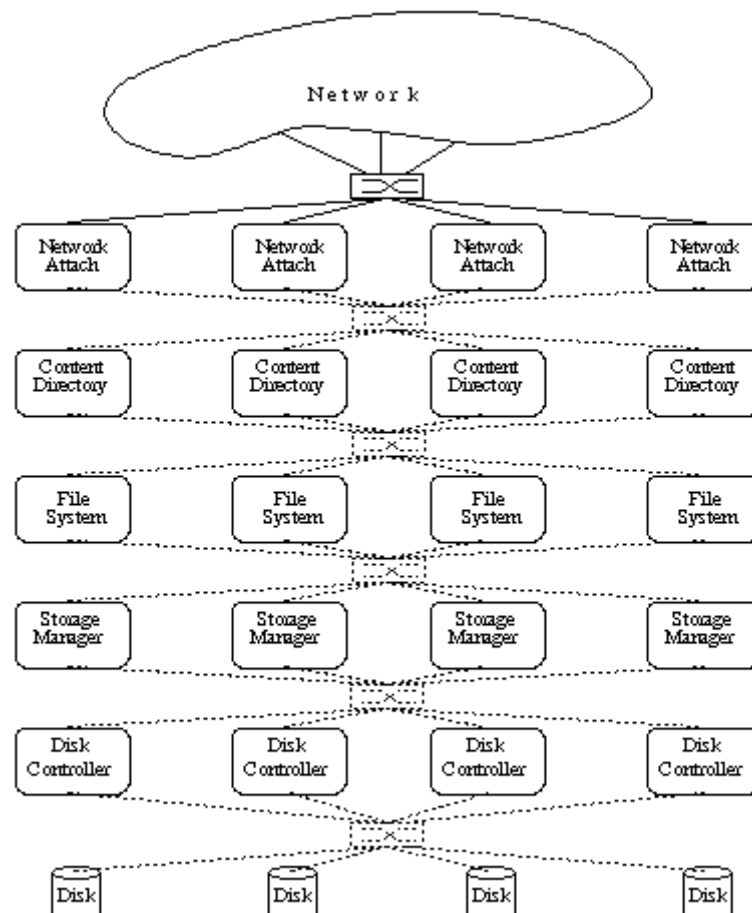
Οι πιο σημαντικές παράμετροι που εμφανίζουν προβλήματα στην κλιμάκωση ενός συστήματος είναι:

- Η αύξηση της βάσης των χρηστών (οδηγεί σε προβλήματα στα υποσυστήματα αποθήκευσης και δικτύου)
- Το μέγεθος δεδομένων των μέσων (έχει επιπτώσεις στο δίκτυο και στην διαθεσιμότητα αποθηκευτικού χώρου)
- Η μη ομοιόμορφη κατανομή των αιτήσεων για ένα χρονικό διάστημα στους διαθέσιμους εξυπηρετητές.

Οι εξυπηρετητές πολυμέσων και ειδικά τα συστήματα λογισμικού πρέπει να αναπτύσσονται με τέτοιο τρόπο ώστε να είναι ανεξάρτητα από περιορισμούς στους πόρους και ευέλικτα για να εγγραφούν την κλιμάκωση, διατηρώντας τον χρόνο απόκρισης, την διαθεσιμότητα και την αξιοπιστία σε ικανοποιητικά επίπεδα.

Δύο είναι οι πιο συχνά χρησιμοποιούμενες αρχιτεκτονικές για την ανάπτυξη εξυπηρετητών με δυνατότητα κλιμάκωσης. Στην μία γίνεται χρήση ομάδων από υπολογιστές που υποστηρίζουν το πληροφορικό σύστημα του εξυπηρετητή, ενώ η δεύτερη αρχιτεκτονική περιλαμβάνει την ανάπτυξη συστημάτων μαζικών παράλληλων μικροεπεξεργασιών όπως ο SB-PRAM , ο Mitra κτλ. .

Μια προσπάθεια παρουσίασης της οργάνωσης κλιμακωτών εξυπηρετητών video με την περιγραφή εναλλακτικών αρχιτεκτονικών και εμπορικών προτάσεων γίνεται στο [Tetz95]. Με δεδομένες τις απαιτήσεις για οικονομική πρόσβαση πληροφορίας, μεγάλο αριθμό χρηστών και μεγάλο μέγεθος αντικειμένων, καταλήγουν σε μια αρχιτεκτονική με πολλά υποσυστήματα- αντίγραφα διασυνδεδεμένα με διακόπτες. Τα υποσυστήματα αυτά ακολουθούν μια ιεραρχία και διακρίνονται στα: υποσύστημα δικτύου, διαχειριστή περιεχομένου, σύστημα αρχείων, σύστημα αποθήκευσης, διαχειριστής δίσκου και τέλος ο φυσικός δίσκος.



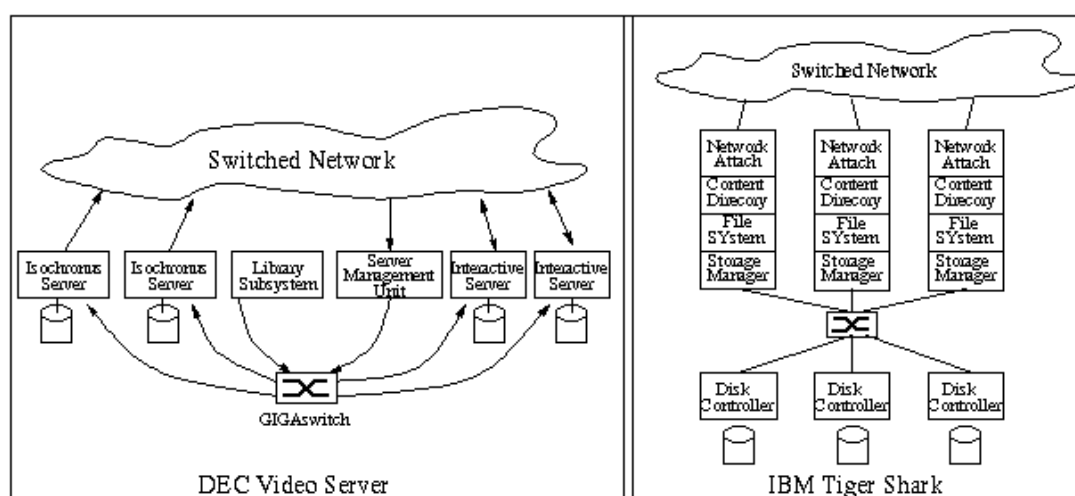
Σχήμα 11: Υποσυστήματα και διακόπτες

Σημαντικό στοιχείο στον σχεδιασμό του κλιμακωτού συστήματος είναι η τοποθέτηση του κύριου διακόπτη που θα ενώνει τα υποσυστήματα. Στο Σχήμα 11 διακρίνονται τα υποσυστήματα και οι πιθανές θέσεις διασύνδεσής τους με διακόπτες. Τα υποσυστήματα που βρίσκονται μεταξύ των δίσκων και του κύριου διακόπτη λειτουργούν παράλληλα, αλλά έχουν την εντύπωση ότι ο αποθηκευτικός χώρος είναι τμηματοποιημένος (partitioned). Το χαμηλότερο επίπεδο που μπορεί να εφαρμοστεί διακόπτης είναι οι δίσκοι. Σε αυτή την περίπτωση όλα τα υποσυστήματα εργάζονται ταυτόχρονα και μπορούν να έχουν πρόσβαση σε όλους τους δίσκους. Όμως αυτή η εφαρμογή είναι ανέφικτη κυρίως λόγω έλλειψης ανάλογου υλικού στους συνηθισμένους δίσκους. Ανεβάζοντας τον διακόπτη στο επίπεδο

του διαχειριστή δίσκου έχουμε τα ίδια αποτελέσματα με την προηγούμενη περίπτωση και το πλεονέκτημα ότι τώρα μπορούμε να χρησιμοποιήσουμε συνηθισμένους δίσκους. Σημαντική διαφοροποίηση έχουμε με την τοποθέτηση διακόπτη στο επίπεδο του υποσυστήματος αποθήκευσης. Κάθε διαχειριστής αποθήκευσης έχει πλέον πρόσβαση μόνο στους δίσκους που είναι άμεσα συνδεδεμένοι με αυτόν. Το γεγονός αυτό απλοποιεί την λειτουργία του, αλλά περιορίζει τον αριθμό των δίσκων που είναι διαθέσιμοι για το ίδιο περιεχόμενο. Έτσι για την αντιμετώπιση της αυξημένης ζήτησης διαφόρων μέσων απαιτείται η χρήση αντιγράφων στους υπόλοιπους κόμβους. Ανάλογα είναι τα αποτελέσματα της τοποθέτησης διακόπτη που θα συνδέει εξωτερικά του διαθέσιμους κόμβους και τμηματοποιεί τον αποθηκευτικό χώρο. Για τον λόγο αυτό εφαρμόζονται διάφορες τεχνικές όπως θα δούμε στην συνέχεια. Σημαντικό πλεονέκτημα είναι ότι έτσι αποφεύγουμε την χρήση εσωτερικών διακοπών και συνεπώς μειώνουμε το κόστος του συστήματος. Από την άλλη πλευρά όμως παρατηρείται αύξηση του επικοινωνιακού κόστους μεταξύ των κόμβων.

Ένα άλλο ζήτημα στα συστήματα εξωτερικής διασύνδεσης είναι η δυνατότητα κλιμάκωσης του διαχειριστή περιεχομένου. Η αντιμετώπιση γίνεται κυρίως με δύο τρόπους. Ο ένας τρόπος είναι ο πελάτης να στέλνει μια αίτηση ανάκτησης και ένα κεντρικό υποσύστημα περιεχομένου να απαντά υποδεικνύοντας τον κόμβο που περιέχει την πληροφορία. Στην συνέχεια ο πελάτης ξαναστέλνει την αίτηση στο σωστό κόμβο. Σύμφωνα με τον δεύτερο τρόπο το κεντρικό υποσύστημα περιεχομένου κατευθύνει μόνο του την αίτηση στον σωστό κόμβο, αναθέτοντάς του την εξυπηρέτηση της αίτησης.

Αν συμβολίσουμε τη ιεραρχία των υποσυστημάτων με τα αρχικά τους (NSFSDD) και την χρήση διακόπτη με * τότε μπορούμε να συμβολίσουμε τα χαρακτηριστικά διαφόρων συστημάτων. Συστήματα που έχουν αναπτυχθεί ή βρίσκονται σε ανάπτυξη είναι κυρίως *NCFSD (με εξωτερικό διακόπτη) ή *NCFSD (με διακόπτη στο επίπεδο του διαχειριστή δίσκων). Στο Σχήμα 12 υπάρχουν δύο εμπορικά συστήματα εξυπηρετητών video. Ο DEC video server είναι της μορφής *NCFSD. Στο ίδιο σχήμα απεικονίζεται και ο IBM Tiger Shark που ακολουθεί το σχήμα *NCFSD.



Σχήμα 12 : Παραδείγματα εμπορικών εξυπηρετητών

Οι αρχιτεκτονικές που ακολουθούν το σχήμα των ομάδων υπολογιστών (cluster of workstations) έχουν πλεονεκτήματα ως προς την κλιμάκωση του συστήματος καθώς όταν ο φόρτος αυξάνεται είναι εύκολη η προσθήκη και άλλων υπολογιστών. Επίσης σημαντικό πλεονέκτημα αναδεικνύεται η δυνατότητα χρήσης συνηθισμένων και οικονομικών συστημάτων υλικού αλλά και λογισμικού αντίθετα με τους παράλληλους υπολογιστές.

Όσον αφορά το ζήτημα της κλιμάκωσης σε ένα εξυπηρετητή αποθήκευσης δεδομένων ροής, ο εξυπηρετητής πρέπει να είναι ικανός να κλιμακώνεται τοπικά ώστε να αντιμετωπίζει τον αυξημένο φόρτο εργασίας, ενώ επίσης ένα διανεμημένο σχήμα από πολλούς εξυπηρετητές πρέπει να είναι εύκολα επεκτάσιμο πάνω στο δίκτυο. Αρχιτεκτονικές που υποστηρίζουν ιεραρχίες στα επίπεδα αποθήκευσης κάνοντας χρήση της τριτεύουσας μνήμης, μπορούν να θεωρηθούν κλιμακούμενα μόνο ως προς την χωρητικότητα δεδομένων.

Η σύγχρονη τεχνολογία των δίσκων μπορεί να υποστηρίξει την αναπαραγωγή μόνο μερικών αντικειμένων ροής. Χρησιμοποιώντας την τεχνολογία RAID είναι πιθανόν να αυξήσουμε το εύρος μετάδοσης με άμεση απόρροια την δυνατότητα υποστήριξης περισσότερων αντικειμένων ροής. Παρά το γεγονός αυτό, τα προβλήματα συμφόρησης δεν πρόκειται να εξαλειφθούν λόγω της έλλειψης δυνατότητας κλιμάκωσης των συστημάτων RAID. Για να επιτύχουμε την κλιμάκωση του αριθμού των πελατών σαν συνάρτηση των πρόσθετων δίσκων, ο εξυπηρετητής μπορεί να εφαρμόσει τεχνικές striping [Ghan97]. Αν και η χρήση συστημάτων RAID και η προσθήκη νέων δίσκων μπορούν να κλιμακώσουν ένα τοπικό εξυπηρετητή δεν αποτελούν λύση ευρείας κλίμακας. Αυτή η διαδικασία σύντομα θα οδηγήσει σε προβλήματα απόδοσης καθώς το αυξημένο προσθετικό εύρος των δίσκων θα υπερβεί τις δυνατότητες του υποσυστήματος δικτύου ή του κεντρικού επεξεργαστή. Η λύση σε αυτά τα προβλήματα είναι η κατανομή ενός αριθμού από εξυπηρετητές στο δίκτυο με απώτερο στόχο να διαμοιραστούν τον φόρτο εργασίας των αιτήσεων των χρηστών πολλά συστήματα αντί για ένα. Ένα κλιμακωτό σύστημα πρέπει να είναι πάντα διαθέσιμο και αυξητικά επεκτάσιμο [Lee95].

Μια πρόταση για την επίτευξη της κλιμάκωσης περιλαμβάνει τον σχεδιασμό της αρχιτεκτονικής βασισμένης στην δημιουργία διανεμημένων αντιγράφων στιγμιότυπων εξυπηρετητών που θα περικλείουν τεχνικές static redundancy και striping ώστε να εξασφαλίσουν ισορροπία στον φόρτο εργασίας (load balancing) [Loug94] [Peg95].

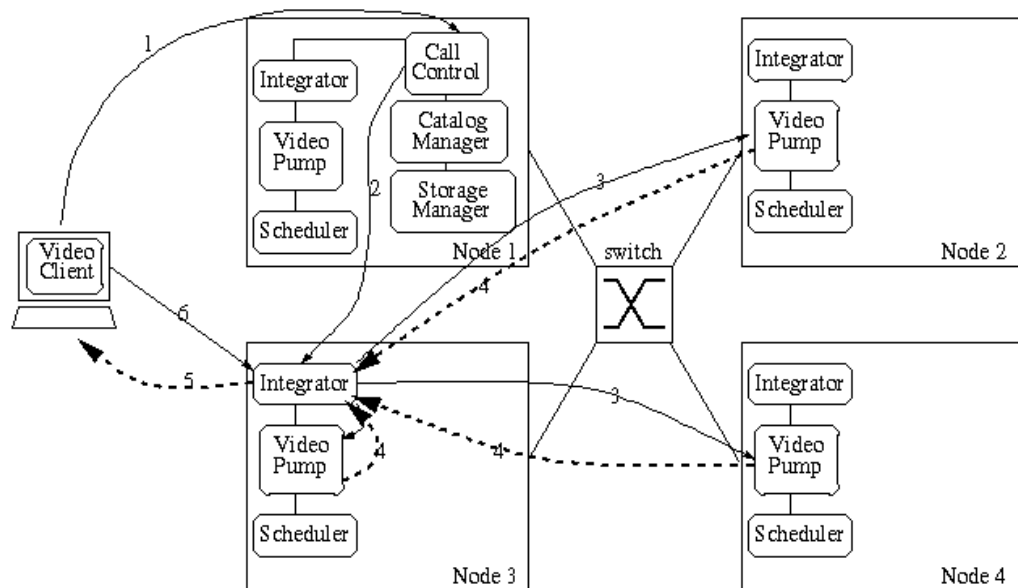
Σύμφωνα με την στατική μέθοδο αντιγραφής, πολλαπλά αντίγραφα ενός μέσου υπάρχουν στο σύνολο των εξυπηρετητών, ένα για κάθε στιγμιότυπο εξυπηρετητή. Ακολουθώντας αυτή την μέθοδο δύο σχήματα μπορούν να εφαρμοστούν. Είτε κάθε εξυπηρετητής θα μπορεί να δέχεται αιτήσεις από ένα προ-καθορισμένο σύνολο πελατών ανάλογα με τις δυνατότητες του εξυπηρετητή και το εύρος του δικτύου, είτε όλοι οι εξυπηρετητές θα μπορούν να προσπελαστούν από όλους τους πελάτες αλλά κάθε φορά οι αιτήσεις των χρηστών θα κατευθύνονται προς εκείνο το στιγμιότυπο με το μικρότερο ποσό εργασίας. Στην τελευταία περίπτωση παράγοντες όπως φόρτος εξυπηρετητή, κυκλοφορία δικτύου και συσκευές που συνθέτουν το δίκτυο (π.χ. ATM διακόπτες), είναι σημαντικοί στην διαδικασία επιλογής του εξυπηρετητή που πρέπει μια νεοεισαχθείσα αίτηση να

προσπελάσει. Σημαντικό πλεονέκτημα της στατικής αντιγραφής είναι ότι προκύπτει ένα σύστημα με υψηλή ανοχή στις βλάβες (fault-tolerance), εφόσον το σύστημα συνεχίζει να λειτουργεί σωστά και στην περίπτωση παύσης λειτουργίας ενός στιγμιότυπου εξυπηρετητή. Το βασικό μειονέκτημα της τεχνικής αυτής προκύπτει από το γεγονός ότι ενώ επιτυγχάνεται η κλιμάκωση του συστήματος, ως προς το συνολικό εύρος ανάκτησης, ωστόσο η χωρητικότητα ολόκληρου του συστήματος περιορίζεται στην χωρητικότητα ενός μόνο εξυπηρετητή. Επίσης ειδική μεταχείριση χρειάζονται οι λειτουργίες ενημέρωσης των μέσων ώστε να παραμένουν τα αντίγραφα πανομοιότυπα.

Εφαρμόζοντας την εναλλακτική τεχνική striping σε επίπεδο δικτύου τα δεδομένα ενός μέσου διαιρούνται σε τμήματα και κατανέμονται σε ένα αριθμό από στιγμιότυπα εξυπηρετητές. Οι απαιτήσεις πραγματικού χρόνου που επιβάλλονται από την αναπαραγωγή του μέσου προϋποθέτει την στενή συνεργασία μεταξύ των εξυπηρετητών ώστε να ικανοποιούνται οι προθεσμίες κατά την ανάκτηση. Ακολουθώντας αυτή την τεχνική επιτυγχάνουμε κλιμάκωση του αριθμού των πελατών που μπορούν να εξυπηρετηθούν σε συνάρτηση με τον αριθμό των διαθέσιμων στιγμιότυπων εξυπηρετητών στο δίκτυο.

Στην βιβλιογραφία υπάρχουν αρκετοί αλγόριθμοι για την τμηματοποίηση δεδομένων πολυμέσων σε ένα σύνολο από δίσκους που μπορούν να εφαρμοστούν σε επίπεδο εξυπηρετητών στο δίκτυο. Για παράδειγμα η τεχνική round-robin διαμοιράζει τα τμήματα δεδομένων σε διαφορετικούς εξυπηρετητές δικτύου αποθηκεύοντας ένα τμήμα σε κάθε στιγμιότυπο κάθε φορά, επαναλαμβάνοντας την διαδικασία κυκλικά. Σημαντικό μειονέκτημα της μεθόδου αυτής είναι η μείωση της διαθεσιμότητας, καθώς τυχών βλάβη σε ένα κόμβο οδηγεί αναγκαστικά σε έλλειψη δεδομένων. Για την αντιμετώπιση συνεπώς του προβλήματος της ανοχής στις βλάβες μπορούν να εφαρμοστούν τεχνικές που συνδυάζουν την striping τεχνική με την μέθοδο των αντιγράφων, όπως οι mirrored-striping , chained declustering, multi-chained declustering.

Παράδειγμα εφαρμογής της αρχιτεκτονικής ομάδων υπολογιστών είναι ο Elvira εξυπηρετητής [Sand96]. Ο Elvira συγκροτείται από ένα αριθμό υπολογιστών οι οποίοι είναι διασυνδεδεμένοι μέσω ενός διακόπτη ATM. Το ATM δίκτυο χρησιμεύει για την ενδοεπικοινωνία μεταξύ των υπολογιστών αλλά και για την μετάδοση των δεδομένων των μέσων. Η αρχιτεκτονική του συστήματος φαίνεται στο Σχήμα 13.

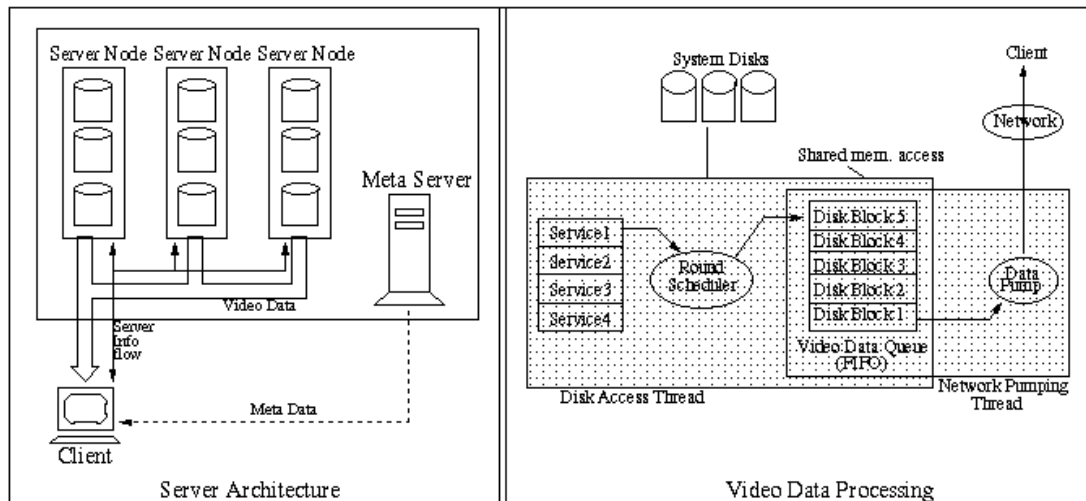


Σχήμα 13: Αρχιτεκτονική του εξυπηρετητή Elvira

Ένας κόμβος είναι υπεύθυνος να δέχεται τις αιτήσεις των πελατών και τρέχει την αντίστοιχη διεργασία (call control process). Η διεργασία αυτή με την βοήθεια του διαχειριστή καταλόγου αποφασίζει αν η αίτηση μπορεί να ικανοποιηθεί. Όλοι οι υπόλοιποι κόμβοι έχουν τα ίδια τμήματα λογισμικού τον ολοκληρωτή, την αντλία video, και τον δρομολογητή (integrator, video pump, scheduler). Η διεργασία υποδοχής των αιτήσεων αποφασίζει επίσης και ποιος κόμβος θα είναι υπεύθυνος για την μετάδοση του μέσου. Την εργασία αυτή μαζί με την αλληλεπίδραση με τον χρήστη την αναλαμβάνει ο ολοκληρωτής στον κόμβο. Η αντλία είναι υπεύθυνη να υποβάλει αιτήσεις ανάκτησης δεδομένων στους δίσκους και να ορίζει τους χρονικούς περιορισμούς. Αν το μέσο έχει αποθηκευτεί σε περισσότερους από ένα κόμβους τότε οι αντλίες αυτών των κόμβων αναλαμβάνουν την ανάκτηση δεδομένων και την μεταφορά τους στον ολοκληρωτή. Ο δρομολογητής γνωρίζοντας τους χρονικούς περιορισμούς προσπαθεί να βελτιστοποιήσει την πρόσβαση στους δίσκους. Σημαντικό μειονέκτημα του Elvira εξυπηρετητή είναι το μεγάλο επικοινωνιακό κόστος που επιφέρει η κλιμάκωση του συστήματος, καθώς το τοπικό δίκτυο εκτός από την ενδοεπικοινωνία των κόμβων επιβαρύνεται και με την μεταφορά δεδομένων των μέσων μεταξύ των κόμβων σε περιπτώσεις διασποράς της αποθήκευσής τους σε διαφορετικούς κόμβους.

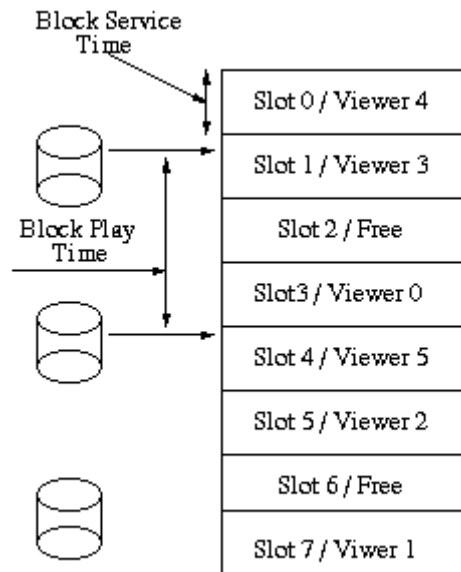
Ένα άλλο πρωτότυπο σύστημα εξυπηρετητή με δυνατότητα επέκτασης και υποστήριξης πολλών πελατών περιγράφεται στο [Gafs98]. Όπως φαίνεται στο Σχήμα 14 η δυνατότητα κλιμάκωσης του προτεινόμενου συστήματος οφείλεται στην υιοθέτηση της τεχνικής χρήσης πολλών στιγμιότυπων διανεμημένων εξυπηρετητών. Η διαχείριση της πληροφορίας που συνοδεύει τα αντικείμενα ροής διαχειρίζονται χωριστά από ένα ειδικό μέτα-εξυπηρετητή. Η επικοινωνία μεταξύ των κόμβων καθώς και το κανάλι αλληλεπίδρασης με το χρήστη στηρίζονται στο πρωτόκολλο TCP, ενώ η μετάδοση των δεδομένων των μέσων γίνεται με χρήση UDP. Τα δεδομένα των μέσων αποθηκεύονται στους κόμβους χρησιμοποιώντας τεχνικές striping και η μετάδοση των δεδομένων λαμβάνει χώρα από

όλους τους εμπλεκόμενους κόμβους-εξυπηρετητές. Ο συντονισμός των κόμβων για την σωστή μετάδοση των δεδομένων είναι υπευθυνότητα του πελάτη, ο οποίος ζήτησε την ανάκτηση, χρησιμοποιώντας το κανάλι ροής πληροφορίας του εξυπηρετητή όπως απεικονίζεται στο σχήμα. Στην δεξιά πλευρά του σχήματος παρουσιάζεται γραφικά η αρχιτεκτονική ενός κόμβου του συστήματος που ακολουθεί την συνήθη αντιμετώπιση διεργασιών ανάκτησης και μετάδοσης δεδομένων. Θετικά σημεία του παραδείγματος αυτού είναι η δυνατότητα εφαρμογής του εξυπηρετητή σε ετερογενές περιβάλλον εξυπηρετητών και η αξιοπιστία.



Σχήμα 14 Παράδειγμα αρχιτεκτονικής επεκτάσιμου εξυπηρετητή

Στο [Bol97] παρουσιάζεται το σύστημα Tiger που στηρίζει το Microsoft Netshow Professional Video server. Ο σχεδιασμός του Tiger έχει σαν στόχο την κλιμάκωση του εξυπηρετητή χρησιμοποιώντας ένα αριθμό από μηχανήματα διασυνδεδεμένα σε ATM δίκτυο. Τα δεδομένα αποθηκεύονται στο σύστημα ακολουθώντας την τεχνική striping σε όλα τα μηχανήματα και σε όλους τους δίσκους σε κάθε κόμβο. Προκειμένου ο εξυπηρετητής να ικανοποιήσει τις απαιτήσεις πραγματικού χρόνου των αιτήσεων και να επιτευχθεί ο συντονισμός λειτουργίας διαφορετικών κόμβων, διατηρεί ένα δρομολόγιο των διεργασιών παρουσίασης που αναπαράγουν τα μέσα. Το μέγεθος του δρομολογίου είναι ανάλογο του συστήματος και του φόρτου εργασίας, συνεπώς μια κεντρική διαχείριση των δρομολογίων θα αποτελούσε εμπόδιο στην κλιμάκωση του συστήματος. Έτσι επιχειρείται μια διανεμημένη διαχείριση σύμφωνα με την οποία κάθε κόμβος-υπολογιστής έχει μια μη ενημερωμένη (out-of-date) μερική άποψη (view) του δρομολογίου. Κάθε κόμβος ενεργοποιείται και στέλνει τα απαραίτητα δεδομένα στις διεργασίες παρουσίασης βασισμένος στην δική του άποψη του δρομολογίου. Το μοντέλο που παρουσιάζεται χαρακτηρίζεται ως «μαζική ψευδαίσθηση» (coherent hallucination), καθώς δεν υπάρχει ένα κεντρικό ολοκληρωμένο δρομολόγιο αλλά κάθε κόμβος έχει την ψευδαίσθηση ότι «βλέπει» ένα τμήμα του κεντρικού δρομολογίου. Για λόγους υλοποίησης το σύστημα Tiger περιορίζεται στην υπόθεση ότι όλα τα αντικείμενα ροής έχουν την ίδια σταθερή απαίτηση στον ρυθμό ανάκτησης. Το δρομολόγιο του συστήματος έχει την μορφή του σχήματος (Σχήμα 15).



Σχήμα 15: Δρομολόγιο Δίσκου του εξυπηρετητή Tiger

Το δρομολόγιο αποτελεί ένα πίνακα με θέσεις που αντιστοιχούν σε κάθε διεργασία παρουσίασης. Κάθε θέση μπορεί να αντιστοιχηθεί στο χρονικό διάστημα εξυπηρέτησης τμήματος (χρόνος παρουσίασης τμήματος / μέγιστος αριθμός αντικειμένων ανά δίσκο), ενώ το μέγεθος ολόκληρου του δρομολογίου προκύπτει από τον χρόνο παρουσίασης τμήματος επί των αριθμό των δίσκων του συστήματος. Κάθε υπολογιστής-κόμβος κρατάει ένα δείκτη στο δρομολόγιο για κάθε δίσκο που διαχειρίζεται. Αυτοί οι δείκτες μετακινούνται στο δρομολόγιο σε πραγματικό χρόνο και κάθε φορά που βρίσκονται στην αρχή μιας νέας θέσης ο κόμβος αρχίζει να στέλνει δεδομένα για την αντίστοιχη διεργασία παρουσίασης.

Για την κλιμάκωση του συστήματος προτείνονται δύο τρόποι. Ο πρώτος υποδεικνύει ότι κάθε κόμβος πρέπει να έχει ένα αντίγραφο ολόκληρου του δρομολογίου που θα είναι στο μεγαλύτερο μέρος του μη ενημερωμένο και ο δεύτερος τρόπος επιτρέπει κάθε κόμβο να παρακολουθεί μόνο ένα μέρος του δρομολογίου που βρίσκεται κοντά σε αυτό που επεξεργάζονται οι δίσκοι του, ενώ ταυτόχρονα κάθε κόμβος πρέπει να διαδίδει την πληροφορία του δρομολογίου κυκλικά σε όλους του κόμβους του συστήματος με τον ίδιο ρυθμό που οι κόμβοι κινούνται μέσα στο δρομολόγιο. Η πρώτη λύση δεν θεωρήθηκε ικανοποιητική για την υλοποίηση του Tiger και ακολουθήθηκε η δεύτερη τεχνική. Έτσι κάθε κόμβος έχει μια μερική άποψη του δρομολογίου δηλαδή παρακολουθεί τις θέσεις που περιγράφουν τις εργασίες, που οι δίσκοι του θα αντιμετωπίσουν σε λίγα δευτερόλεπτα στο μέλλον. Κατά διαστήματα ο κόμβος μεταδίδει τα περιεχόμενα του δρομολογίου στον επόμενο κόμβο ενημερώνοντας για την κατάσταση επεξεργασίας των αιτήσεων. Με αυτό τον τρόπο διαδίδεται η πληροφορία συνεχώς σε όλους τους κόμβους συντονίζοντας την λειτουργία τους. Ένα από τα μειονεκτήματα της τεχνικής αυτής είναι προφανώς το επικοινωνιακό κόστος μεταξύ των κόμβων του συστήματος.

➤ Σχεδιασμός του εξυπηρετητή αποθήκευσης

3

Σε αυτό το κεφάλαιο γίνεται μια περιγραφή του πληροφορικού συστήματος πολυμέσων Kydonia και της αρχιτεκτονικής του. Παρουσιάζεται σε μεγαλύτερη λεπτομέρεια ο εξυπηρετητής αποθήκευσης και ο μηχανισμός δρομολόγησης των αιτήσεων. Γίνεται επίσης μια τεχνική περιγραφή του συστήματος σε επίπεδο λογισμικού και με την βοήθεια πειραματικών εφαρμογών αναδεικνύονται μερικά προβλήματα που η έρευνα και οι εργασίες στα πλαίσια της παρούσας διατριβής έρχονται να καλύψουν. Στο τέλος του κεφαλαίου περιγράφονται σε γενικές γραμμές τα τρέχοντα χαρακτηριστικά του συστήματος και ιδιαίτερα του τμήματος του λογισμικού που αναλαμβάνει την διαχείριση των χρονικά ευαίσθητων δεδομένων.

3.1 Ο Εξυπηρετητής Πολυμέσων Kydonia

Το πρόγραμμα Kydonia άρχισε στο εργαστήριο διανεμημένων πληροφοριακών συστημάτων με απώτερο σκοπό να διερευνήσει θέματα σχεδιασμού και να αναπτύξει την τεχνολογία για τα πληροφορικά συστήματα πολυμέσων του μέλλοντος. Τα κύρια χαρακτηριστικά του συστήματος είναι: η διαχείριση αποθήκευσης των αντικειμένων πολυμέσων, μηχανισμοί δρομολόγησης, μετάδοση δεδομένων σε ετερογενή δίκτυα, λειτουργικότητα βάσεων δεδομένων πολυμέσων, μοντελοποίηση πολυμέσων, μέθοδοι προσπέλασης κειμένου και video και τεχνικές πλοήγησης. Αντικειμενικός στόχος του συστήματος ήταν να αποτελέσει μια πρότυπη πλατφόρμα έρευνας και ανάπτυξης πληροφοριακών συστημάτων πολυμέσων και να υποστηρίξει διανεμημένες εφαρμογές όπως ταινίες κατ' απαίτηση και ψηφιακές βιβλιοθήκες.

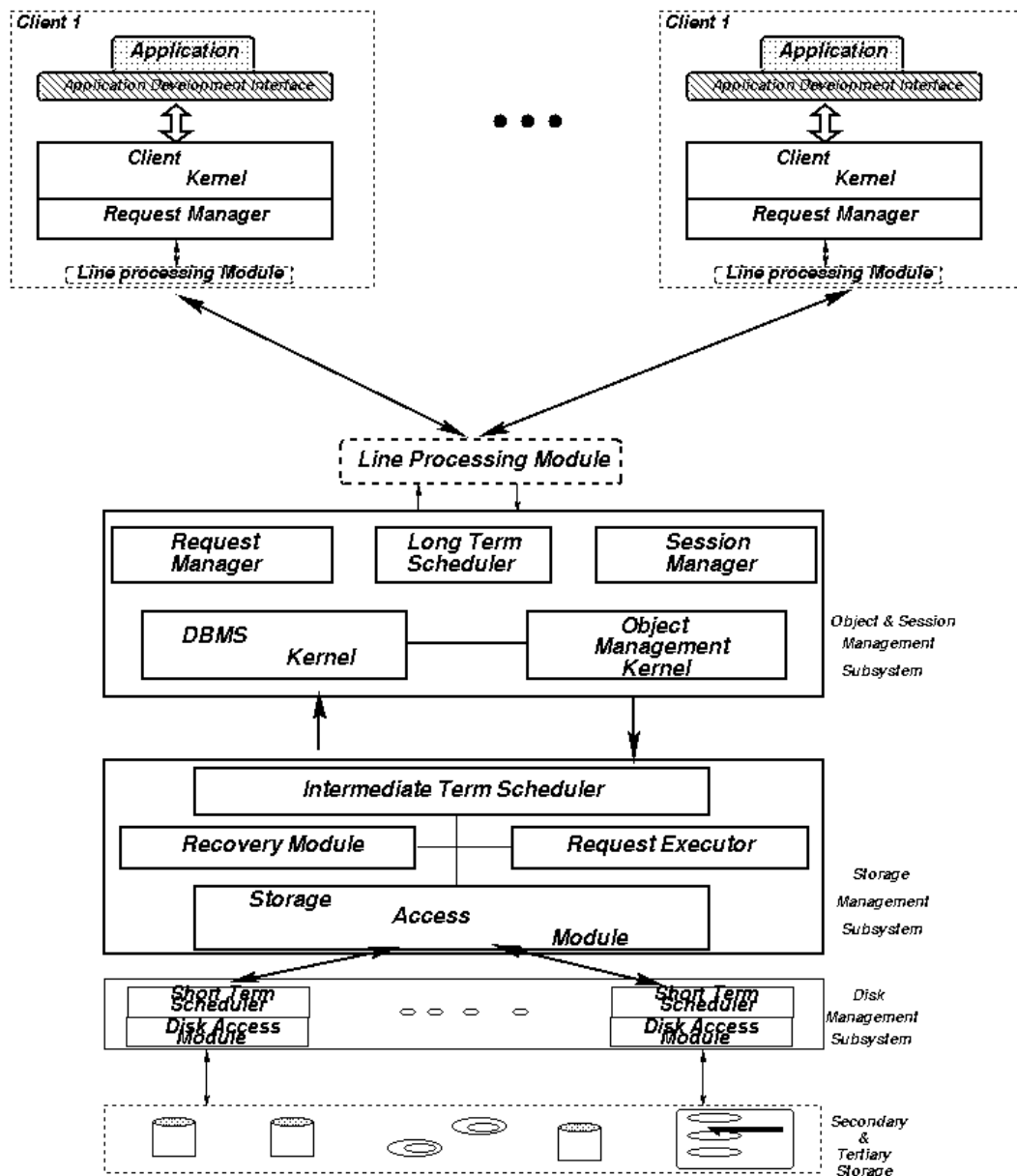
Η ανάπτυξη του συστήματος είναι βασισμένη στην έρευνα στις εξής περιοχές: εξυπηρετητές αποθήκευσης και ιεραρχίες αποθήκευσης, δρομολόγηση πραγματικού χρόνου της προσπέλασης δεδομένων, μετάδοση χρονικά ευαίσθητων δεδομένων, τεχνικές αναπαράστασης πολυμέσων, ανάκτηση βάσει περιεχομένου και πλοήγηση. Αρκετές εφαρμογές έχουν αναπτυχθεί έτσι ώστε να εμφανιστούν περισσότερο λεπτομερής απαιτήσεις για το σύστημα και για λόγους πειραματισμών. Σημαντικό είναι επίσης και το γεγονός της ολοκλήρωσης του συστήματος με πρότυπα στην τεχνολογία αιχμής, όπως MPEG-2 συμπίεση για video και ήχο και HTML για την αναπαράσταση εγγράφων σε συνδυασμό με την απαραίτητη λειτουργικότητα για την διαχείρισή τους.

Οι πιο πρόσφατες κατευθύνσεις ανάπτυξης αφορούν μια αρχιτεκτονική συμβατή με το πρότυπο DAVIC, το οποίο εμπεριέχει την χρήση MPEG-2 transport streams για την μετάδοση χρονικά ευαίσθητων δεδομένων πάνω από ATM δίκτυα, όπως επίσης την

ολοκλήρωση προτύπων σαν το DSM-CC και CORBA, τα οποία παρέχουν διαλειτουργικότητα μεταξύ διανεμημένων συστημάτων.

3.2 Η Αρχιτεκτονική του συστήματος Kydonia

Το σύστημα ακολουθεί το σχήμα πελάτη-εξυπηρετητή και αποτελείται από τρία βασικά κομμάτια όπως εμφανίζονται στο Σχήμα 16 και περιγράφονται στην συνέχεια.



Σχήμα 16: Η αρχιτεκτονική του πληροφορικού συστήματος Kydonia

3.2.1 Το υποσύστημα υποστήριξης του πελάτη

Το υποσύστημα αυτό αναπτύχθηκε με στόχο να υποστηρίξει την ανάπτυξη και την εκτέλεση των εφαρμογών πολυμέσων που θα έτρεχαν πάνω στον εξυπηρετητή. Το υποσύστημα αποτελείται από τέσσερις διαφορετικές οντότητες: τον διαχειριστή επικοινωνίας (CM), το OVM (object virtual machine), τον διαχειριστή αντικειμένων (OM) και το υποσύστημα ανάπτυξης εφαρμογών (ADI). Οι παραπάνω οντότητες συνεργάζονται μεταξύ τους για να υποστηρίξουν την εκτέλεση εφαρμογών που χρειάζονται πρόσβαση στον εξυπηρετητή Kydonia. Περιληπτικά ολόκληρο το υποσύστημα του πελάτη αναλαμβάνει την δημιουργία και διαχείριση των μηνυμάτων που αποστέλλονται/έρχονται στον/από τον εξυπηρετητή, διαχειρίζεται τα αντικείμενα και υποστηρίζει λειτουργίες προσωρινής αποθήκευσης (caching). Τέλος μέσω του ADI επιτυγχάνεται η ανάπτυξη και η εγκατάσταση νέων εφαρμογών. Το ADI εξάγει την υποστηριζόμενη λειτουργικότητα για: διαχείριση του αντικειμενοστραφούς σχήματος της βάσης δεδομένων (π.χ. δημιουργία κλάσεων/στιγμιότυπων, διαγραφή/ενημέρωση στιγμιότυπων), συνόδους και διαχείριση συνδιαλλαγών, δημιουργία δεικτών και υποβολή ερωτήσεων και πλοήγηση πάνω σε HTML έγγραφα και MPEG-2 video.

3.2.2 Το υποσύστημα δικτύου και επικοινωνίας

Ο επεξεργαστής γραμμής (LPM) αναλαμβάνει την διανομή των δεδομένων και των μηνυμάτων ελέγχου πάνω από το δίκτυο. Μια διεργασία LPM τρέχει σε κάθε πελάτη και είναι υπεύθυνη για την επικοινωνία πελάτη-εξυπηρετητή, ενώ μια άλλη πιο πολύπλοκη διεργασία τρέχει στην πλευρά του εξυπηρετητή με δυνατότητα υποστήριξης πολλών πελατών

3.2.3 Το υποσύστημα διαχείρισης αντικειμένων και συνόδων

Το υποσύστημα αυτό (OSM) εκτός από την αποκωδικοποίηση και δρομολόγηση των εισερχόμενων μηνυμάτων, αναλαμβάνει την παροχή λειτουργικότητας όπως: διαχείριση συνόδων και συνδιαλλαγών, διαχείριση των αντικειμένων στην κύρια μνήμη, δεικτοδότηση, ερωτήσεις και υποστήριξη καταλόγου.

Όπως φαίνεται από το σχήμα αποτελείται από αρκετά διαφορετικά κομμάτια. Ο δρομολογητής μακράς εμβέλειας (LTS) αναλαμβάνει την αρχική δρομολόγηση των αιτήσεων με την ανάθεση στατικών προτεραιοτήτων. Συνεργάζεται και με άλλα κομμάτια λογισμικού (π.χ. με τον διαχειριστή καταλόγου) για την συλλογή πληροφορίας σε μια προσπάθεια εκτίμησης του φόρτου του συστήματος. Ο Διαχειριστής αντικειμένων (OM) είναι υπεύθυνος για την πρόσβαση στα αντικείμενα που βρίσκονται στην κύρια μνήμη του εξυπηρετητή. Ο Διαχειριστής συνδιαλλαγών εγγυάται την ορθή εκτέλεση των συνδιαλλαγών σε συνεργασία με τον ελεγκτή ταυτοχρονισμού. Τέλος ο διαχειριστής συνόδων αρχικοποιεί και τερματίζει συνόδους των πελατών και αποθηκεύει πληροφορία που περιγράφει την χρήση των πόρων του συστήματος κατά την διάρκεια μιας συνόδου.

Οι ερωτήσεις στις παραμέτρους των αντικειμένων εκφράζονται με μια γλώσσα όμοια με την SQL και ο διαχειριστής δεικτοδότησης και ερωτήσεων βελτιστοποιεί την εκτέλεση τους.

Το υποσύστημα OSM αποτελεί το σημείο πρόσβασης στον εξυπηρετητή από τους πελάτες και βελτιώνει την συνολική ευελιξία του συστήματος, επιτρέποντας την επεξεργασία και διαχείριση των αντικειμένων του συστήματος και των εφαρμογών εσωτερικά στον εξυπηρετητή.

3.2.4 Το υποσύστημα αποθήκευσης

Η αποθήκευση των δεδομένων στην δευτερεύουσα και τριτεύουσα μνήμη αναλαμβάνεται από δύο βασικά κομμάτια λογισμικού, τα υποσυστήματα διαχείρισης αποθήκευσης και διαχείρισης δίσκων.

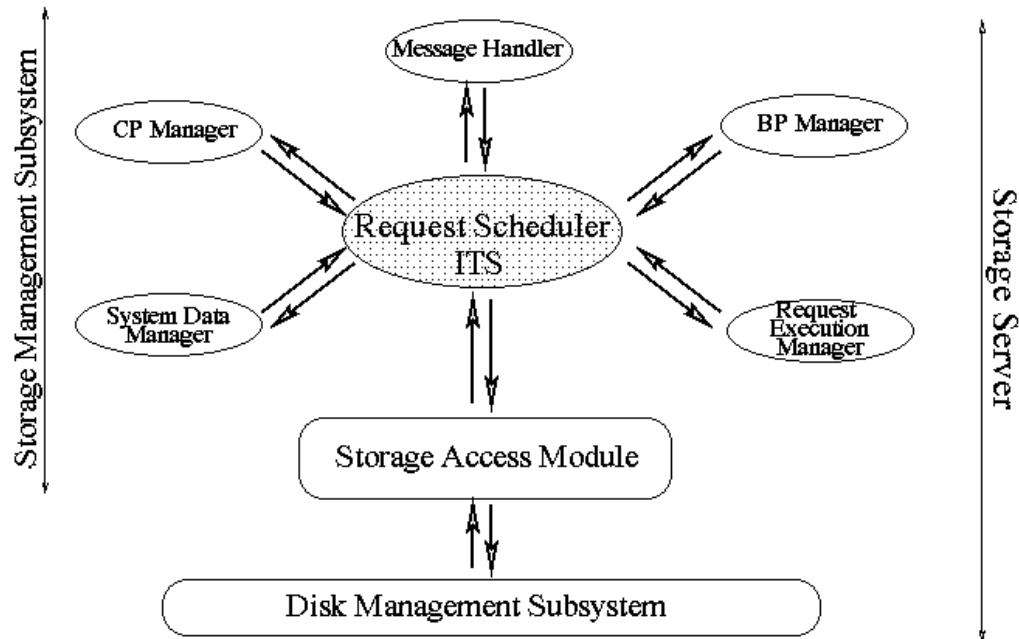
Οι αρχικές εκδόσεις του υποσυστήματος διαχείρισης υποστήριζαν ένα περιβάλλον αντικειμενοστραφούς διαχείρισης πάνω σε ιεραρχίες αποθήκευσης. Προοδευτικά το υποσύστημα εμπλουτίστηκε με δυνατότητες και μηχανισμούς για την υποστήριξη πολλών πελατών, ενώ οι πιο πρόσφατες προσπάθειες έχουν επικεντρωθεί προς την μετατροπή του σε ένα ισχυρό εξυπηρετητή αποθήκευσης πολυμέσων.

Η οργάνωση της πληροφορίας μέσα στο υποσύστημα διαχείρισης αποθήκευσης είναι ιεραρχική και αποτελείται από τρία επίπεδα: Τα λογικά αρχεία, τα φυσικά αρχεία και τα μέσα αποθήκευσης. Εφόσον αυτό το υποσύστημα είναι ο τελικός αποδέκτης των αιτήσεων για αποθήκευση και ανάκτηση αντικειμένων, τεχνικές όπως δρομολόγηση της προσπέλασης της δευτερεύουσας και τριτεύουσας μνήμης, ομαδοποίηση αντικειμένων (object clustering), παράλληλη αποθήκευση (striping), αντίγραφα αντικειμένων (object replication), πολιτικές προσωρινής αποθήκευσης και στρατηγικές αντικατάστασης δεδομένων (data replacement), οι οποίες έχουν σημαντική επίδραση στην απόδοση του συστήματος, υποστηρίζονται από τον εξυπηρετητή αποθήκευσης.

Η ομαδοποίηση των δεδομένων (data clustering) σε λογικό και φυσικό επίπεδο έχει επίσης αναπτυχθεί. Η λογική ομαδοποίηση των δομημένων (ή παραδοσιακών) αντικειμένων βασίζεται στην κλάση στην οποία ανήκουν και στην σχετική τους θέση στην ιεραρχία της κλάσης. Η εφαρμογή μπορεί να καθορίσει τους κανόνες ομαδοποίησης. Όλα τα αντικείμενα που ανήκουν σε μια λογική ομάδα αποθηκεύονται συνεχόμενα στον δίσκο (φυσικό επίπεδο). Εφόσον τα δεδομένα σε μια φυσική ομάδα αποθηκεύονται συνεχόμενα στον δίσκο, η φυσική ομάδα είναι στην πραγματικότητα η μονάδα αποθήκευσης. Κατά την δημιουργία μια λογική ομάδα αντιστοιχεί σε μια φυσική ομάδα. Καθώς όμως τα αντικείμενα αποθηκεύονται/ενημερώνονται στις λογικές ομάδες, περισσότερες φυσικές ομάδες δημιουργούνται, σύμφωνα με κανόνες που έχουν καθοριστεί από τον διαχειριστή του συστήματος (system administrator), και ανήκουν στην ίδια λογική ομάδα. Ο προγραμματιστής εφαρμογών έχει την δυνατότητα να ρυθμίσει τον εξυπηρετητή αποθήκευσης, έτσι ώστε να πετύχει καλύτερη απόδοση. Αυτό μπορεί να γίνει μέσω μιας αυστηρά καθορισμένης script γλώσσας που επιτρέπει την ρύθμιση των χαρακτηριστικών των φυσικών αρχείων, του μεγέθους των φυσικών ομάδων, όσο και τον ορισμό και της θέσης των διαμερίσεων στους δίσκους.

Τα αντικείμενα μεγάλου μεγέθους και λειτουργίες σε αυτά έχουν μοντελοποιηθεί στο υποσύστημα αποθήκευσης, έτσι ώστε να υποστηρίξουν δεδομένα πολυμέσων με αυξημένες

απαιτήσεις αποθηκευτικού χώρου. Ο εξυπηρετητής διαιρεί αντικείμενα ήχου και video σε λογικά κομμάτια καθώς τα αποθηκεύει στους δίσκους. Τα κομμάτια αυτά τοποθετούνται συνεχόμενα ή διεσπαρμένα στο μέσο αποθήκευσης. Οι δύο αυτές διαφορετικές τεχνικές υποστηρίζονται με την δυνατότητα συνεχόμενης αποθήκευσης ενός αντικειμένου στο δίσκο ή επιτρέποντας τον συνδυασμό των πολιτικών striping και interleaving σε πολλαπλά μέσα αποθήκευσης. Στο Σχήμα 17 περιγράφεται γραφικά η αρχιτεκτονική του υποσυστήματος διαχείρισης αποθήκευσης και των δομικών του στοιχείων.



Σχήμα 17: Η αρχιτεκτονική του υποσυστήματος διαχείρισης αποθήκευσης
Ο δρομολογητής αιτήσεων αποτελεί την καρδιά επεξεργασίας του υποσυστήματος αποθήκευσης. Ελέγχει την ροή της πληροφορίας και διαχειρίζεται την εξυπηρέτηση των αιτήσεων. Επίσης αλληλεπιδρά με άλλα κομμάτια λογισμικού για την συλλογή των απαραίτητων για την δρομολόγηση στοιχείων ή για την εκτέλεση συγκεκριμένων λειτουργιών.

Ο διαχειριστής μηνυμάτων (Message Handler) είναι ο αποδέκτης των μηνυμάτων από άλλες οντότητες λογισμικού σε χαμηλότερα ή υψηλότερα επίπεδα στην αρχιτεκτονική και είναι υπεύθυνος για την αποκωδικοποίησή τους και την προώθησή τους στις κατάλληλες ουρές του συστήματος. Ο δρομολογητής αιτήσεων (ITS) είναι τμήμα της συνολικής δρομολόγησης τριών επιπέδων που εφαρμόζεται και χαρακτηρίζεται μεσαίας κλίμακας. Ο δρομολογητής αιτήσεων αλληλεπιδρά και διαχειρίζεται την λειτουργία των: διαχειριστή εκτέλεσης (REM), του διαχειριστή αποταμιευτή (BPM), του διαχειριστή αποταμιευτή επικοινωνίας (CPM), και του υποσυστήματος πρόσβασης αποθήκευσης (SAM). Το υποσύστημα SAM αποτελεί το βασικό κομμάτι της αποθηκευτικής μηχανής. Είναι υπεύθυνο για την εφαρμογή των μηχανισμών ομαδοποίησης, διαχείρισης μεγάλων αντικειμένων πολυμέσων, ανάνηψης και διαχείρισης των αποθηκευτικών μέσων (διατηρώντας την δικιά του πληροφορία προσπέλασης των μέσων, παρακάμπτοντας το σύστημα αρχείων του λειτουργικού). Το υποσύστημα πρόσβασης αποθήκευσης διαχειρίζεται επίσης όλα τα μέσα αποθήκευσης με ένα τρόπο διαφανή. Ένα γενικό μοντέλο έχει αναπτυχθεί, το οποίο λαμβάνει υπόψη όλα τα χαρακτηριστικά διαφορετικών μέσων αποθήκευσης (οπτικά,

μαγνητικά, μέσα τριτεύουσας μνήμης) και εφαρμόζει μια αποδοτική αντιστοίχιση ανάμεσα σε ένα γενικό σύνολο λειτουργιών φυσικού επιπέδου με τις ειδικές για κάθε μέσο λειτουργίες.

Το υποσύστημα διαχείρισης δίσκων βρίσκεται στο χαμηλότερο επίπεδο της αρχιτεκτονικής και έχει ως κύριο στόχο την εξυπηρέτηση των αιτήσεων που χρειάζονται πρόσβαση σε μέσα δευτερεύουσας και τριτεύουσας μνήμης. Αποτελείται από ένα αριθμό διεργασιών προσπέλασης δίσκου που καλούνται DAMs. Κάθε DAM αντιστοιχεί σε ένα μοναδικό οδηγό δίσκου και αναλαμβάνει την διαχείριση, δρομολόγηση και διανομή των αιτούμενων δεδομένων στο υποσύστημα διαχείρισης αποθήκευσης για παραπέρα επεξεργασία. Οι διεργασίες δίσκου έχουν την δυνατότητα να διανεμηθούν σε έναν αριθμό υπολογιστικών μηχανών δίνοντας με αυτόν τον τρόπο μεγαλύτερη ευελιξία στον εξυπηρετητή αποθήκευσης. Ακολουθώντας το σχήμα αυτό έγινε δυνατή η υποστήριξη της παράλληλης προσπέλασης πολλαπλών μέσων και συνεπώς η υποστήριξη των απαιτητικών δεδομένων πολυμέσων, από τον εξυπηρετητή αποθήκευσης.

3.2.5 Ο αρχικός σχεδιασμός και ο μηχανισμός δρομολόγησης

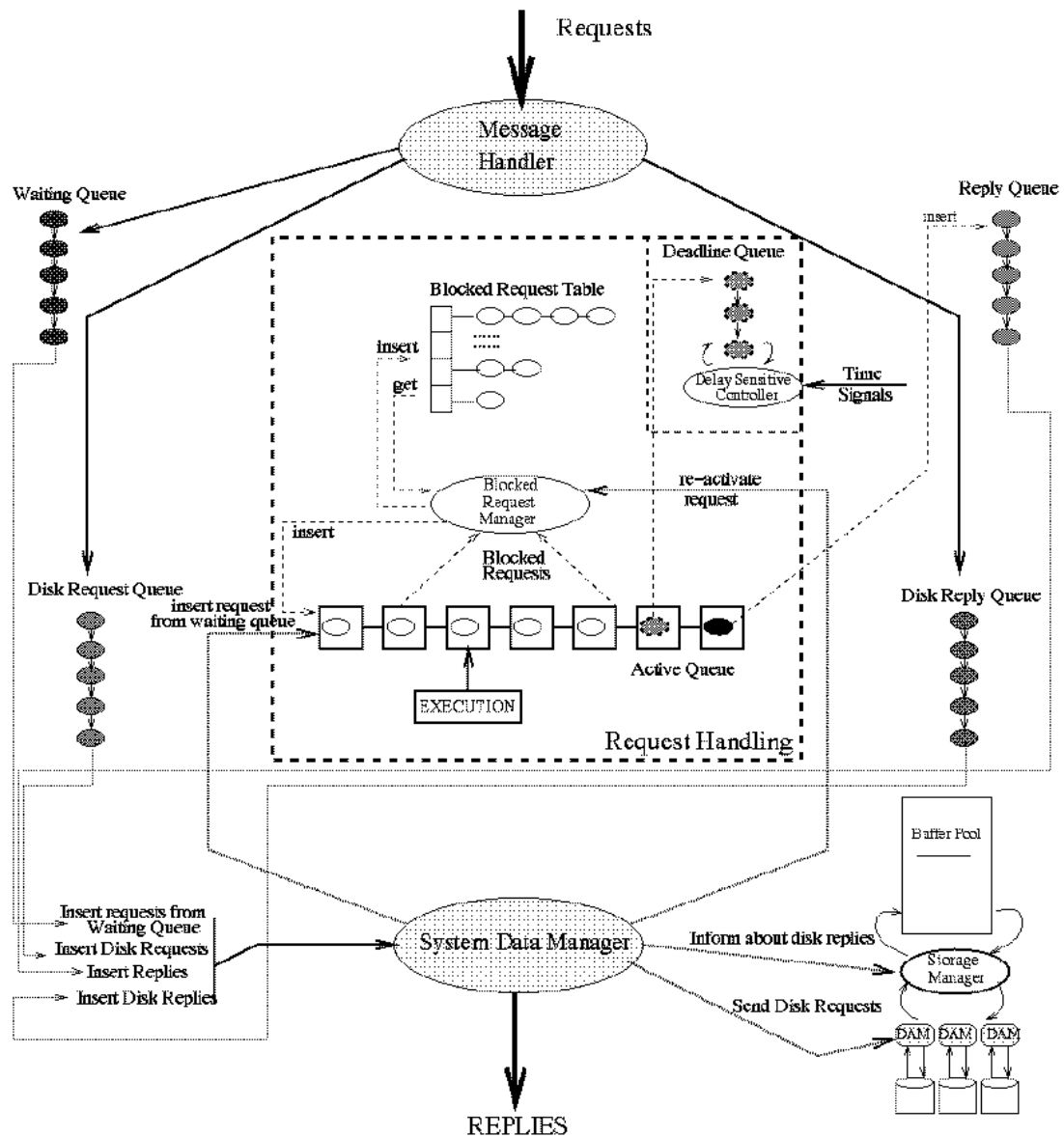
Το σύστημα Kydonia ενσωματώνει αρκετούς δρομολογητές. Ο σχεδιασμός προϋποθέτει την ύπαρξη μιας ιεραρχίας δρομολόγησης τριών επιπέδων, στην οποία επιτρέπεται η ανεξάρτητη λειτουργία των διαφορετικών δρομολογητών. Όπως έχει αναφερθεί στο υψηλότερο σημείο βρίσκεται ο δρομολογητής μεγάλης εμβέλειας LTS στο υποσύστημα OSM. Στον εξυπηρετητή αποθήκευσης εφαρμόζονται τα δύο επόμενα επίπεδα δρομολογητών μεσαίας και μικρής εμβέλειας ITS και STS. Ο ITS έχει σαν στόχο να διαχειριστεί την πρόσβαση στην δευτερεύουσα μνήμη κατά την εκτέλεση των αιτήσεων με ευέλικτο τρόπο. Συνεπώς με στόχο την αύξηση της απόδοσης ανάκτησης δεδομένων, Ο ITS εκμεταλλεύεται τις τεχνικές παράλληλης προσπέλασης των μέσων αποθήκευσης και προσπαθεί να επιβάλλει εξυπηρέτηση πραγματικού χρόνου. Στο υποσύστημα διαχείρισης δίσκων ενεργούν πολλοί δρομολογητές STS, ο οποίοι οδηγούν την εκτέλεση των υποαιτήσεων προσπέλασης δίσκου για κάθε δίσκο χωριστά. Ευνόητο είναι ότι αυτοί οι δρομολογητές προσπαθούν με την πληροφορία που διαθέτουν να καθορίσουν την σειρά εκείνη της εκτέλεσης των υποαιτήσεων που θα εξασφαλίζει την ελαχιστοποίηση των χρόνων μετακινήσεων της κεφαλής και θα εγγυάται την ικανοποίηση των προθεσμιών για τις αιτήσεις πραγματικού χρόνου.

Στην επόμενη ενότητα περιγράφεται ο αρχικός σχεδιασμός και ανάπτυξη του δρομολογητή μεσαίας εμβέλειας ή αλλιώς δρομολογητή αιτήσεων του εξυπηρετητή αποθήκευσης

3.2.5.1 Ο Δρομολογητής αιτήσεων

Κύριες υπευθυνότητες του δρομολογητή αιτήσεων είναι η δρομολόγηση: της πρόσβασης στην τριτεύουσα μνήμη, της εκτέλεσης ομάδων αιτήσεων (batch requests), της ανάκτησης παραδοσιακών δεδομένων και της ανάκτησης δεδομένων ευαίσθητων στις χρονικές καθυστερήσεις. Το σχήμα εμφανίζει το μοντέλο δρομολόγησης που εφαρμόστηκε στον εξυπηρετητή αποθήκευσης. Τρεις ουρές προτεραιότητας, η ουρά αναμονής, η ενεργή και η ουρά απόκρισης, μαζί με δύο ουρές πρώτης άφιξης (FIFO queues) για τις υποαιτήσεις

δίσκου (disk request, disk reply), μια ουρά μικρότερης προθεσμίας (EDF queue) και ένα δυναμικό πίνακα κατακερματισμού (dynamic hashing table) για τις ανασταλμένες αιτήσεις (blocked), αποτελούν τις βασικές δομές υποστήριξης του δρομολογητή αιτήσεων.



Σχήμα 18: Το μοντέλο του μηχανισμού δρομολόγησης

Υπευθυνότητα του διαχειριστή μηνυμάτων είναι η λήψη των μηνυμάτων η αποκωδικοποίησή τους και η τοποθέτηση ανάλογα με τον τύπο τους στις αντίστοιχες ουρές. Οι ουρές αυτές επεξεργάζονται περιοδικά από τον διαχειριστή ουρών του συστήματος. Οι αιτήσεις ανάκτησης δεδομένων ροής αντιμετωπίζονται από τον ελεγκτή χρονικά ευαίσθητων δεδομένων και τοποθετούνται στην ουρά προθεσμιών

Η προτεραιότητα των αιτήσεων στην ουρά αναμονής (που κρατά τις εισερχόμενες αιτήσεις) και στην ενεργή ουρά (που κρατά τις τρέχουσες αιτήσεις οι οποίες εκτελούνται ταυτόχρονα), καθορίζεται από αρκετούς παράγοντες όπως ο τύπος της αίτησης, ο τύπος δεδομένων, η παλαιότητα (aging), και η φάση επεξεργασίας και υπολογίζεται μέσω μιας ευριστικής μεθόδου με χρήση βαρών. Η αίτηση με την μεγαλύτερη προτεραιότητα στην ενεργή ουρά επιλέγεται από τον δρομολογητή σε κάθε κύκλο εξυπηρέτησης να αρχίσει ή να

συνεχίσει την εκτέλεσή της. Όταν η εκτέλεση μιας αίτησης δεν μπορεί να συνεχίσει λόγω έλλειψης διαθέσιμων πόρων ή λόγω ανάγκης προσπέλασης των αποθηκευτικών μέσων, τότε η αίτηση αυτή εξέρχεται από την διεργασία εκτέλεσης αποκτώντας μια τιμή που υποδηλώνει την κατάστασή της. Αν η τιμή αυτή σημαίνει ότι χρειάζεται πρόσβαση σε κάποιο δίσκο, τότε η αίτηση μετακινείται από την ενεργή ουρά στην ουρά ανασταλμένων αιτήσεων στην οποία παραμένει μέχρι να είναι διαθέσιμα τα κατάλληλα δεδομένα. Διαφορετικά η προτεραιότητα της αίτησης επανα-υπολογίζεται λαμβάνοντας υπόψη την τιμή εξόδου και ενημερώνεται η θέση της στην ενεργή ουρά.

Όλες οι ουρές του συστήματος επεξεργάζονται περιοδικά από μια διεργασία που καλείται διαχειριστής του συστήματος. Οι ενέργειες αυτής της διεργασίας περιλαμβάνουν: Μετακίνηση των αιτήσεων από την ουρά αναμονής στην ενεργή ουρά, υποβολή των υποαιτήσεων προσπέλασης δίσκου σε ανενεργούς δίσκους, ενημέρωση του διαχειριστή ανασταλμένων αιτήσεων για πρόσφατες απαντήσεις από τις διεργασίες διαχείρισης των δίσκων και τέλος αποστολή των αποτελεσμάτων των αιτήσεων στα ανώτερα επίπεδα της ιεραρχίας του συστήματος

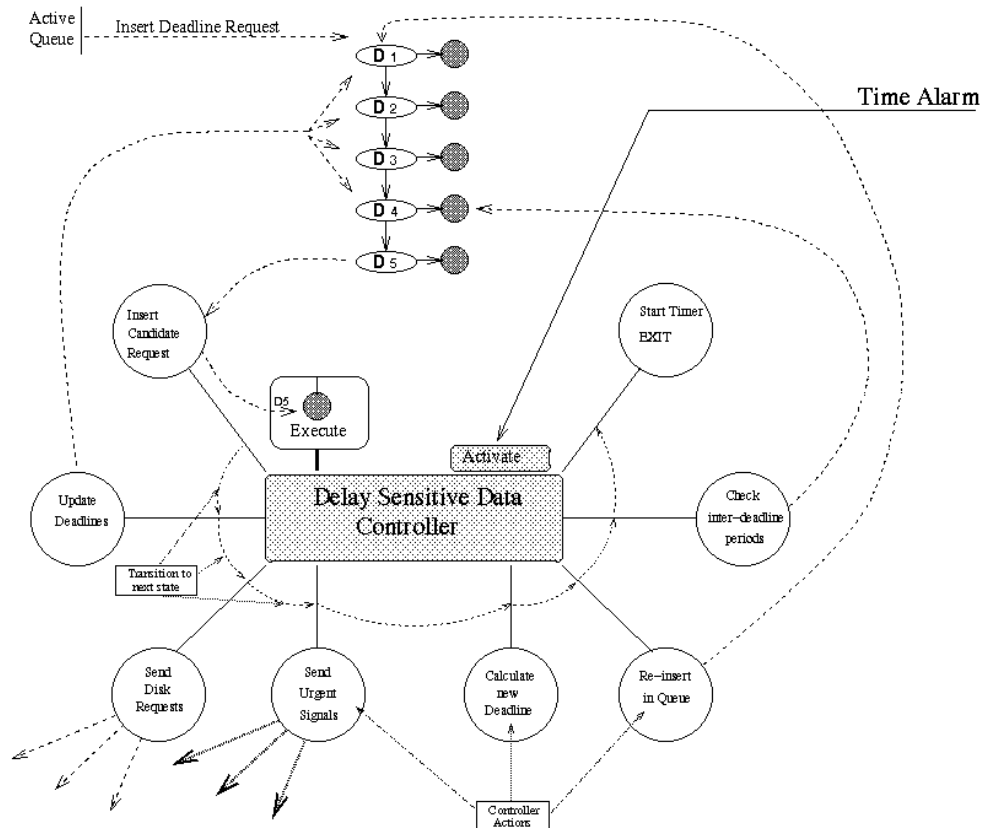
3.2.5.2 Υποστήριξη δεδομένων ροής

Ο αρχικός σχεδιασμός του μηχανισμού δρομολόγησης προσπάθησε να καλύψει και την ανάγκη υποστήριξης δεδομένων, τα οποία αποκτούν την ορθή τους έννοια μόνο όταν η ανάκτησή τους συμβαίνει συνεχόμενα σε πραγματικό χρόνο με συγκεκριμένες απαιτήσεις όπως αντικείμενα ήχου ή video και αποδίδονται με τον όρο δεδομένα συνεχόμενης ροής (Stream objects). Συνεπώς ο δρομολογητής αιτήσεων λαμβάνει υπόψη παραμέτρους όπως ο ρυθμός κατανάλωσης δεδομένων, ο φόρτος εργασίας των δίσκων, ο διαθέσιμος χώρος των προσωρινών αποταμιευτών και τα χαρακτηριστικά των δίσκων (ρυθμός μεταφοράς, χρόνος αναζήτησης κτλ.) , έτσι ώστε να επιβάλλει μια ικανοποιητική δρομολόγηση στην εκτέλεση των αιτήσεων που ζητούν ανάκτηση δεδομένων συνεχόμενης ροής.

Όταν μια αίτηση που ζητά ανάκτηση ενός αντικειμένου συνεχόμενης ροής εισάγεται στην ενεργή ουρά αρχίζει να εκτελείται και συνεχίζει μέχρι η αίτηση να αποκτήσει μια τιμή που υποδηλώνει ότι είναι έτοιμη για την έναρξη της διαδικασίας αναπαραγωγής (playback). Αυτό σημαίνει ότι έχει δεσμευθεί ο κατάλληλος χώρος στον προσωρινό αποταμιευτή επικοινωνίας (ακολουθώντας το double-buffering σχήμα), το πρώτο τμήμα δεδομένων είναι διαθέσιμο από τους δίσκους και όλη η πληροφορία δεικτών για την τοποθέτηση του αντικειμένου στα μέσα αποθήκευσης είναι διαθέσιμη (πληροφορία απαραίτητη για την άμεση ανάκτηση των υπόλοιπων τμημάτων). Σε αυτή την φάση η αίτηση μετακινείται από την ενεργή ουρά στην ουρά προθεσμιών καθορίζοντας ταυτόχρονα την προθεσμία της επόμενης φάσης επεξεργασίας της αίτησης. Η εξυπηρέτηση της αίτησης σε αυτό το σημείο αναλαμβάνεται από τον ελεγκτή χρονικά ευαίσθητων δεδομένων (DSD-Controller). Ο ελεγκτής αυτός ενεργοποιείται περιοδικά από χρονικά σήματα διακοπής (time interrupts) που υποδηλώνουν ότι η πιο πρόσφατη προθεσμία κάποιας αίτησης έχει εξαντληθεί.

Όταν ο ελεγκτής είναι ενεργός επεξεργάζεται την πρώτη αίτηση από την ουρά προθεσμιών, αποστέλλει τις κατάλληλες υποαιτήσεις προσπέλασης δίσκου και επαναλαμβάνει την ίδια διαδικασία για όλες τις αιτήσεις που έχουν σχεδόν την ίδια

προθεσμία. Αν η προθεσμία μιας αίτησης είναι αρκετά μακριά από τον τρέχοντα χρόνο, ο ελεγκτής καθίσταται ανενεργός, μέχρι να συμβεί η επόμενη χρονική διακοπή.



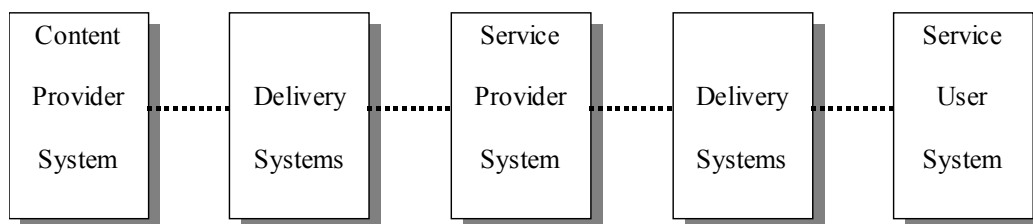
Σχήμα 19: Μηχανή καταστάσεων του ελεγκτή χρονικά ευαίσθητων δεδομένων
Στο σχήμα εικονίζονται οι ενέργειες του διαχειριστή χρονικά ευαίσθητων δεδομένων μετά την ενεργοποίησή του από σήμα χρονικού συναγερμού. Με τους κύκλους αναπαριστούνται τα στάδια από τα οποία διέρχεται ο διαχειριστής και η σειρά διαδοχής τους από αριστερά προς τα δεξιά.

Κατά τη διάρκεια των ανενεργών καταστάσεων του ελεγκτή χρονικά ευαίσθητων δεδομένων, ο δρομολογητής αιτήσεων εξυπηρετεί τις υπόλοιπες αιτήσεις που δεν έχουν αυστηρές απαιτήσεις πραγματικού χρόνου, επιτυγχάνοντας έναν μηχανισμό πολύπλεξης της επεξεργασίας παραδοσιακών δεδομένων και δεδομένων συνεχόμενης ροής. Στο Σχήμα 19 εμφανίζονται οι φάσεις λειτουργίας του DSD-Controller.

3.3 Ο εξυπηρετητής πολυμέσων Kydonia-D

Στα πλαίσια του ερευνητικού ευρωπαϊκού προγράμματος SICMA, έγιναν οι απαραίτητες ενέργειες και μετατροπές στον εξυπηρετητή Kydonia, ώστε να ακολουθήσει το πρότυπο DAVIC [DAV95]. Το πρότυπο DAVIC, καθορίζει το πλαίσιο με το οποίο παρέχονται υπηρεσίες πολυμέσων από μεγάλης κλίμακας εξυπηρετητές, μέσω συστημάτων διανομής, στα Set Top Units στην πλευρά των πελατών-χρηστών. Ο κύριος στόχος του προτύπου είναι να καταστήσει δυνατή την διαλειτουργικότητα ανάμεσα σε διαφορετικές οντότητες, ώστε να επιτύχει την ολοκλήρωση των ψηφιακών οπτικό-ακουστικών εφαρμογών και υπηρεσιών. Η συνολική αρχιτεκτονική των υπηρεσιών ορίζεται με δηλώσεις στοιχείων

υπηρεσίας (service elements), λειτουργίες, συστημάτων ζεύξης και επαφής (interfaces), ροών πληροφορίας (information flows) και στοιβάς πρωτοκόλλων (protocol stacks). Ιδιαίτερη σημασία αποδίδεται στην μετάδοση MPEG-2 transport streams πάνω από ATM δίκτυα υψηλών ταχυτήτων.



Σχήμα 20: Σύστημα DAVIC

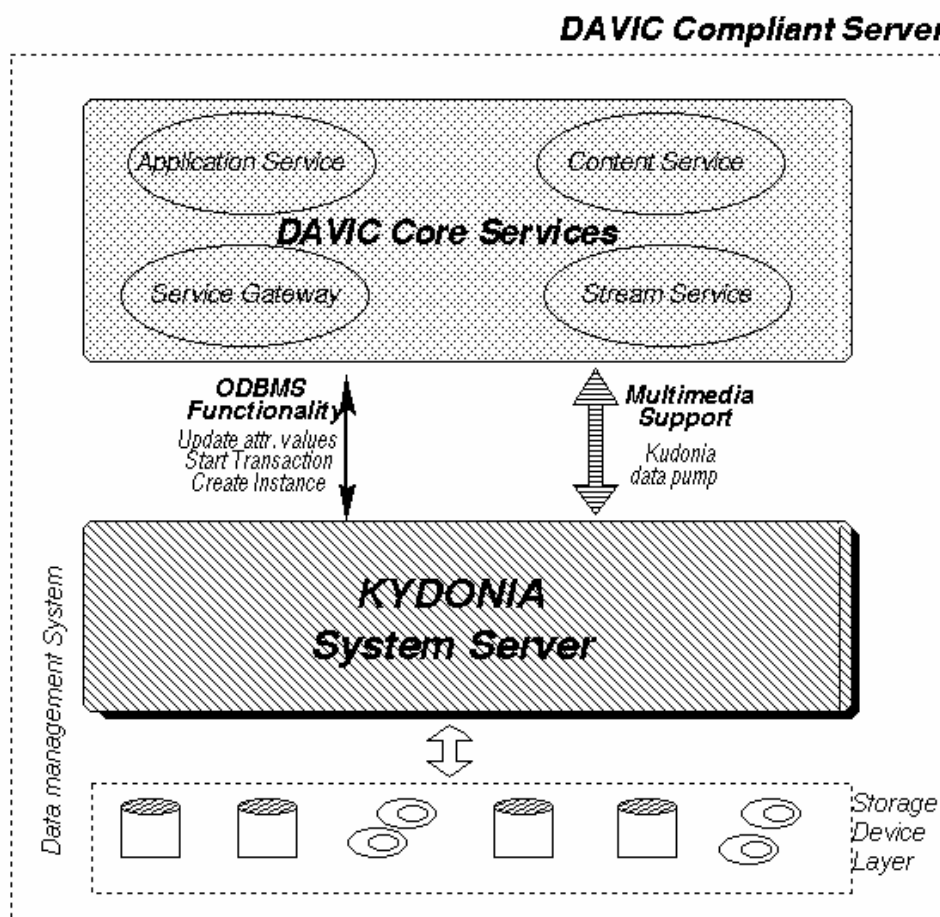
Στο σχήμα εμφανίζονται τα επιμέρους υποσυστήματα από τα οποία αποτελείται ένα πληροφορικό σύστημα συμβατό με το πρότυπο DAVIC. Το σύστημα παροχής περιεχομένου είναι υπεύθυνο για την είσοδο δεδομένων και πληροφορίας, ενώ σύστημα παροχής υπηρεσιών αποτελεί τον κεντρικό εξυπηρετητή. Τα επιμέρους κομμάτια ενώνονται μεταξύ τους με συστήματα μεταφοράς δεδομένων

Σύμφωνα με το πρότυπο, ο εξυπηρετητής πρέπει να ακολουθεί μια προσέγγιση σε διανεμημένο σχήμα και αντικειμενοστραφή. Επίσης πρέπει να αποτελείται από στοιχεία υπηρεσίας τα οποία παρέχουν μέσω ενός καλά ορισμένου συστήματος ζεύξης μια ποικιλία από υπηρεσίες για την υποστήριξη εφαρμογών πολυμέσων.

Η ανάπτυξη της λειτουργικότητας, όπως ορίζεται από το DAVIC, πάνω σε ένα σύστημα βάσεων δεδομένων πολυμέσων, όπως είναι το Kydonia, παρέχει σημαντικά πλεονεκτήματα σχετικά με άλλες προσεγγίσεις. Τα πλεονεκτήματα αναφέρονται στους τομείς της διαχείρισης αντικειμένων πολυμέσων, της οργάνωσης και επέκτασης της λειτουργικότητας του συστήματος, της πλοήγησης και συσχέτισης αντικειμένων και τέλος στην ανάπτυξη επιπρόσθετων μηχανισμών υποστήριξης όπως η ασφάλεια, τιμολόγηση και ερωτήσεις.

Η προσπάθεια και οι εργασίες που έλαβαν χώρα ώστε ο εξυπηρετητής Kydonia να γίνει συμβατός με το πρότυπο DAVIC, ενίσχυσαν τον εξυπηρετητή με καινούργια χαρακτηριστικά όπως νέα πρωτόκολλα και συστήματα ζεύξης, ενώ παράλληλα οδήγησαν στην ενσωμάτωση στο λογισμικό του εξυπηρετητή νέων μηχανισμών για την αποδοτικότερη διαχείριση των δεδομένων πολυμέσων, ώστε να ανταποκριθεί στις απαιτήσεις ποιότητας υπηρεσιών (Quality Of Service) που απορρέουν από το πρότυπο. Με βάση την αρχιτεκτονική του DAVIC για τον εξυπηρετητή έχουν αναπτυχθεί τα ακόλουθα βασικά στοιχεία υπηρεσιών (Core Service Elements): Η υπηρεσία περιεχομένου (Content Service), η οποία αναλαμβάνει την εγκατάσταση νέων εφαρμογών μέσα στον εξυπηρετητή, η υπηρεσία εισόδου (Service Gateway) που αναλαμβάνει την οργάνωση των υπηρεσιών (service domain) και παρέχει την δυνατότητα στους εξωτερικούς πελάτες να ανακαλύψουν και να χρησιμοποιήσουν διαθέσιμες υπηρεσίες και η υπηρεσία ροής (Stream Service) η οποία οδηγεί την πρόσβαση σε χρονικά ευαίσθητα αντικείμενα και επιτρέπει στους πελάτες να έχουν έλεγχο της ροής των δεδομένων. Τα συστήματα ζεύξης (APIs) των στοιχείων υπηρεσίας είναι κυρίως βασισμένα στο DSM-CC User to User specification.

Τα δεδομένα συνεχόμενης ροής μεταδίδονται μέσω του MPEG-2 Transport πρωτοκόλλου, ενώ για την μεταφορά των κλήσεων των υπηρεσιών από τους πελάτες ακολουθείται η αρχιτεκτονική CORBA και χρησιμοποιείται το Orbix 2.0.1 προϊόν.



Σχήμα 21:Ο Kydonia-D εξυπηρετητής

Τα βασικά στοιχεία υπηρεσίας (Core Service elements) αναπτύχθηκαν πάνω στον εξυπηρετητή βάσης πολυμέσων Kydonia και αποτελούν το κέλυφος με το οποίο επιτυγχάνεται η συμβατότητα με το πρότυπο DAVIC καταλήγοντας στον αναμορφωμένο Kydonia-D εξυπηρετητή.

Ο απώτερος στόχος του DAVIC είναι να αυξήσει την διαλειτουργικότητα και συνεπώς η εσωτερική υλοποίηση του εξυπηρετητή δεν καθορίζεται. Έτσι αυξάνεται η ευελιξία καθώς είναι υπευθυνότητα του εξυπηρετητή να οργανώσει και να διαχειριστεί τα δεδομένα πολυμέσων με τέτοιο τρόπο ώστε να ικανοποιήσει καλύτερα τις απαιτήσεις στην ροή δεδομένων που επιβάλλονται από τα ανώτερα επίπεδα. Το σύστημα Kydonia ενεργεί ως διαχειριστής δεδομένων (data manager) μέσα στον DAVIC εξυπηρετητή και χρησιμοποιείται από τα στοιχεία υπηρεσίας για την δημιουργία, οργάνωση και ανάκτηση πληροφορίας πολυμέσων.

Παράλληλα με την ανάπτυξη λειτουργικότητας συμβατής με το DAVIC, πάνω στον εξυπηρετητή Kydonia στα πλαίσια του προγράμματος SICMA, έγινε ουσιαστική έρευνα, πειραματισμοί και ανάπτυξη πάνω στον εξυπηρετητή αποθήκευσης που γίνεται τελικά ο αποδέκτης όλου του φόρτου για αποθήκευση και ανάκτηση δεδομένων. Κύριος στόχος ήταν η αύξηση της απόδοσης εξυπηρέτησης δεδομένων συνεχόμενης ροής (streams) με τον σχεδιασμό της αρχιτεκτονικής και την κατάλληλη χρήση τεχνικών ανάπτυξης λογισμικού που θα εξασφάλιζαν την βελτιστοποίηση των πόρων του συστήματος με απόρροια την μεγιστοποίηση της απόδοσης αλλά και θα παρείχαν τις απαραίτητες εγγυήσεις ποιότητας

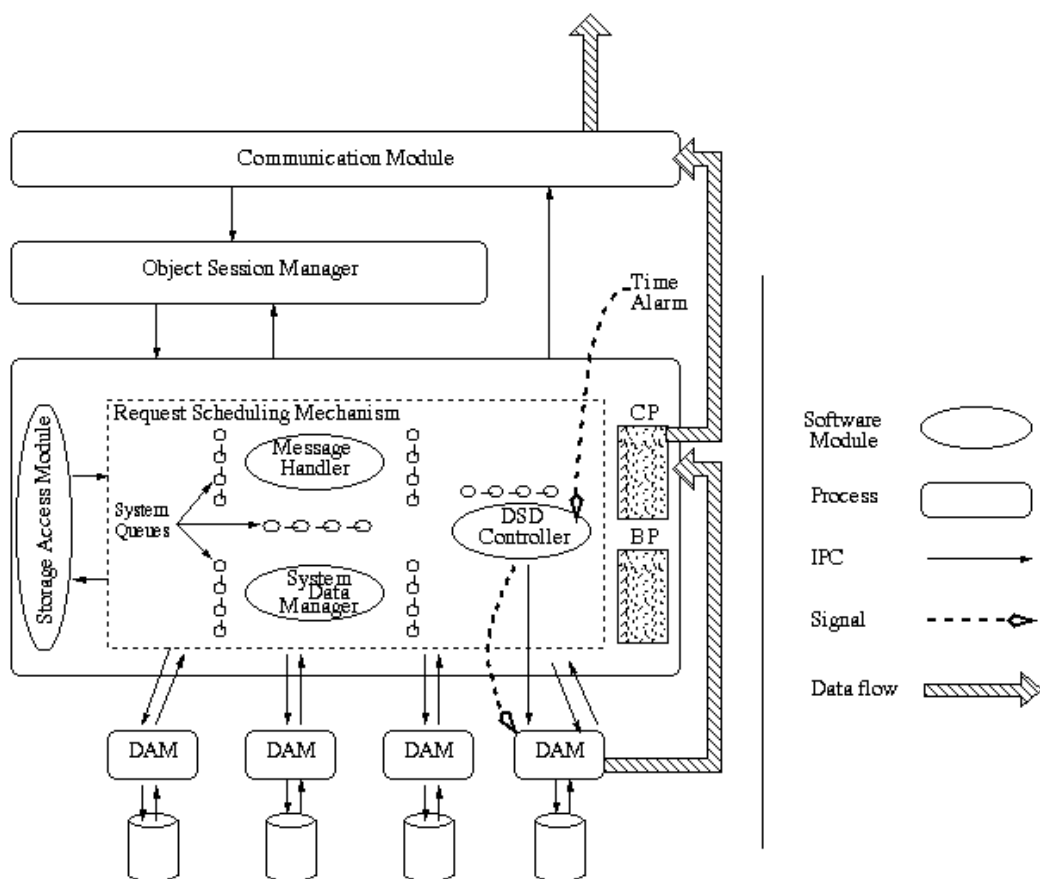
εξυπηρέτησης. Τα αποτελέσματα αυτής της προσπάθειας περιγράφονται στην τρέχουσα διατριβή.

3.4 Αρχιτεκτονική του μηχανισμού δρομολόγησης & τεχνική επισκόπηση

Ένας εξυπηρετητής αποθήκευσης πολυμέσων πρέπει να παρέχει πρόσβαση σε αντικείμενα πολυμέσων που περιλαμβάνουν κείμενο, ήχο, εικόνες και video. Όπως έχει αναφερθεί στο 2ο κεφάλαιο, ο σχεδιασμός ενός εξυπηρετητή αποθήκευσης πολυμέσων διαφέρει από τον σχεδιασμό των παραδοσιακών εξυπηρετητών καθώς οι απαιτήσεις για την υποστήριξη των δεδομένων συνεχόμενης ροής είναι μεγάλες. Στην ενότητα 3.2.5 παρουσιάστηκε ο σχεδιασμός του μηχανισμού δρομολόγησης που εφαρμόζεται στο Kydonia. Σε αυτή την ενότητα θα εξετάσουμε την τεχνική εφαρμογή αυτού του σχεδιασμού και θα αναφέρουμε τα αποτελέσματά της.

Στο Σχήμα 22 εικονίζονται οι διεργασίες οι οποίες υποστηρίζουν τον εξυπηρετητή Kydonia, τα ανεξάρτητα κομμάτια λογισμικού, οι αλληλεπιδράσεις, ο τρόπος επικοινωνίας τους, οι βασικές δομές υποστήριξης και η εσωτερική ροή δεδομένων. Σύμφωνα με το σχήμα ο κύριος όγκος των εργασιών λαμβάνει χώρα στο υποσύστημα διαχείρισης αποθήκευσης. Ο δρομολογητής αιτήσεων αναλαμβάνει την δρομολόγηση όλων των αιτήσεων που έρχονται στον εξυπηρετητή αποθήκευσης. Ο διαχειριστής μηνυμάτων οδηγεί τα εισερχόμενα μηνύματα στις κατάλληλες ουρές ανάλογα με την προέλευσή τους και τον τύπο τους. Οι αιτήσεις αυτές αναλύονται, επεξεργάζονται και μεταφράζονται σε νέες υποαιτήσεις προσπέλασης δίσκων. Ο δρομολογητής αιτήσεων αποτελεί στην ουσία μια πολύπλοκη μηχανή καταστάσεων (state machine) που τρέχει διαρκώς και υποστηρίζεται από δομές ουρών, με στόχο να πετύχει την ψευδο-ταυτόχρονη (concurrent) επεξεργασία των αιτήσεων. Περιοδικά ο διαχειριστής του συστήματος στέλνει πακέτα υποαιτήσεων για επεξεργασία στους ανενεργούς δίσκους. Η λειτουργία κάθε δίσκου διαχειρίζεται από την αντίστοιχη διεργασία διαχείρισης της προσπέλασης (DAM). Η επικοινωνία των διεργασιών (IPC) γίνεται με χρήση sockets.

Η διαχείριση των αιτήσεων ανάκτησης δεδομένων συνεχόμενης ροής γίνεται, μετά από μια αρχική προ-επεξεργασία, από τον ελεγκτή χρονικά ευαίσθητων δεδομένων DSD-Controller. Ο ελεγκτής αυτός σχεδιάστηκε για να επιβάλλει δρομολόγηση προθεσμιών EDF. Οι προθεσμίες αναπαριστούν ένα χρονικό σημείο κατά το οποίο πρέπει να υποβληθεί το επόμενο σύνολο υποαιτήσεων προσπέλασης δίσκου, ώστε να μην παραβιαστούν οι απαιτήσεις αναπαραγωγής του αντικειμένου. Δηλαδή ο ελεγκτής επιλέγει την αίτηση εκείνη με την μικρότερη προθεσμία, την επεξεργάζεται και στέλνει τις κατάλληλες υποαιτήσεις προσπέλασης στις αντίστοιχες διεργασίες δίσκου. Ταυτόχρονα υποστηρίζεται και ένας μηχανισμός αποστολής επειγόντων σημάτων (signals) στις διεργασίες δίσκου ώστε να διακόψουν το συντομότερο δυνατό την τρέχουσα εργασία των δίσκων, να δεχθούν τα αποσταλμένα μηνύματα των υποαιτήσεων και να εκτελέσουν τις εντολές ανάκτησης δεδομένων από τα μέσα αποθήκευσης.



Σχήμα 22: Τεχνική περιγραφή του εξυπηρετητή αποθήκευσης

Στο σχήμα εμφανίζεται η αρχιτεκτονική λογισμικού του εξυπηρετητή με έμφαση στο ενδο-επικοινωνιακό του σχήμα. Διακρίνονται τα κομμάτια λογισμικού, οι διεργασίες που δημιουργούνται και τρέχουν στο υπολογιστικό σύστημα και οι τρόποι επικοινωνίας τους. Εμφανής είναι επίσης η μεταβίβαση των αιτήσεων προς τον διαχειριστή αποθήκευσης και η ροή δεδομένων, όπως αυτή διαμορφώνεται για τα δεδομένα συνεχόμενης ροής

Στην προσπάθεια πολύπλεξης των αιτήσεων παραδοσιακών δεδομένων και των αιτήσεων που ζητούν ανάκτηση δεδομένων συνεχόμενης ροής, οι αιτήσεις χωρίς χρονικούς περιορισμούς εξυπηρετούνται στα μέσο-διαστήματα κατά τα οποία ο ελεγκτής είναι ανενεργός. Η ενεργοποίηση του ελεγκτή γίνεται με σήματα χρονικού συναγερμού (time alarm signals), που παράγονται με τον μηδενισμό ενός μετρητή χρόνου. Ο ελεγκτής μεταπηδά στην ανενεργό φάση όταν η επόμενη προθεσμία είναι αρκετά μακριά και αρχικοποιεί τον μετρητή χρόνου ώστε να σταλεί από το λειτουργικό σύστημα σήμα χρονικού συναγερμού στην διεργασία του διαχειριστή αποθήκευσης, όταν αυτή η προθεσμία εκπνεύσει. Στην ουσία δηλαδή ο ελεγκτής χρονικά ευαίσθητων δεδομένων αποτελεί ρουτίνα χειρισμού σήματος (signal handling process), η οποία καλείται κάθε φορά που λαμβάνεται ένα σήμα χρονικού συναγερμού από την διεργασία, διακόπτοντας την κανονική εκτέλεση των εργασιών του εξυπηρετητή. Στο Σχήμα 22 διακρίνεται επίσης το μονοπάτι που ακολουθεί η ροή δεδομένων μέσω των υποσυστημάτων του εξυπηρετητή. Τα δεδομένα που ανακτώνται από τους δίσκους εγγράφονται σε ειδικό προδεδειγμένο χώρο, που ακολουθεί το σχήμα διπλού αποταμιευτή (double buffer) στον προσωρινό αποταμιευτή επικοινωνίας (Communication Pool). Οι αντίστοιχες διεργασίες δίσκου ειδοποιούν με το κατάλληλο

μήνυμα τον κεντρικό δρομολογητή αιτήσεων ότι δεδομένα, μέρος κάποιου λογικού κομματιού ενός αντικειμένου ροής, είναι διαθέσιμα. Μόλις ο δρομολογητής ενημερωθεί ότι όλα τα απαραίτητα δεδομένα έχουν ανακτηθεί, αποστέλλει μήνυμα στον διαχειριστή επικοινωνίας (τον LPM για το παλαιότερο σύστημα ή στο στοιχείο υπηρεσίας ροής stream service element, για την DAVIC-συμβατή έκδοση), το οποίο περιέχει πληροφορία σχετική με την θέση των δεδομένων στον αποταμιευτή επικοινωνίας. Στην συνέχεια ο διαχειριστής επικοινωνίας αναλαμβάνει την αποστολή δεδομένων στους αιτούμενους πελάτες και επικοινωνεί με τον διαχειριστή αποθήκευσης για την αποδέσμευση χώρου στον αποταμιευτή καθώς αποστέλλονται δεδομένα.

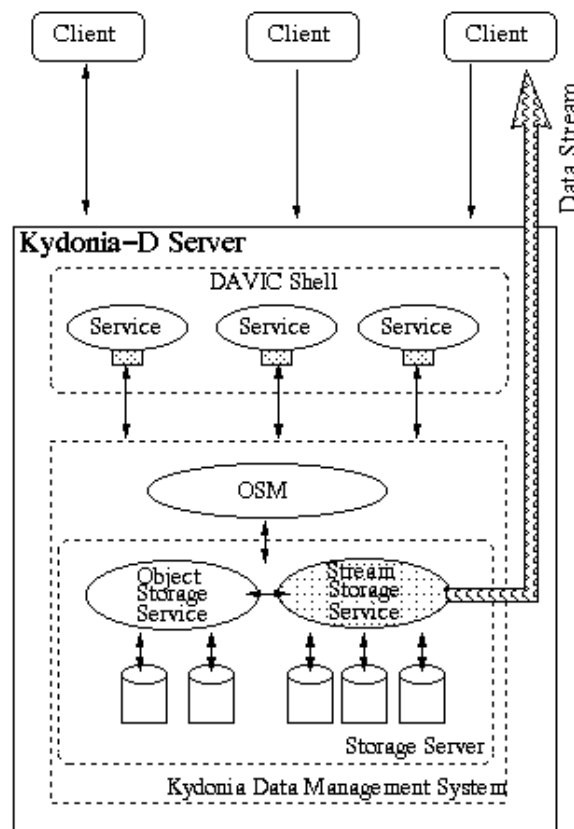
Η αρχική αυτή προσέγγιση για την αντιμετώπιση του προβλήματος δρομολόγησης της εξυπηρέτησης των αιτήσεων και της ροής δεδομένων μέσα στον εξυπηρετητή αποθήκευσης υπήρξε αρκετά επιτυχής. Με την ανάπτυξη και ενσωμάτωση του μηχανισμού δρομολόγησης που παρουσιάστηκε στον εξυπηρετητή αποθήκευσης έγινε δυνατή η υποστήριξη δεδομένων που απαιτούν συνεχόμενη ροή πραγματικού χρόνου, εξυπηρετώντας ταυτόχρονα πολλαπλούς πελάτες. Η εφαρμογή αυτού του μηχανισμού δρομολόγησης συντέλεσε σε ένα βαθμό στην προσπάθεια βελτίωσης των επιδόσεων του εξυπηρετητή με επιμέρους στόχους: την αύξηση της ρυθμοαπόδοσης (throughput), την μείωση του μέσου χρόνου απόκρισης των αιτήσεων, την άμβλυνση των διαφοροποιήσεων στον χρόνο απόκρισης και τέλος στην παροχή εγγυήσεων για την ικανοποίηση των απαιτήσεων πραγματικού χρόνου κατά την ανάκτηση μέσων ευαίσθητων στις καθυστερήσεις.

3.5 Αποτελέσματα-Βελτιώσεις

Η ανάπτυξη του εξυπηρετητή αποθήκευσης και του μηχανισμού δρομολόγησης αν και έδωσε την δυνατότητα διαχείρισης δεδομένων συνεχόμενης ροής εμφάνιζε μερικά μειονεκτήματα, τα οποία είχαν αντίκτυπο στην συνολική απόδοση του συστήματος και στην ποιότητα των παρεχόμενων υπηρεσιών. Κατά την υποστήριξη εφαρμογών πολυμέσων που ζητούσαν ανάκτηση δεδομένων ροής, είχαν παρατηρηθεί ορισμένα προβλήματα απόδοσης, ιδιαίτερα όταν ο εξυπηρετητής καλούνταν να εξυπηρετήσει μεγάλο αριθμό πελατών. Μια πιο προσεκτική μελέτη της αρχιτεκτονικής εμφάνισε το γεγονός ότι το σύστημα αποτελούνταν από αρκετές διεργασίες που έτρεχαν ταυτόχρονα. Η μεγάλη αλληλεπίδραση των διεργασιών αυτών και ο τρόπος επικοινωνίας τους, ανέδειξε ότι σε περιπτώσεις υψηλού φόρτου εργασίας διακινούνταν ένας μεγάλος αριθμός μηνυμάτων και χρησιμοποιούνταν ένα πλήθος σημάτων μεταξύ των διεργασιών, γεγονός που πιθανόν δημιουργούσε ανεπιθύμητες καθυστερήσεις. Επιπλέον σε αυτές τις συνθήκες υψηλού φόρτου το σύστημα εμφάνιζε συχνά φαινόμενα αστάθειας που αποδόθηκαν τελικά στην μεγάλη χρήση των σημάτων διακοπής που προκαλούσαν αλλαγή στην κανονική εκτέλεση των διεργασιών. Όπως έχει αναφερθεί στην προηγούμενη ενότητα σήματα χρονικού συναγερμού χρησιμοποιούνται για την λειτουργία του ελεγκτή χρονικά ευαίσθητων δεδομένων και για την επείγουσα διακοπή της κανονικής λειτουργίας των διεργασιών προσπέλασης δίσκου, προκειμένου να ικανοποιήσουν τις προθεσμίες των δεδομένων ροής. Ιδιαίτερα για την τελευταία περίπτωση αποδείχθηκε ότι ήταν αδύνατο να διακοπεί η διαδικασία του συστήματος για ανάκτηση

δεδομένων από συσκευή, εφόσον αυτή είχε ήδη κληθεί. Συνεπώς το σήμα επείγουσας διακοπής λειτουργίας της διεργασίας προσπέλασης δίσκου, γινόταν αντιληπτό από τη διεργασία μετά το πέρας της τρέχουσας εντολής προσπέλασης δεδομένων, γεγονός που ευνόητα μπορούσε να δημιουργήσει συνεχόμενες καθυστερήσεις. Αλλαγή χρειαζόταν επίσης και στην βασική πολιτική δρομολόγησης καθώς είναι γενικά παραδεκτό ότι δρομολόγηση προθεσμιών (EDF) δεν μπορεί να προσφέρει την καλύτερη απόδοση. Επιπλέον η ακρίβεια των μετρητών χρόνου που χρησιμοποιούνταν για την αποστολή σημάτων χρονικού συναγερμού ήταν αμφισβητήσιμη.

Όταν είχε αρχίσει η ανάπτυξη του εξυπηρετητή αποθήκευσης και αργότερα του μηχανισμού δρομολόγησης η πολυνηματική τεχνολογία δεν είχε επικρατήσει ακόμη στα λειτουργικά συστήματα. Έτσι στην ανάπτυξη του εξυπηρετητή και για την δυνατότητα επεξεργασίας πολλαπλών αιτήσεων χρησιμοποιήθηκαν μηχανές καταστάσεων με χρήση ουρών, προκειμένου να πετύχουμε την ψευδο-ταυτόχρονη εκτέλεση των αιτήσεων και άλλων εργασιών. Επιπρόσθετα η αναπόφευκτη χρήση τεχνικών busy waiting είχε όπως ήταν φυσικό αρνητική επίδραση στην προσπάθεια βελτιστοποίησης της χρήσης του επεξεργαστή.



Σχήμα 23: Ο Kydonia-D εξυπηρετητής

Στο σχήμα διακρίνεται η υπηρεσία αποθήκευσης δεδομένων ροής της διατριβής. Η υπηρεσία αυτή αναλαμβάνει εξ' ολοκλήρου την διαχείριση των δεδομένων ροής από την ανάκτησή τους μέχρι την αποστολή δεδομένων στους πελάτες.

Ένα από τα σημαντικότερα προβλήματα της αρχικού σχεδιασμού και υλοποίησης ήταν η διαμεσολάβηση του υποσυστήματος διαχείρισης αποθήκευσης για την αποστολή

δεδομένων, τα οποία ανακτώνταν από τις διεργασίες αποθήκευσης, ενημερωνόταν ο δρομολογητής και τελικά η υπεύθυνη διεργασία αποστολής. Σε αυτήν τη ροή δεδομένων όπως αποδίδεται γραφικά στο Σχήμα 22 είναι ευνόητο ότι υπεισέρχονταν ανεπιθύμητες καθυστερήσεις και τελικά η διεργασία δρομολόγησης αποτελούσε σημείο συμφόρησης (bottleneck).

Όλα τα παραπάνω αρνητικά σημεία, έπρεπε συνεπώς να αντιμετωπιστούν εφόσον ο απώτερος στόχος ήταν η ανάπτυξη ενός πληροφορικού συστήματος εξυπηρέτησης πολυμέσων μεγάλης κλίμακας. Έτσι στα πλαίσια της μεταπτυχιακής διατριβής χρειάστηκε μια ριζική αλλαγή στο σχεδιασμό και την ανάπτυξη του συστήματος. Έπρεπε να διαχωριστεί το τμήμα που ήταν υπεύθυνο για την διαχείριση των δεδομένων συνεχόμενης ροής, εφόσον το μέγεθος των πολυμέσων είναι μεγάλο και δικαιολογεί την δέσμευση δίσκων αποκλειστικά για την αποθήκευσή τους, αλλά και δίνει μεγαλύτερη ευελιξία για την υποστήριξη μηχανισμών κλιμάκωσης του εξυπηρετητή. Έπρεπε επίσης να ανασχεδιαστεί ο μηχανισμός δρομολόγησης και να δοθεί ιδιαίτερη προσοχή στον τρόπο και στις τεχνικές υλοποίησης. Αυτό το νέο τμήμα αποτελεί την υπηρεσία αποθήκευσης ροής (stream storage service) που αναλαμβάνει την αποθήκευση των δεδομένων ροής, περιλαμβάνει τον μηχανισμό δρομολόγησης και τμήματα λογισμικού για την υποστήριξή του. Η υπηρεσία ροής αναδεικνύεται στο πιο σημαντικό κομμάτι του εξυπηρετητή αποθήκευσης για την υποστήριξη εφαρμογών πολυμέσων όπως video κατ' απαίτηση (Σχήμα 23). Το μέγεθος των δεδομένων και οι υψηλοί ρυθμοί μετάδοσής τους, προϋποθέτουν δρομολόγηση πραγματικού χρόνου με συντονισμό των αποθηκευτικών μέσων, επιλογή της πολιτικής τοποθέτησης δεδομένων, άμεση και κατάλληλη διοχέτευση των δεδομένων στο δίκτυο. Λόγω των αυξημένων απαιτήσεων πρέπει να χρησιμοποιηθούν οι κατάλληλοι τρόποι και τεχνικές ανάπτυξης λογισμικού για την βελτιστοποίηση των πόρων ενός συστήματος και αύξηση της παραγωγικότητας.

Ο σχεδιασμός της υπηρεσίας ροής έγινε με στόχο να δημιουργηθεί ένα πλήρες μοντέλο ανάπτυξης ενός εξυπηρετητή δεδομένων ροής (stream server) με έμφαση στο τεχνικό μέρος. Με την ανάπτυξη αυτής της υπηρεσίας ο εξυπηρετητής είναι σε θέση να επεξεργάζεται αιτήσεις πολλών πελατών για ανάκτηση πολυμέσων, από ένα πλήθος αποθηκευτικών μέσων, με ένα αποδοτικό μοντέλο δρομολόγησης βασισμένο σε κύκλους εξυπηρέτησης και δυνατότητες αλληλεπίδρασης. Ο δρομολογητής έχει την ανάγκη συνεργασίας και με άλλα τμήματα λογισμικού που αναλαμβάνουν ειδικές εργασίες (π.χ. διαχειριστής αποθηκευτικών μέσων, ελεγκτής υποβολής αιτήσεων κτλ.). Ο σχεδιασμός προϋποθέτει να στηριχθεί η ανάπτυξη του λογισμικού σε προγραμματισμό πραγματικού χρόνου, πολυνηματική επεξεργασία, κατάλληλη διαχείριση μνήμης, συντονισμό και έλεγχο των αποθηκευτικών μέσων, ώστε να έχουμε εγγυήσεις σωστής εξυπηρέτησης των αιτήσεων αλλά και βέλτιστη χρήση των πόρων του συστήματος (CPU, μνήμη, δίκτυο) για την καλύτερη απόδοση.

3.6 Περίληψη

Στο κεφάλαιο αυτό παρουσιάστηκε το πληροφορικό σύστημα Kydonia, πάνω στο οποίο βασίστηκαν οι εργασίες της μεταπτυχιακής διατριβής. Τα κύρια χαρακτηριστικά του συστήματος είναι η διαχείριση της αποθήκευσης δεδομένων πολυμέσων, μηχανισμοί δρομολόγησης, μετάδοση δεδομένων σε ετερογενή δίκτυα, λειτουργικότητα βάσεων δεδομένων, μοντελοποίηση πολυμέσων και μέθοδοι προσπέλασης και πλοήγησης σε κείμενο και video. Η αρχιτεκτονική του συστήματος ακολουθεί το σχήμα πελάτη-εξυπηρετητή και αποτελείται από τα υποσυστήματα υποστήριξης του πελάτη, επικοινωνίας, διαχείρισης αντικειμένων και συνόδων και αποθήκευσης δεδομένων.

Ο εξυπηρετητής Kydonia ενσωματώνει τρία επίπεδα δρομολογητών. Στο υποσύστημα αποθήκευσης εφαρμόζεται ο δρομολογητής αιτήσεων, ο οποίος έχει σαν στόχο την αποδοτική διαχείριση της πρόσβασης στην δευτερεύουσα μνήμη και την εξυπηρέτηση των αιτήσεων με περιορισμούς πραγματικού χρόνου.

Στα πλαίσια του ερευνητικού προγράμματος SICMA, έλαβαν χώρα οι απαραίτητες ενέργειες και μετατροπές στον εξυπηρετητή Kydonia, ώστε να ακολουθήσει το πρότυπο DAVIC ([DAV95]). Το πρότυπο DAVIC καθορίζει το πλαίσιο με το οποίο παρέχονται υπηρεσίες πολυμέσων από μεγάλης κλίμακας εξυπηρετητές, μέσω συστημάτων διανομής, στην πλευρά των πελατών (Set Top Units). Ο κύριος στόχος είναι η επίτευξη της διαλειτουργικότητας ανάμεσα σε διαφορετικές οντότητες και η ολοκλήρωση των οπτικο-ακουστικών εφαρμογών και υπηρεσιών.

Η προσπάθεια και οι εργασίες που έλαβαν χώρα, ώστε ο εξυπηρετητής Kydonia να γίνει συμβατός με το πρότυπο DAVIC, ενίσχυσαν το σύστημα με νέα χαρακτηριστικά (όπως νέα πρωτόκολλα και συστήματα ζεύξης), ενώ παράλληλα οδήγησαν στην ενσωμάτωση νέων μηχανισμών για την αποδοτικότερη διαχείριση των δεδομένων πολυμέσων. Ιδιαίτερα στον εξυπηρετητή αποθήκευσης, ο οποίος τελικά είναι ο αποδέκτης όλου του φόρτου διαχείρισης των πολυμέσων, έγινε ουσιαστική έρευνα και πολλοί πειραματισμοί.

Το αρχικό σύστημα Kydonia αποτελούνταν από ένα σύνολο διεργασιών, που αναλάμβαναν την λειτουργία των διαφορετικών υποσυστημάτων. Η επικοινωνία μεταξύ των υποσυστημάτων επιτυγχανόταν με τεχνικές IPC. Ο δρομολογητής αιτήσεων του υποσυστήματος αποθήκευσης αποτελούσε μια μηχανή καταστάσεων που έτρεχε διαρκώς και υποστηριζόταν από πολλαπλές δομές ουρών, με στόχο την επίτευξη της ψευδο-ταυτόχρονης επεξεργασίας των αιτήσεων. Ιδιαίτερα για την εξυπηρέτηση των αιτήσεων συνεχόμενης ροής επιβάλλονταν η πολιτική δρομολόγησης προθεσμιών (EDF), με χρήση ενός μηχανισμού χρονικών σημάτων διακοπής.

Η εφαρμογή της παραπάνω αρχιτεκτονικής σε πραγματικές εφαρμογές, εμφάνισε προβλήματα απόδοσης, τα οποία αποδόθηκαν στο μεγάλο κόστος επικοινωνίας των διεργασιών, την πολιτική δρομολόγησης EDF και τον βαρύ μηχανισμό δρομολόγησης με την βοήθεια πολλαπλών ουρών. Επίσης η χρήση μεγάλου αριθμού σημάτων διακοπής σε πραγματικές συνθήκες εμφάνιζε προβλήματα αστάθειας.

Οι εργασίες για την βελτίωση του εξυπηρετητή δεδομένων συνεχόμενης ροής και τον ανασχεδιασμό του μηχανισμού δρομολόγησης, αποτελεί το υλικό της παρούσας μεταπτυχιακής διατριβής. Ο νέος σχεδιασμός της υπηρεσίας διαχείρισης δεδομένων ροής, που περιγράφεται στα επόμενα κεφάλαια, προϋποθέτει ανάπτυξη λογισμικού με χρήση της πολυνηματικής τεχνολογίας, προγραμματισμό πραγματικού χρόνου, κατάλληλη διαχείριση μνήμης, συντονισμό και έλεγχο των εργασιών ανάκτησης και μετάδοσης δεδομένων, ώστε να προκύψει ένα σύστημα που να μπορεί να προσφέρει εγγυήσεις σωστής εξυπηρέτησης των αιτήσεων και βέλτιστη αξιοποίηση των πόρων του συστήματος για την καλύτερη απόδοση.

➤ **Ανάπτυξη υποσυστήματος υποστήριξης της υπηρεσίας διαχείρισης δεδομένων ροής**



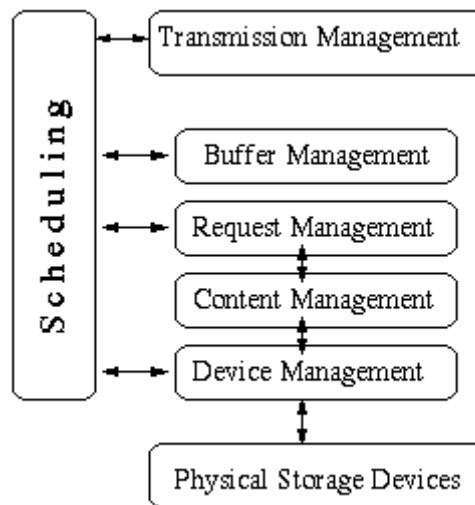
Σε αυτό το κεφάλαιο γίνεται περιγραφή όλου του μηχανισμού και των δομών υποστήριξης της υπηρεσίας διαχείρισης δεδομένων ροής στον εξυπηρετητή αποθήκευσης. Γίνεται λεπτομερή αναφορά στα τμήματα λογισμικού που αναλαμβάνουν την διαχείριση των μέσων αποθήκευσης, την οργάνωση των αντικειμένων, την προσωρινή αποταμίευση των δεδομένων και τέλος την μετάδοση των δεδομένων στους πελάτες. Τα τμήματα αυτά λογισμικού σχεδιάστηκαν και αναπτύχθηκαν ειδικά για την αντιμετώπιση των δεδομένων συνεχόμενης ροής και υποστηρίζουν τον μηχανισμό δρομολόγησης της υπηρεσίας ροής που περιγράφεται στο 6ο κεφάλαιο.

4.1 Ανάγκες της υπηρεσίας διαχείρισης δεδομένων ροής

Τα παραδοσιακά συστήματα αρχείων και οι εξυπηρετητές αποθήκευσης σχεδιάστηκαν και αναπτύχθηκαν για την διαχείριση της αποθήκευσης και ανάκτησης των συνηθισμένων παραδοσιακών δεδομένων, όπως κείμενο, αριθμητικά δεδομένα και δυαδικά αρχεία. Τα συστήματα αυτά αναλαμβάνουν την εξυπηρέτηση δεδομένων συνεχόμενης ροής με παρόμοιο τρόπο, με αποτέλεσμα να μην παρέχουν εγγυήσεις στον ρυθμό ανάκτησης δεδομένων και συνεπώς να εμφανίζονται προβλήματα στην παρουσίαση αυτών των δεδομένων. Έτσι η ανάπτυξη ενός συστήματος λογισμικού για την διαχείριση δεδομένων ροής προϋποθέτει τον σχεδιασμό ενός συστήματος αποθήκευσης που θα λαμβάνει υπόψη του τα χαρακτηριστικά των χρονικά ευαίσθητων δεδομένων και θα εγγυάται την ικανοποίηση των απαιτήσεών τους.

Επιπρόσθετα ένα σύστημα διαχείρισης πολυμέσων χρειάζεται να έχει τον έλεγχο για ένα μεγάλο αριθμό από διαφορετικές εργασίες. Οι εργασίες αυτές αναφέρονται στην διαχείριση περιεχομένου (δηλαδή στην οργάνωση της αποθήκευσης και τοποθέτησης των αντικειμένων ροής), στην διαχείριση της προσωρινής αποταμίευσης των δεδομένων και τέλος στους μηχανισμούς μετάδοσης των ανακτημένων δεδομένων στους αιτούμενους πελάτες. Τα διαφορετικά αυτά επίπεδα εργασιών περιγράφονται γραφικά στο Σχήμα 24. Στο υπόλοιπο του κεφαλαίου θα γίνει αναφορά στα υποσυστήματα λογισμικού που

αναπτύχθηκαν για να καλύψουν τις ανάγκες όλων των επιπέδων εργασιών του σχήματος και συνιστούν βασικό κομμάτι υποστήριξης της υπηρεσίας διαχείρισης δεδομένων ροής.



Σχήμα 24: Τα επίπεδα διαχείρισης εργασιών της υπηρεσίας ροής

Στο παραπάνω σχήμα γίνεται μια κατηγοριοποίηση των εργασιών που καλείται να διαχειριστεί η υπηρεσία δεδομένων ροής. Υπάρχει ανάγκη για ανάπτυξη λογισμικού για την διαχείριση των συσκευών αποθήκευσης και του περιεχομένου σύμφωνα με τα ιδιαίτερα χαρακτηριστικά των δεδομένων ροής, την επεξεργασία των αιτήσεων των πελατών, την προσωρινή αποταμίευση δεδομένων και τέλος την μετάδοση των δεδομένων στους πελάτες. Όλες αυτές οι εργασίες πρέπει να υποστηρίζονται από ένα αποδοτικό μηχανισμό δρομολόγησης.

Η πρόκληση για την υπηρεσία ροής, είναι ο σχεδιασμός και η ανάπτυξη ενός ολοκληρωμένου μηχανισμού δρομολόγησης, ο οποίος θα οδηγεί και θα συντονίζει τις εργασίες όλων των επιπέδων και επιπρόσθετα θα παρέχει εγγυήσεις πραγματικού χρόνου για την ικανοποίηση των απαιτήσεων παρουσίασης των δεδομένων ροής. Αυτός ο μηχανισμός περιγράφεται αναλυτικά στο 6ο κεφάλαιο.

4.2 Αποθήκευση δεδομένων ροής

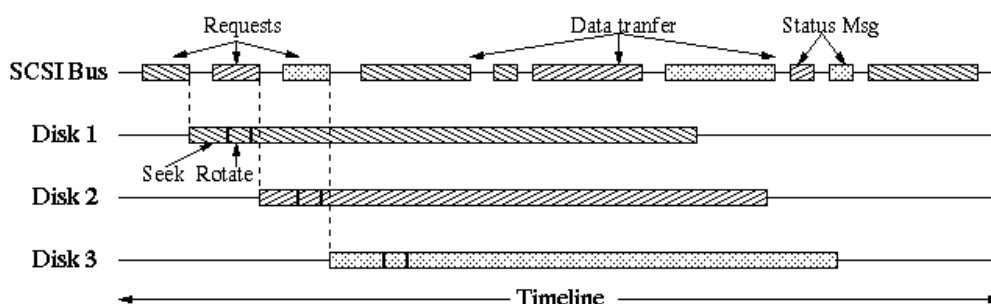
4.2.1 Φυσικό επίπεδο αποθήκευσης

Η τεχνολογία των μαγνητικών δίσκων έχει εμφανίσει εντυπωσιακές εξελίξεις τις τελευταίες δεκαετίες και έχουν επικρατήσει ως η κύρια επιλογή για την μόνιμη αποθήκευση ψηφιακών δεδομένων. Τα λειτουργικά συστήματα παρέχουν στον προγραμματιστή μια αφηρημένη εικόνα, ότι ο κάθε δίσκος αποτελεί ένα γραμμικό πίνακα από σταθερού μεγέθους κομμάτια (media blocks). Ο μηχανισμός αυτός απλοποιεί την ανάπτυξη εφαρμογών, απομακρύνοντας τον προγραμματιστή από λεπτομέρειες φυσικού επιπέδου. Η υποστήριξη όμως δεδομένων πολυμέσων επηρεάζεται σε μεγάλο βαθμό από τις διακυμάνσεις στον χρόνο εξυπηρέτησης των αιτήσεων ανάκτησης δεδομένων από τους δίσκους, καθώς πρέπει να ακολουθεί περιορισμούς πραγματικού χρόνου.

Η εφαρμογή της υπηρεσίας διαχείρισης δεδομένων ροής έγινε πάνω σε μαγνητικούς δίσκους τεχνολογίας SCSI. Το πρότυπο SCSI αποτελεί ένα καθορισμό διαύλου (bus

specification) αλλά και καθορισμό εντολών (command specification) για την αποδοτική χρήση του διαύλου. Όπως φαίνεται στο Σχήμα 4 (κεφ. 2) προσαρμοστής SCSI (host adapter) διασυνδέει τον δίαυλο του συστήματος με τον δίαυλο SCSI. Οι συσκευές αποθήκευσης σε αυτή την αρχιτεκτονική καλούνται στόχοι (targets) και συνδέονται πάνω στον δίαυλο.

Κύριος στόχος του προτύπου SCSI είναι η παροχή διαφάνειας στην χρήση περιφερειακών συσκευών χωρίς να είναι απαραίτητη η γνώση των φυσικών τους χαρακτηριστικών. Οι συσκευές και ο προσαρμοστής διαύλου κατηγοριοποιούνται σε τρία επίπεδα. Τα επίπεδα αυτά προσφέρουν διαφορετικές επιδόσεις και είναι: FAST (10 Mbytes/sec), FAST&WIDE (20 Mbytes/sec), ULTRA (40 Mbytes/sec). Το μονοπάτι δεδομένων για ένα FAST SCSI δίαυλο είναι 1 byte και ο μηχανισμός διευθυνσιοδότησης επιτρέπει την χρήση μέχρι οκτώ συσκευών συμπεριλαμβανομένου και του προσαρμοστή διαύλου. Κάθε συσκευή έχει σταθερή προτεραιότητα, που βρίσκεται σε γραμμική αντιστοιχία με το αναγνωριστικό στόχου (target id) της συσκευής. Ο τελευταίος αριθμός κυμαίνεται από 0 έως 7. Ο FAST/WIDE SCSI δίαυλος μπορεί να μεταφέρει ταυτόχρονα δύο byte δεδομένων και μπορεί να υποστηρίξει μέχρι 16 συσκευές. Επόμενο είναι ότι σε αυτήν την περίπτωση το εύρος μετάδοσης δεδομένων είναι διπλάσιο εξαιτίας του ευρύτερου φυσικού μονοπατιού δεδομένων.



Σχήμα 25: Ο Τρόπος χρήσης του κοινού διαύλου στο πρότυπο SCSI

Η χρήση του κοινού διαύλου από περισσότερες από μια συσκευές επιτυγχάνεται λόγω του σχήματος σύνδεσης-αποσύνδεσης που ακολουθεί το πρότυπο SCSI. Σύμφωνα με αυτό το σχήμα, όταν μια συσκευή δίσκου δέχεται μια εντολή μέσω του διαύλου, τότε αποσυνδέεται από αυτόν και αρχίζει να εκτελεί την κατάλληλη διαδικασία αναζήτησης (μετακίνησης της κεφαλής) εσωτερικά. Μόλις η κεφαλή τοποθετηθεί στην σωστή θέση, η συσκευή περιμένει να βρεθούν τα κατάλληλα δεδομένα κάτω από την κεφαλή με την περιστροφή του δίσκου. Καθώς αρχίζει η ανάκτηση των δεδομένων η συσκευή δεν συνδέεται άμεσα με τον δίαυλο, αλλά χρησιμοποιεί ένα μικρό αποταμιευτή για την προσωρινή διατήρηση των ανακτημένων δεδομένων. Όταν ο αποταμιευτής αυτός γεμίσει τότε γίνεται η επανασύνδεση με τον δίαυλο για την μετάδοση των δεδομένων. Η μετάδοση λαμβάνει χώρα εκρηκτικά (burst transmission), εκμεταλλευόμενη ολόκληρο το εύρος μετάδοσης του διαύλου.

Ακολουθώντας τον παραπάνω μηχανισμό μια συσκευή μπορεί να εργάζεται συνεχώς εκτελώντας εντολές ανάκτησης δεδομένων μεγάλου μεγέθους, αλλά παραμένει συνδεδεμένη με τον δίαυλο μόνο κατά ένα ποσοστό του συνολικού χρόνου. Το γεγονός αυτό επιτρέπει και σε άλλες συσκευές να χρησιμοποιούν τον ίδιο δίαυλο για την πολύπλεξη των εντολών αλλά και των δεδομένων που μεταφέρονται. Ωστόσο όμως, όπως αναφέρεται στην βιβλιογραφία, αλλά και εύκολα μπορεί να διαπιστωθεί πειραματικά, η επεκτασιμότητα της αρχιτεκτονικής SCSI περιορίζεται χαμηλότερα από τα επίπεδα που θεωρητικά προκύπτουν. Ιδιαίτερα λόγω της χρήσης προτεραιοτήτων, η εφαρμογή μεγάλου αριθμού δίσκων πάνω στον ίδιο δίαυλο, γρήγορα έχει σαν αποτέλεσμα να εμφανίζεται ο δίσκος με την μικρότερη προτεραιότητα να έχει και την χειρότερη απόδοση.

4.2.2 Λειτουργίες υποστήριξης των μέσων αποθήκευσης

Το λειτουργικό σύστημα UNIX αποτελεί ένα σύστημα γενικού σκοπού με μεγάλη ποικιλία υλοποιήσεων (Solaris, HP-UX, Linux κ.α.). Εφόσον η προσπέλαση της πρωτεύουσας μνήμης είναι τάξεις μεγέθους ταχύτερη από αυτήν της δευτερεύουσας, στο UNIX υπάρχει ένας μηχανισμός ο οποίος διατηρεί σε κατάλληλη δομή της κύριας μνήμης σελίδες δεδομένων που έχουν πρόσφατα ανακτηθεί από τα μέσα αποθήκευσης, με απώτερο στόχο την ελαχιστοποίηση των προσπελάσεων των μέσων αποθήκευσης.

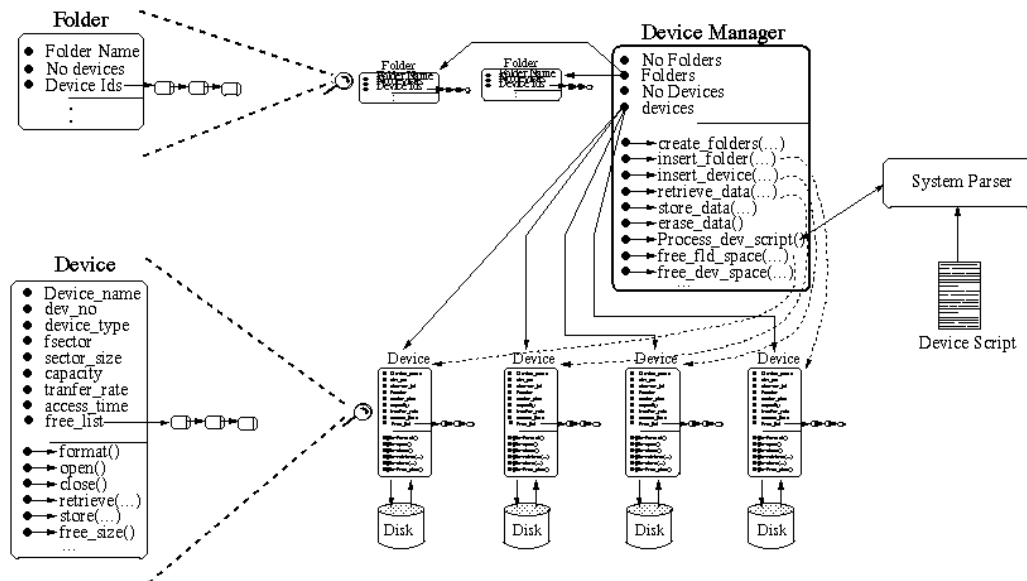
Αν και ο παραπάνω μηχανισμός είναι ιδιαίτερα χρήσιμος για την υποστήριξη παραδοσιακών εφαρμογών, εντούτοις δεν είναι κατάλληλος για την ανάπτυξη της υπηρεσίας ροής, ο σχεδιασμός της οποίας προϋποθέτει τον πλήρη έλεγχο της πρόσβασης σε κάθε επίπεδο μνήμης, ώστε να μπορέσει να δρομολογήσει σωστά και σε πραγματικό χρόνο την εξυπηρέτηση των αιτήσεων ανάκτησης πολυμέσων. Η υπηρεσία ροής χρειάζεται επομένως έναν μηχανισμό πρόσβασης των μέσων αποθήκευσης, ο οποίος θα είναι ανεξάρτητος των περιορισμών του λειτουργικού συστήματος και του υλοποιημένου συστήματος αρχείων, με στόχο την εφαρμογή πιο κατάλληλων πολιτικών αποταμίευσης αλλά και μηχανισμών οργάνωσης της αποθήκευσης πολυμέσων.

Το λειτουργικό σύστημα Solaris παρέχει ένα σταθερό σύστημα χρήσης των περιφερειακών συσκευών, μέσω της αφαίρεσης σε επίπεδο αρχείων. Η αφαίρεση αυτή επιτρέπει την διαχείριση των συσκευών με την κλήση των βασικών λειτουργιών (άνοιγμα, κλείσιμο, διάβασμα, γράψιμο) των αρχείων. Επιπρόσθετα στο λειτουργικό σύστημα παρέχονται τα τμήματα λογισμικού που αποτελούν τους οδηγούς των συσκευών (device drivers) για την υλοποίηση της παραπάνω αφαίρεσης.

Η πρόσβαση στα αρχεία γίνεται με δύο μεθόδους, είτε με χρήση του block interface που υιοθετεί μηχανισμούς προσωρινής αποταμίευσης, είτε μέσω του raw interface, που παρακάμπτει τους προηγούμενους μηχανισμούς. Η ανάπτυξη της υπηρεσίας ροής στηρίζεται πάνω στο raw interface για λόγους βελτιστοποίησης, το οποίο επιτρέπει την ταχύτερη λειτουργία των δίσκων και αποφεύγει τους περιορισμούς του παραδοσιακού συστήματος αρχείων του λειτουργικού συστήματος.

4.2.3 Ο διαχειριστής των μέσων αποθήκευσης

Στο Σχήμα 26 διακρίνεται η αρχιτεκτονική του λογισμικού για την υποστήριξη των μέσων αποθήκευσης που διαχειρίζεται η υπηρεσία δεδομένων συνεχόμενης ροής. Η κλάση Device αναλαμβάνει την διαχείριση της πληροφορίας που σχετίζεται με ένα μέσο αποθήκευσης. Η πληροφορία αυτή αναφέρεται στα τεχνικά χαρακτηριστικά της συσκευής αποθήκευσης, μερικά από τα οποία εμφανίζονται στο σχήμα. Όπως αναφέρθηκε στην προηγούμενη ενότητα η υπηρεσία ροής παρακάμπτει το σύστημα αρχείων του λειτουργικού συστήματος και προσπελαίνει άμεσα την συσκευή δίσκου (raw device interface). Ακολουθώντας την αφαίρεση που παρέχουν οι οδηγοί συσκευών του λειτουργικού, κάθε δίσκος αντιμετωπίζεται σαν ένας γραμμικός χώρος αποθήκευσης, η πρόσβαση στον οποίο επιτυγχάνεται με λογικές διευθύνσεις. Τα στιγμιότυπα της κλάσης Device επιτρέπουν ένα αριθμό από λειτουργίες πάνω στα μέσα αποθήκευσης, ενώ κάθε στιγμιότυπο διατηρεί μια λίστα, οι κόμβοι της οποίας αντιπροσωπεύουν ελεύθερα τμήματα στον αποθηκευτικό χώρο της συσκευής. Κάθε φορά που μια λειτουργία εγγραφής ή διαγραφής δεδομένων εκτελείται, το αντίστοιχο στιγμιότυπο της κλάσης Device συμβουλευέται και ενημερώνει την λίστα των ελεύθερων τμημάτων. Η διαδικασία εκχώρησης τμήματος χώρου για την αποθήκευση δεδομένων ακολουθεί την πολιτική αναζήτησης χώρου στο ελεύθερο τμήμα που είναι πιο κοντά στο μέγεθος εκχώρησης που απαιτείται (πολιτική best-fit).



Σχήμα 26: Υποσύστημα υποστήριξης των μέσων αποθήκευσης

Ο διαχειριστής των μέσων αποθήκευσης (Device Manager) αποτελεί το κεντρικό αντικείμενο το οποίο είναι υπεύθυνο για την οργάνωση και την προσπέλαση των συσκευών αποθήκευσης. Μέσω του διαχειριστή συσκευών και ενός γενικού συνόλου λειτουργιών επιτρέπεται η πρόσβαση και η διαχείριση όλων των αποθηκευτικών μέσων με βάση το αναγνωριστικό τους. Η κεντρική αυτή οντότητα αποτελεί ένα από τα βασικά τμήματα λογισμικού που όπως περιγράφεται στο κεφάλαιο 6 είναι απαραίτητο για την υποστήριξη του δρομολογητή και των εξυπηρετητών δίσκου. Μερικές από τις λειτουργίες που παρέχονται από τον διαχειριστή συσκευών εμφανίζονται στο Σχήμα 26.

Ο διαχειριστής συσκευών καθορίζει επίσης και τον διαχωρισμό ή την συνένωση διαφορετικών δίσκων σε λογικό επίπεδο, σε μια προσπάθεια οργάνωσης του συνολικού χώρου αποθήκευσης. Με αυτόν τον τρόπο εισάγεται η έννοια του φακέλου αντικειμένων (Object Folder) με την οποία επιτυγχάνεται η ομαδοποίηση των δίσκων και προαιρετικά η δέσμευση χώρου σε αυτούς. Κάθε φάκελος φέρει ένα μοναδικό όνομα και στην γενική περίπτωση αντιπροσωπεύεται με ένα αντικείμενο. Το αντικείμενο αυτό έχει παραμέτρους το όνομα του φακέλου, τον αριθμό των αποθηκευτικών μέσων που καλύπτει, καθώς και μια λίστα με τα αναγνωριστικά των συσκευών. Η ανάπτυξη του μηχανισμού αυτού οργάνωσης του αποθηκευτικού χώρου είναι πολύ σημαντική και δίνει την δυνατότητα στον διαχειριστή του συστήματος (system administrator) να καθορίσει σε υψηλό επίπεδο τον τρόπο αποθήκευσης των αντικειμένων, θέτοντας όρια στην διασπορά της αποθήκευσης σε πολλαπλούς δίσκους, αλλά και δεσμεύοντας ομάδες δίσκων είτε για την αποθήκευση αντικειμένων με ιδιαίτερα χαρακτηριστικά, είτε για λόγους διαφοροποίησης ετερογενών δίσκων.

Από τα προηγούμενα γίνεται εύκολα κατανοητό ότι είναι δυνατόν οι ίδιοι δίσκοι να συνυπάρχουν στα πλαίσια πολλών φακέλων χωρίς κανένα περιορισμό. Επειδή όμως μπορεί να υπάρξουν περιπτώσεις στις οποίες αυτό το γεγονός να μην είναι θεμιτό, παρέχεται επιπρόσθετα ο μηχανισμός δέσμευσης χώρου στον δίσκο για συγκεκριμένο φάκελο. Έτσι ο φάκελος δε καθορίζει πλέον μόνο τους δίσκους στους οποίους επιτρέπεται η αποθήκευση των αντικειμένων που ανήκουν σε αυτόν, αλλά και το μέγεθος και την θέση του χώρου που πρέπει να δεσμευθεί σε κάθε δίσκο.

Μια από τις πιο σημαντικές λειτουργίες του διαχειριστή συσκευών είναι αυτή της επεξεργασίας ενός αρχείου που περιέχει δηλώσεις σε μια αυστηρά καθορισμένη γλώσσα (script). Ο διαχειριστής συνεργάζεται με τον αναλυτή του συστήματος (parser), ο οποίος δέχεται σαν είσοδο το αρχείο script, το οποίο ενημερώνει το σύστημα για τις συσκευές που βρίσκονται συνδεδεμένες προς χρήση από την υπηρεσία ροής και τους φακέλους αντικειμένων που πρέπει να δημιουργηθούν. Ο αναλυτής του συστήματος ύστερα από την κατάλληλη λεκτική και συντακτική ανάλυση εξάγει την χρήσιμη πληροφορία από το script και την μεταφέρει στον διαχειριστή συσκευών μέσω κατάλληλων δομών δεδομένων. Ο τελευταίος αξιοποιεί αυτήν την πληροφορία και αναλαμβάνει την εκτέλεση των απαραίτητων ενεργειών, όπως είναι η αναγνώριση των δίσκων, η προσαρμογή τους στο σύστημα αποθήκευσης της υπηρεσίας και η δημιουργία των φακέλων. Η επεξεργασία του script αρχείου λαμβάνει χώρα κάθε φορά που το σύστημα επανακινείται, ώστε να διαπιστώνονται τυχόν αλλαγές που συνέβηκαν στο υλικό του υπολογιστικού συστήματος (εισαγωγή νέων δίσκων, αφαίρεση δίσκων, δημιουργία νέου φακέλου κτλ.). Στην συνέχεια ακολουθεί ένα παράδειγμα ενός script αρχείου, το οποίο ενημερώνει την υπηρεσία διαχείρισης δεδομένων ροής ότι υπάρχουν διαθέσιμοι τρεις δίσκοι για την αποθήκευση δεδομένων και καθορίζει τους φακέλους αντικειμένων που πρέπει να δημιουργηθούν.

```
//Device & Folder Information file
STORAGE DEVICES {
device_id:                dev1
device_type:              MAGNETIC
device_name:              "/dev/rdsk/c0t1d0s0"
```

```

capacity:          999 MB
first_sector:      3
sector_size:       512 B
trans_rate:        4000 KB/sec
avg_access:        10 msec
contl_id           0

device_id:         dev2
dserver_id:        dserver1
device_type:       MAGNETIC
device_name:       "/dev/rdsk/c0t2d0s0"
capacity:          4095 MB
first_sector:      3
sector_size:       1024 B
trans_rate:        4000 KB/sec
avg_access:        11 msec
contl_id           0

device_id:         dev3
dserver_id:        dserver1
device_type:       MAGNETIC
device_name:       "/dev/rdsk/c1t1d0s0"
capacity:          4095 MB
first_sector:      3
sector_size:       1024 B
trans_rate:        4000 KB/sec
avg_access:        11 msec
contl_id           1

device_id:         dev4
dserver_id:        dserver1
device_type:       MAGNETIC
device_name:       "/dev/rdsk/c1t2d0s0"
capacity:          4095 MB
first_sector:      3
sector_size:       1024 B
trans_rate:        4000 KB/sec
avg_access:        11 msec
contl_id           1
}
DSERVERS {
dserver1          LOCAL          3 disks
dserver2          REMOTE         147.27.1.6      3500      2 disks
}
FOLDERS {
Fold1             "system"        dev1
Fold2             "Video"         dev2 dev3
Fold3             "Audio"         dev4
}

```

4.2.4 Η αποθήκευση δεδομένων πολυμέσων

Η ψηφιοποίηση του ήχου και του video καταλήγει σε μια σειρά από δείγματα (samples) και εικόνες (frames) αντίστοιχα. Αυτή η σειρά καταγεγραμμένων εικόνων και δειγμάτων καλείται μέσο συνεχόμενης ροής (media stream). Έτσι τα ψηφιακά δεδομένα που αναπαριστούν ένα μέσο, αποκτούν σημασία όταν τα δεδομένα αυτά αναπαράγονται με συγκεκριμένες απαιτήσεις ρυθμού και συνεχόμενα σε πραγματικό χρόνο.

Ένα σύστημα αποθήκευσης δεδομένων συνεχόμενης ροής πρέπει να διαιρέσει τα συνολικά δεδομένα σε μικρότερα τμήματα (blocks), καθώς τα αποθηκεύει στους δίσκους. Κάθε τμήμα μπορεί να καλύψει αρκετά φυσικά τμήματα στον δίσκο (disk blocks). Σύμφωνα με την βιβλιογραφία στον τομέα αυτό, η αποθήκευση των τμημάτων αυτών μπορεί να γίνει είτε συνεχόμενα είτε διασκορπισμένα όσον αφορά την φυσική τους τοποθέτηση στα μέσα αποθήκευσης. Η συνεχόμενη αποθήκευση των τμημάτων έχει το πλεονέκτημα της εύκολης υλοποίησης, αλλά τα προβλήματα εσωτερικού κατακερματισμού (fragmentation) και του αυξημένου κόστους κατά την ενημέρωση, λόγω απαραίτητων αντιγραφών που γίνονται για να διατηρηθεί η συνέχεια, εμφανίζουν την τεχνική αυτή ως ακατάλληλη για περιβάλλοντα στα οποία συμβαίνουν συχνά αλλαγές στα δεδομένα. Από την άλλη πλευρά τοποθετήσεις που

επιτρέπουν την διασκορπισμένη αποθήκευση των τμημάτων, αποφεύγουν τα μειονεκτήματα της συνεχόμενης αντιμετώπισης.

Η πιο σημαντική διαφορά στις δύο αυτές τεχνικές τοποθέτησης είναι οι ενέργειες αναζήτησης της κεφαλής που πρέπει να λάβουν χώρα. Στην τεχνική συνεχόμενης τοποθέτησης εκτός της αρχικής μετακίνησης της κεφαλής όλα τα υπόλοιπα τμήματα δεδομένων μπορούν να ανακτηθούν χωρίς να μεσολαβούν επανατοποθετήσεις της κεφαλής. Αντίθετα στην δεύτερη αντιμετώπιση, είναι εμφανές ότι στην ανάκτηση κάθε τμήματος έχουμε αυξημένο κόστος λόγω των απαραίτητων μετακινήσεων της κεφαλής ανάμεσα στα διασκορπισμένα τμήματα. Ένας αποτελεσματικός τρόπος αντιμετώπισης αυτού του κόστους είναι να αυξήσουμε το μέγεθος των τμημάτων, ώστε να περιορίσουμε τις συνέπειες των ενεργειών αναζήτησης των δίσκων.

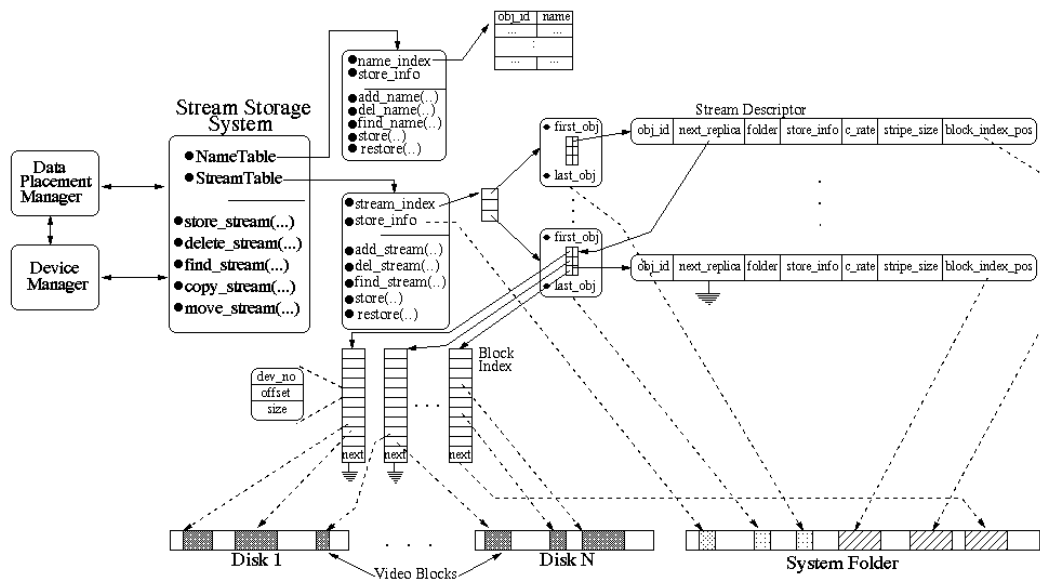
Η τεχνική διασκορπισμένης τοποθέτησης δεν περιορίζεται μόνο στα πλαίσια ενός δίσκου. Τα τμήματα που δημιουργούνται από την διαίρεση του αντικειμένου ροής μπορούν να αποθηκευτούν σε πολλαπλούς δίσκους, ακολουθώντας τεχνικές striping. Έτσι με αυτόν τον τρόπο μπορούμε να παρεμβάλλουμε (interleave) την αποθήκευση ενός μέσου ροής ανάμεσα σε πολλούς δίσκους. Η μονάδα παρεμβολής συνήθως είναι ένα τμήμα του αντικειμένου και αντιπροσωπεύει το μέγιστο μέγεθος λογικά συνεχόμενων δεδομένων που μπορούν να αποθηκευτούν σε συνεχόμενο φυσικό χώρο σε ένα δίσκο. Κάθε τμήμα του αντικειμένου ροής μπορεί να περιέχει σταθερό ή μεταβλητό αριθμό από μονάδες φυσικής αποθήκευσης. Αν τα δεδομένα συνεχόμενης ροής έχουν συμπιεστεί με τεχνικές μεταβλητού ρυθμού (variable bit rate – VBR), τότε οι απαιτήσεις σε χώρο αποθήκευσης θα διαφέρουν ανάμεσα στα τμήματα, με δεδομένο ότι η διαίρεση των τμημάτων γίνεται με τέτοιο τρόπο ώστε κάθε τμήμα να αντιπροσωπεύει σταθερό χρόνο αναπαραγωγής του μέσου. Από την άλλη μεριά, αν ένα σύστημα αποθήκευσης επιβάλλει σταθερού μεγέθους τμήματα αντικειμένων, τότε τα τμήματα αυτά αντιπροσωπεύουν μεταβλητό χρόνο αναπαραγωγής του μέσου.

Γίνεται επομένως εμφανές ότι τα συστήματα διαχείρισης αποθήκευσης δεδομένων πολυμέσων έχουν την δυνατότητα να επιλέξουν ανάμεσα σε δύο πολιτικές αποθήκευσης, την πολιτική τοποθέτησης μεταβλητού μεγέθους τμημάτων (variable-size block placement) και την πολιτική σταθερού μεγέθους τμημάτων (fixed-size block placement). Η επιλογή της πιο κατάλληλης στρατηγικής πρέπει να γίνει λαμβάνοντας υπόψη το περιβάλλον που πρέπει να υποστηρίξει το σύστημα αποθήκευσης και τις πολιτικές δρομολόγησης που θα ακολουθήσει. Και για τις δύο όμως τεχνικές υπάρχει το ίδιο ζήτημα, για το οποίο υπάρχουν αρκετές αναφορές στην βιβλιογραφία, και αναφέρεται στην βέλτιστη επιλογή του μεγέθους των τμημάτων, ώστε να αυξηθεί η συνολική παραγωγικότητα του συστήματος.

4.2.5 Το σύστημα αποθήκευσης αντικειμένων ροής

Το σύστημα αποθήκευσης αντικειμένων ροής αποτελεί ένα από τα πιο βασικά τμήματα λογισμικού που υποστηρίζουν την υπηρεσία διαχείρισης δεδομένων ροής. Το σύστημα αυτό αναλαμβάνει την οργάνωση της αποθήκευσης των αντικειμένων, δηλαδή στην ουσία υποστηρίζει ένα σύστημα αρχείων για αντικείμενα πολυμέσων. Στο Σχήμα 27 εμφανίζονται

γραφικά οι βασικές δομές υποστήριξης του συστήματος αποθήκευσης αντικειμένων ροής και αναπαριστάται ο τρόπος αποθήκευσης της πληροφορίας στα φυσικά μέσα.



Σχήμα 27: Το σύστημα αποθήκευσης αντικειμένων συνεχόμενης ροής

Ο διαχειριστής του συστήματος (system administrator) ορίζει πάντα ένα ειδικό φάκελο που ονομάζεται φάκελος του συστήματος, ο οποίος καθορίζει τον δίσκο ή τον χώρο σε δίσκους που θα χρησιμοποιηθούν για την αποθήκευση πληροφορίας του συστήματος, όπως είναι οι ειδικοί πίνακες δεικτοδότησης και πρόσβασης της δευτερεύουσας μνήμης. Το σύστημα αποθήκευσης διατηρεί δύο δομές, τον πίνακα ονομάτων (name table) και τον πίνακα αντικειμένων ροής (stream table). Με τον πίνακα ονομάτων η υπηρεσία αποκτά μεγαλύτερη ανεξαρτησία, που είναι θεμιτή σε ορισμένες περιπτώσεις και επιτρέπει την πρόσβαση στα αντικείμενα ροής που έχουν αποθηκευτεί στο σύστημα με βάση ένα μοναδικό όνομα. Πιο συγκεκριμένα ο πίνακας ονομάτων αποτελεί ένα πίνακα κατακερματισμού (hashing table), ο οποίος περιέχει την αντιστοιχία του ονόματος με το αναγνωριστικό του αντικειμένου.

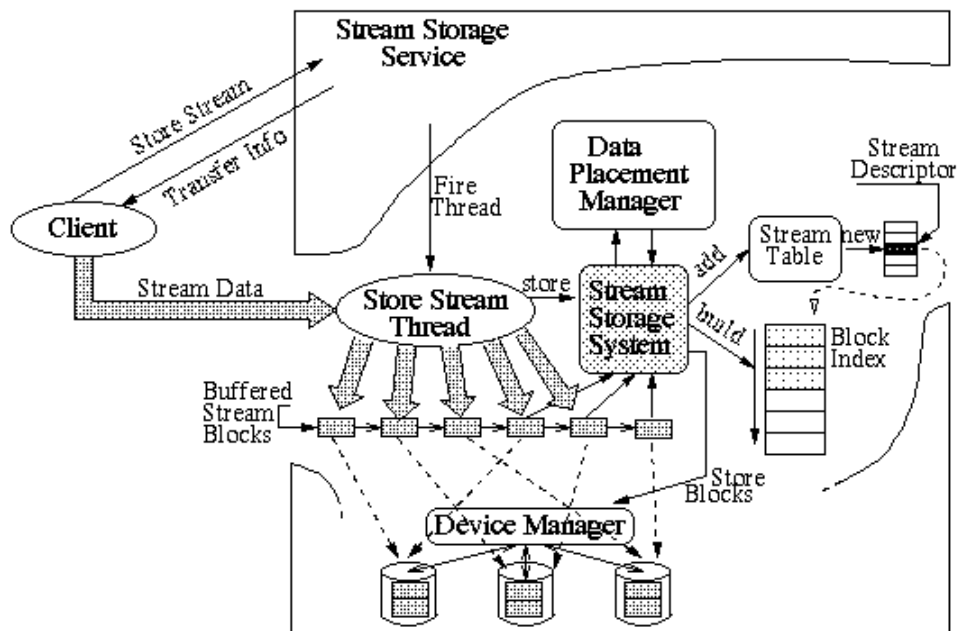
Το αναγνωριστικό του αντικειμένου είναι ένας ακέραιος αριθμός και αποτελεί το βασικό κλειδί για την εύρεση και την προσπέλαση ενός αποθηκευμένου αντικειμένου, η οποία γίνεται μέσω του πίνακα των αντικειμένων ροής. Ο πίνακας αυτός υλοποιεί ένα δέντρο που κρατά την πληροφορία αποθήκευσης των περιγραφών των αντικειμένων ροής. Το πρώτο επίπεδο του δέντρου βρίσκεται πάντα αποθηκευμένο σε σταθερή θέση στους πρώτους τομείς του δίσκου του συστήματος και ανακτάται στην κύρια μνήμη με την εκκίνηση της υπηρεσίας ροής. Οι κόμβοι του πρώτου επιπέδου αναπαριστούν τις θέσεις πάνω στον φάκελο του συστήματος που είναι αποθηκευμένα κομμάτια του πίνακα των αντικειμένων ροής. Κάθε κομμάτι περιέχει ένα μεγάλο αριθμό από εγγραφές (σε αύξουσα σειρά ως προς το αναγνωριστικό) και διακρίνεται από τον αριθμό του πρώτου και του τελευταίου αναγνωριστικού.

Κάθε εγγραφή περιέχει μια σειρά από παραμέτρους που όλες μαζί συνιστούν τον περιγραφέα αποθήκευσης ενός αντικειμένου και εμφανίζονται στο Σχήμα 27. Ένα πεδίο του περιγραφέα δείχνει την θέση που βρίσκεται αποθηκευμένος ο πίνακας πρόσβασης των τμημάτων του αντικειμένου (block-index). Το πεδίο αυτό υποδεικνύει δηλαδή στο σύστημα αποθήκευσης τον φυσικό δίσκο, τον αριθμό του αρχικού τομέα και το μέγεθος του πίνακα πρόσβασης. Ο πίνακας αυτός όταν αναπτυχθεί στην κύρια μνήμη ενημερώνει την υπηρεσία ροής για την τοποθέτηση των τμημάτων του αντικειμένου ροής στα μέσα αποθήκευσης. Όλες αυτές οι δομές και η μετακίνησή τους από την πρωτεύουσα στην δευτερεύουσα μνήμη διαχειρίζονται μέσω των μεθόδων της κλάσης StreamTable (add_stream, delete_stream, find_stream, store/restore). Αντίστοιχες λειτουργίες σε υψηλότερο επίπεδο διαθέτει ο διαχειριστής αποθήκευσης. Οι μέθοδοι αυτοί αποτελούν την υλοποίηση της λειτουργικότητας όσον αφορά την αποθήκευση που παρέχεται από την υπηρεσία ροής. Έτσι επιτρέπεται η αποθήκευση, η εύρεση, η διαγραφή, η αντιγραφή και η μετακίνηση δεδομένων ροής. Για την επίτευξη των παραπάνω λειτουργιών το σύστημα αποθήκευσης συνεργάζεται με τον διαχειριστή συσκευών δίσκου, ενώ ειδικά για την λειτουργία της αποθήκευσης συμβουλευεται τον διαχειριστή τοποθέτησης δεδομένων (Data Placement Manager) για τον τρόπο αποθήκευσης των αντικειμένων ροής.

4.2.6 Διαδικασία αποθήκευσης των αντικειμένων ροής

Η υπηρεσία διαχείρισης δεδομένων ροής ακολουθεί την τεχνική διάσπασης των αντικειμένων σε τμήματα δεδομένων όπως αναφέρθηκε σε προηγούμενη ενότητα. Με τον τρόπο αυτό η αποθήκευση μπορεί να λάβει χώρα είτε σε συνεχόμενο χώρο είτε διασπαρμένα μέσα στον δίσκο, είτε ακόμα διασκορπισμένα σε πολλαπλούς διαφορετικούς δίσκους ακολουθώντας τεχνικές striping. Το σύστημα αποθήκευσης αναπτύχθηκε με στόχο να προσφέρει όλη την λειτουργικότητα που είναι απαραίτητη, ώστε να υποστηριχθούν όλες οι γνωστές τεχνικές τοποθέτησης δεδομένων χωρίς να υπάρχουν περιορισμοί.

Ο πίνακας πρόσβασης ενός αντικειμένου, που αναφέρθηκε στην προηγούμενη ενότητα, αποτελεί έναν πίνακα από εγγραφές που αναπαριστούν την τοποθέτηση σε φυσικό επίπεδο κάθε τμήματος του αντικειμένου. Η εγγραφή συνεπώς έχει ως παραμέτρους το αναγνωριστικό του δίσκου, τον αριθμό του αρχικού τομέα και το μέγεθος του τμήματος. Οι τρεις αυτές τιμές είναι αρκετές ώστε να βρεθεί και να ανακτηθεί ένα τμήμα δεδομένων από το υποσύστημα των μέσων αποθήκευσης. Με τον τρόπο αυτό μπορεί να υποστηριχθεί σε χαμηλότερο επίπεδο οποιαδήποτε τεχνική τοποθέτησης. Τα τμήματα των αντικειμένων μπορεί να είναι μεταβλητού μεγέθους, ένα γεγονός που βρίσκει εφαρμογή στην αποθήκευση αντικειμένων μεταβλητού ρυθμού (VBR). Στην περίπτωση που ένα αντικείμενο ροής είναι πολύ μεγάλο είναι φυσικό να δημιουργείται ένας σχετικά μεγάλος πίνακας πρόσβασης. Η πιθανότητα αυτή σε συνδυασμό με το γεγονός ότι η υπηρεσία ροής έχει σαν στόχο την εξυπηρέτηση μεγάλου αριθμού πελατών, που σημαίνει ότι αρκετοί πίνακες πρόσβασης πρέπει να βρίσκονται στην κύρια μνήμη σε όλη την διάρκεια της εξυπηρέτησης των αιτήσεων, εμφανίζεται το πρόβλημα έλλειψης αρκετού μεγέθους διαθέσιμης κύριας μνήμης.



Σχήμα 28: Η διαδικασία αποθήκευσης ενός αντικειμένου συνεχόμενης ροής

Το σύστημα αποθήκευσης επιτρέπει, ως λύση στο προηγούμενο πρόβλημα, την χρήση τμημάτων του πίνακα πρόσβασης στις περιπτώσεις εκείνες που είναι υπερβολικά μεγάλος. Έτσι κατά την διάρκεια της αναπαραγωγής ανακτάται το πρώτο τμήμα του πίνακα πρόσβασης, που αντιστοιχεί σε ένα μεγάλο μέρος του αποθηκευμένου αντικειμένου. Όταν η εξυπηρέτηση προχωρήσει αρκετά ώστε ανακτηθούν όλα σχεδόν τα τμήματα αντικειμένου που περιγράφονται από το πρώτο τμήμα του πίνακα πρόσβασης, δρομολογείται η ανάκτηση του δεύτερου τμήματος για να συνεχιστεί η ίδια διαδικασία.

Στο Σχήμα 28 περιγράφεται γραφικά η διαδικασία που ακολουθείται για την αποθήκευση ενός αντικειμένου ροής. Ένας πελάτης που βρίσκεται τοπικά ή απομακρυσμένα στέλνει μια αίτηση αποθήκευσης αντικειμένου στην υπηρεσία ροής. Η τελευταία δέχεται την αίτηση, την αποκωδικοποιεί και στην συνέχεια δημιουργεί ένα καινούργιο νήμα που θα αναλάβει την αποθήκευση, επιστρέφοντας στον πελάτη πληροφορία για τον τρόπο μεταφοράς των δεδομένων του αντικειμένου στο νέο νήμα. Το νέο νήμα που δημιουργήθηκε αρχίζει να λαμβάνει τα δεδομένα που πρέπει να αποθηκευτούν και χρησιμοποιεί λειτουργικότητα του συστήματος αποθήκευσης. Όπως προκύπτει από το αντίστοιχο σχήμα το σύστημα αποθήκευσης συμβουλευτεί τον διαχειριστή τοποθέτησης δεδομένων για τον τρόπο αποθήκευσης. Στην παρούσα φάση του συστήματος ο διαχειριστής τοποθέτησης λαμβάνει απλές αποφάσεις, ύστερα από συνεργασία με τον διαχειριστή συσκευών δίσκου και την επεξεργασία της πληροφορίας που δέχεται από το σύστημα αποθήκευσης. Η πληροφορία αυτή περιλαμβάνει το φάκελο στον οποίο ανήκει το αντικείμενο προς αποθήκευση, τον ρυθμό κατανάλωσης δεδομένων του αντικειμένου ροής και τον κύκλο εξυπηρέτησης. Με βάση το όνομα του φακέλου ο διαχειριστής τοποθέτησης βρίσκει τον μέγιστο αριθμό δίσκων στους οποίους μπορεί να διασπαστεί η αποθήκευση και τον διαθέσιμο χώρο σε αυτούς. Έχοντας συγκεντρώσει όλη

την απαραίτητη πληροφορία ο διαχειριστής τοποθέτησης καταλήγει στην απόφαση ποια τεχνική striping θα ακολουθηθεί και τον τρόπο δημιουργίας των τμημάτων. Τις αποφάσεις αυτές τις επιστρέφει σαν αποτέλεσμα στο σύστημα αποθήκευσης, το οποίο μέσα στο περιβάλλον επεξεργασίας του νήματος, αναλαμβάνει την διάσπαση των εισερχόμενων δεδομένων σε τμήματα τα οποία αποταμιεύονται προσωρινά σε ειδικό χώρο στην κύρια μνήμη και μέσω του διαχειριστή συσκευών αποθηκεύει τα τμήματα αυτά με κατάλληλο τρόπο στους κατάλληλους δίσκους. Ταυτόχρονα με αυτήν την διαδικασία χτίζονται σταδιακά και οι απαραίτητες δομές για τα μετα-δεδομένα. Έτσι δημιουργείται ένας νέος περιγραφέας αντικειμένου, ο οποίος εκχωρεί μια νέα εγγραφή στον πίνακα των αντικειμένων ροής, ενώ με βάση την πληροφορία της φυσικής θέσης των δεδομένων που επιστρέφεται κατά την αποθήκευση των τμημάτων από τον διαχειριστή συσκευών δίσκων, δημιουργείται ο πίνακας πρόσβασης του αντικειμένου. Μόλις τελειώσει η αποθήκευση όλων των δεδομένων, ο πίνακας πρόσβασης έχει ολοκληρωθεί, οπότε με την σειρά του αποθηκεύεται με την ταυτόχρονη ενημέρωση της φυσικής θέσης του στον περιγραφέα του αντικειμένου και τερματίζεται το νήμα επεξεργασίας της αίτησης αποθήκευσης.

4.3 Διαχείριση προσωρινής μνήμης

Η διαχείριση προσωρινής μνήμης έχει μελετηθεί επαρκώς για τα παραδοσιακά συστήματα βάσεων δεδομένων [Eff84]. Ο αντικειμενικός στόχος των προτεινόμενων σχημάτων διαχείρισης της μνήμης είναι η προώθηση του διαμοιρασμού του διαθέσιμου χώρου και η ελάττωση των προσπελάσεων των δίσκων, ώστε να αυξηθεί η παραγωγικότητα του συστήματος. Τα σχήματα όμως αυτά προέκυψαν ύστερα από μελέτη και πειραματισμούς πάνω σε παραδοσιακά δεδομένα.

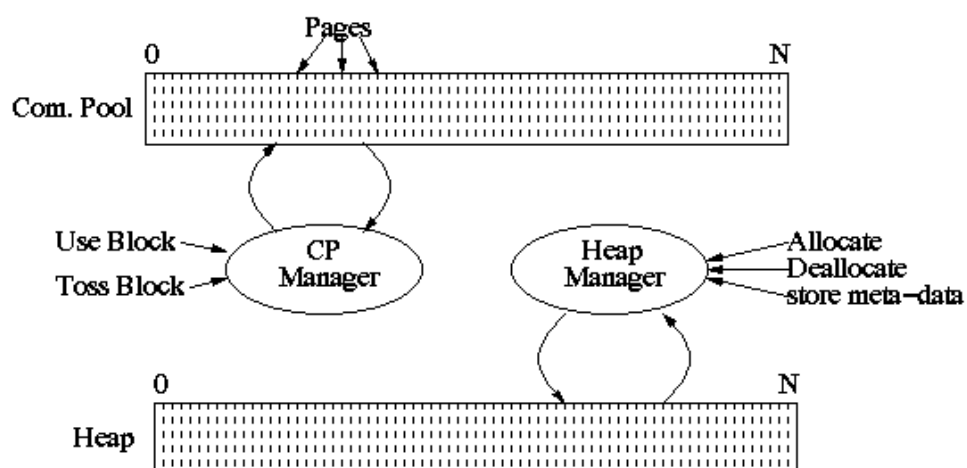
Οι πρώτες αρχές διαχείρισης προσωρινής μνήμης για δεδομένα πολυμέσων εμφανίζονται στα [And92][Chris91]. Η πιο διαδεδομένη πολιτική διαχείρισης μνήμης για πολυμέσα σε εξυπηρετητές αποθήκευσης είναι αυτή που ακολουθεί το σχήμα του διπλού αποταμιευτή. Για την εξυπηρέτηση μιας αίτησης ανάκτησης δεδομένων δεσμεύεται ένας χώρος στην προσωρινή μνήμη ο οποίος χωρίζεται στα δύο. Αρχικά το πρώτο τμήμα χρησιμοποιείται για την ανάκτηση δεδομένων από τους δίσκους. Μόλις το τμήμα αυτό γεμίσει με δεδομένα η διαδικασία της ανάκτησης συνεχίζει χρησιμοποιώντας το δεύτερο τμήμα, ενώ ταυτόχρονα ξεκινάει η διαδικασία αποστολής δεδομένων στον πελάτη από το πρώτο τμήμα. Το προηγούμενο σχήμα εργασίας με την συνεχή εναλλαγή χρήσης των τμημάτων λαμβάνει χώρα σε ολόκληρη την διάρκεια εξυπηρέτησης της αίτησης.

Η πολιτική χρήσης-απόρριψης (use & toss) UAT προτείνεται στο [Chris91] για την διαχείριση της προσωρινής μνήμης κατά την εξυπηρέτηση δεδομένων πολυμέσων και αποτελεί μια καλύτερη προσέγγιση. Σύμφωνα με την πολιτική αυτή, μόλις ένα τμήμα δεδομένων χρησιμοποιηθεί από τον αποταμιευτή, στην συνέχεια απορρίπτεται ώστε η ίδια θέση να αξιοποιηθεί για την αποταμίευση άλλων δεδομένων. Η UAT πολιτική λαμβάνει υπόψη την συνεχή φύση των αντικειμένων ροής που αναδεικνύει την μικρή πιθανότητα της προσπέλασης δεδομένων ενός αντικειμένου που έχουν ήδη ανακτηθεί και αποσταλεί στον

πελάτη. Έτσι με την πολιτική αυτή γίνεται μια προσπάθεια για την ελαχιστοποίηση του χώρου προσωρινής μνήμης που είναι απαραίτητος για την εξυπηρέτηση πολλών αιτήσεων ανάκτησης ανεξάρτητων αντικειμένων συνεχόμενης ροής. Αρκετές άλλες πολιτικές διαχείρισης μνήμης για πολυμέσα υπάρχουν στην βιβλιογραφία οι οποίες ακολουθούν την αρχή του διαμοιρασμού της μνήμης.

Οι μηχανισμοί προσωρινής αποταμίευσης είναι απαραίτητοι στην λειτουργία της υπηρεσίας διαχείρισης δεδομένων ροής για τις ανάγκες ανάκτησης και αποστολής δεδομένων αλλά και για την υποστήριξη των διεργασιών που λαμβάνουν χώρα εσωτερικά στο σύστημα. Η τελευταία περίπτωση αναφέρεται στην ανάγκη διατήρησης στην κύρια μνήμη των κατάλληλων δομών, όπως είναι για παράδειγμα οι περιγραφείς πρόσβασης του αντικειμένου, που υποδεικνύουν την φυσική θέση αποθήκευσης των τμημάτων δεδομένων και είναι απαραίτητη η γρήγορη προσπέλασή τους σε όλη την διάρκεια της ανάκτησης. Έτσι στο σύστημα έχουν αναπτυχθεί δύο διαχειριστές προσωρινής μνήμης που οδηγούν την πρόσβαση σε δύο διαφορετικούς αποταμιευτές.

Ο πρώτος είναι ο αποταμιευτής επικοινωνίας (communication pool) και χρησιμοποιείται για την υποστήριξη της ανάκτησης και αποστολής δεδομένων. Ο δεύτερος είναι ο εσωτερικός αποταμιευτής (Heap) ο οποίος αποβλέπει στην αποδοτική υποστήριξη των λειτουργιών του συστήματος. Και οι δυο αποταμιευτές διαχειρίζονται με βάση σελίδες μικρού μεγέθους. Όπως είναι φυσικό οι δύο αποταμιευτές ακολουθούν διαφορετικές πολιτικές, εφόσον η φύση των δεδομένων που αποταμιεύονται σε κάθε περίπτωση είναι διαφορετική. Στο Σχήμα 29 εμφανίζονται ο αποταμιευτής επικοινωνίας και ο εσωτερικός αποταμιευτής με τους αντίστοιχους διαχειριστές τους. Η πολιτική χρήσης-απόρριψης εφαρμόζεται στον διαχειριστή του αποταμιευτή επικοινωνίας όπως προκύπτει από το σχήμα.



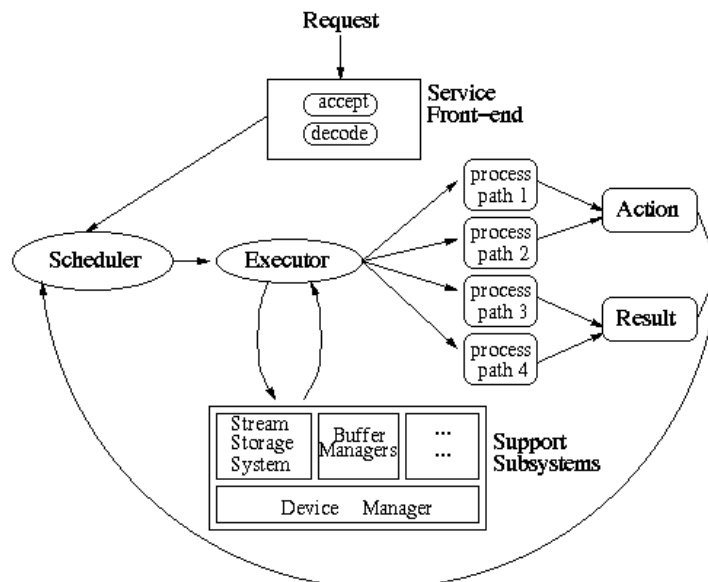
Σχήμα 29: Οι διαχειριστές των προσωρινών αποταμιευτών
Στο σχήμα εμφανίζονται οι δυο αποταμιευτές με τους αντίστοιχους διαχειριστές που χρησιμοποιούνται από την υπηρεσία ροής. Οι αποταμιευτές οργανώνονται σε σελίδες και η πρόσβαση τους ελέγχεται από τις εφαρμοζόμενες πολιτικές των διαχειριστών.

Ο εσωτερικός αποταμιευτής έχει γενικό χαρακτήρα. Όπως θα δούμε στο κεφάλαιο 6 καλύπτει όλες τις ανάγκες προσωρινής μνήμης που προκύπτουν ακολουθώντας το σχήμα

εκχώρησης-απελευθέρωσης (allocate-deallocate), παρακάμπτοντας τις συνηθισμένες υπηρεσίες διαχείρισης μνήμης του λειτουργικού. Στον ίδιο χώρο του εσωτερικού αποταμιευτή αποθηκεύονται και δομές μετα-δεδομένων (όπως πίνακες δεικτοδότησης και περιγραφείς), οι οποίες διαχειρίζονται ακολουθώντας την πολιτική της πιο πρόσφατης αναφοράς (LRU).

4.4 Διαχείριση αιτήσεων

Η υπηρεσία διαχείρισης δεδομένων ροής αποτελεί στην ουσία έναν ανεξάρτητο εξυπηρετητή αντικειμένων συνεχόμενης ροής. Αντικειμενικός στόχος της υπηρεσίας είναι η ταυτόχρονη εξυπηρέτηση μεγάλου αριθμού πελατών που ζητούν την αναπαραγωγή αντικειμένων πολυμέσων. Έτσι στα πλαίσια του σχήματος πελάτη-εξυπηρετητή, οι χρήστες στέλνουν μηνύματα αιτήσεων στον εξυπηρετητή. Οι αιτήσεις αυτές περιέχουν πληροφορία σχετικά με το αντικείμενο που επιθυμούν να προσπελάσουν και το είδος ενέργειας (play, pause, jump ...) που πρέπει να εκτελεστεί.



Σχήμα 30: Το μοντέλο διαχείρισης αιτήσεων

Ο εκτελεστής είναι ένα βασικό αντικείμενο του συστήματος και υλοποιεί όλα τα δυνατά μονοπάτια επεξεργασίας των αιτήσεων που υποστηρίζει η υπηρεσία ροής. Η επεξεργασία λαμβάνει χώρα αξιοποιώντας την λειτουργικότητα των υποσυστημάτων υποστήριξης που έχουν αναπτυχθεί.

Ένας ολόκληρος μηχανισμός αντιμετώπισης και επεξεργασίας των αιτήσεων έχει αναπτυχθεί στην υπηρεσία ροής και περιγράφεται γραφικά στο Σχήμα 30. Το υποσύστημα εισόδου της υπηρεσίας αναλαμβάνει να δεχτεί την αίτηση από κάποιον πελάτη και να την αποκωδικοποιήσει. Στην συνέχεια ο κεντρικός δρομολογητής του συστήματος οδηγεί την εκτέλεση της αίτησης που βρίσκεται σε αναμονή μέσω του εκτελεστή (executor). Ο εκτελεστής αποτελεί ένα βασικό αντικείμενο του συστήματος, οι μέθοδοι του οποίου καθορίζουν την επεξεργασία των εισερχόμενων αιτήσεων. Με δεδομένο την πληροφορία

που φέρει η αίτηση, ο εκτελεστής επιβάλλει και το κατάλληλο μονοπάτι εκτέλεσης μέσω της αντίστοιχης μεθόδου.

Όπως προκύπτει από το σχήμα, κατά την διάρκεια της επεξεργασίας των αιτήσεων ο εκτελεστής συνεργάζεται με τα υποσυστήματα υποστήριξης της υπηρεσίας ροής, με στόχο την συγκέντρωση πληροφορίας αλλά και εκτέλεση διάφορων εργασιών. Το τέλος της επεξεργασίας μιας αίτησης εμφανίζει ένα αποτέλεσμα ή μια ενέργεια που πρέπει να λάβει χώρα στο εσωτερικό της υπηρεσίας. Αυτό το αποτέλεσμα ή ενέργεια επιστρέφεται στον κεντρικό δρομολογητή για την συνέχιση της εξυπηρέτησης.

Για έναν εξυπηρετητή πολυμέσων μεγάλης κλίμακας είναι πολύ σημαντικό να παρέχει υπηρεσίες χωρίς διακυμάνσεις στην ποιότητά τους. Αυτός είναι και ο αντικειμενικός στόχος του μηχανισμού δρομολόγησης, ο οποίος αναπτύχθηκε με γνώμονα την εξυπηρέτηση πολλών ανεξάρτητων πελατών, χωρίς η επεξεργασία ενός πελάτη να έχει επιδράσεις σε κάποιον άλλο. Αναπόφευκτα όμως υπάρχουν περιπτώσεις αύξησης του φόρτου εργασίας σε επίπεδα που υπερκαλύπτουν τα φυσικά όρια του υπολογιστικού συστήματος. Είναι επομένως ευνόητη η ανάγκη ύπαρξης ενός τμήματος λογισμικού, το οποίο θα ελέγχει την υποβολή νέων αιτήσεων λαμβάνοντας υπόψη τις δυνατότητες των υποσυστημάτων δίσκων, προσωρινής μνήμης, δικτύου και τέλος του επεξεργαστή.

Η υπηρεσία ροής όπως θα αναφερθεί στο επόμενο κεφάλαιο εξυπηρετεί πολλούς πελάτες ταυτόχρονα εφαρμόζοντας κύκλους επεξεργασίας, ένα γεγονός που συμβαδίζει με την περιοδική φύση των αντικειμένων ροής. Σε κάθε κύκλο ανακτώνται από τα μέσα αποθήκευσης τμήματα δεδομένων για κάθε πελάτη. Στόχος του ελεγκτή υποβολής είναι να εγγυηθεί ότι απαιτήσεις ποιότητας των ήδη εξυπηρετούμενων πελατών δεν θα παραβιαστούν εξαιτίας της εισόδου ενός νέου πελάτη. Οι αλγόριθμοι ελέγχου υποβολής αιτήσεων που υπάρχουν στην βιβλιογραφία κατατάσσονται σε δύο βασικές κατηγορίες: τους αιτιοκρατικούς (deterministic) και τους στατιστικούς. Η βαθύτερη μελέτη αλγορίθμων ελέγχου υποβολής αιτήσεων δεν αποτέλεσε στόχο της παρούσας διατριβής. Η τρέχουσα υλοποίηση εφαρμόζει ένα απλό αλγόριθμο, ο οποίος λαμβάνει υπόψη τον φορτίο των δίσκων του συστήματος, τον διαθέσιμο χώρο στους προσωρινούς αποταμιευτές και τον φόρτο του επεξεργαστή. Έτσι με βάση το αποτέλεσμα κάθε νέας αίτησης που επεξεργάστηκε από τον εκτελεστή, ο δρομολογητής αποφασίζει αν η αίτηση πρέπει να γίνει δεκτή ή να απορριφθεί, σε συνδυασμό με την πληροφορία της τρέχουσας κατάστασης του συστήματος που συγκεντρώνει.

Η εφαρμογή και η υλοποίηση του μοντέλου διαχείρισης των αιτήσεων του σχήματος, περιγράφονται στο κεφάλαιο 6 με αρκετή λεπτομέρεια, καθώς συνδέεται άμεσα με τον εφαρμοζόμενο μηχανισμό δρομολόγησης.

4.5 Διαχείριση μετάδοσης

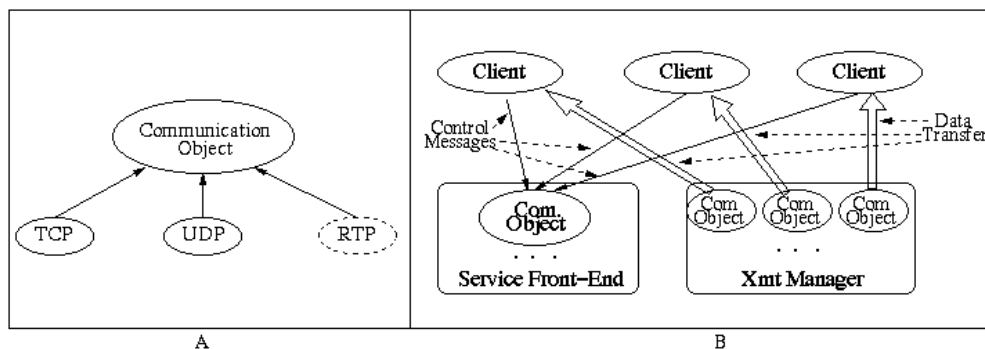
Ο σχεδιασμός προβλέπει ότι η υπηρεσία διαχείρισης δεδομένων ροής δεν είναι μόνο υπεύθυνη για την αποθήκευση και ανάκτηση των αντικειμένων. Αντίθετα αποτελεί μια

ολοκληρωμένη αντιμετώπιση και περιλαμβάνει και τμήματα λογισμικού που αναλαμβάνουν την μετάδοση των δεδομένων στους πελάτες.

Τα κύρια ζητήματα τα οποία προκύπτουν για την μετάδοση δεδομένων πολυμέσων πάνω από τα συνηθισμένα δίκτυα πακέτων είναι το εύρος μετάδοσης (bandwidth), οι καθυστερήσεις (delays) και φαινόμενα εκρηκτικής μετάδοσης (burstiness). Αντικείμενα ήχου και video αποτελούνται από δεδομένα μεγάλου μεγέθους τα οποία πρέπει να μεταδοθούν μέσω δικτύου. Το μεγάλο μέγεθος αλλά και οι απαιτήσεις αναπαραγωγής των πολυμέσων επιβάλλουν η μετάδοση να έχει χαρακτήρα συνεχόμενης ροής και όχι διακριτής εκρηκτικής μετάδοσης τμημάτων δεδομένων. Το γεγονός αυτό έρχεται σε αντίθεση με την μετάδοση των παραδοσιακών δεδομένων, όπως κείμενο, αρχεία και γραφικά, τα οποία μπορούν να αντιμετωπίσουν καθυστερήσεις ανάμεσα στα πακέτα μετάδοσης.

Στην προσπάθεια επίτευξης μετάδοσης συνεχόμενης ροής τον σημαντικότερο ρόλο έρχεται να αναλάβει ο εξυπηρετητής αποθήκευσης πολυμέσων. Κι αυτό γιατί πρέπει να εφαρμόσει την κατάλληλη πολιτική δρομολόγησης και τα κατάλληλα πρωτόκολλα μετάδοσης ώστε να ικανοποιηθούν οι απαιτήσεις αναπαραγωγής των αντικειμένων ροής στην πλευρά του πελάτη.

Το δημοφιλές πρωτόκολλο TCP/IP αν και στηρίζει την συντριπτική πλειοψηφία των εφαρμογών του διαδικτύου, εντούτοις δεν είναι κατάλληλο για την υποστήριξη της μετάδοσης δεδομένων ευαίσθητων στις χρονικές καθυστερήσεις.



Σχήμα 31: Επικοινωνία και μετάδοση δεδομένων

A: Η κλάση που αναπαριστά το αντικείμενο επικοινωνίας παρέχει ένα επίπεδο αφαίρεσης ως προς το υλοποιημένο πρωτόκολλο που χρησιμοποιείται. B: Για κάθε ανάγκη λήψης/μετάδοσης δεδομένων η υπηρεσία ροής δημιουργεί ένα αντικείμενο επικοινωνίας. Το υποσύστημα εισόδου χρησιμοποιεί ένα αντικείμενο επικοινωνίας, για την λήψη των αιτήσεων και την αποστολή των απαντήσεων, το οποίο υλοποιεί το πρωτόκολλο TCP/IP. Αντίθετα ο διαχειριστής μετάδοσης δρομολογεί κατάλληλα πολλαπλά αντικείμενα επικοινωνίας για την αποστολή των ανακτημένων δεδομένων στους πελάτες, ακολουθώντας το UDP/IP πρωτόκολλο.

Μια σαφώς καταλληλότερη λύση αποτελεί το πρωτόκολλο UDP/IP, καθώς μπορεί να χρησιμοποιηθεί σε έναν μηχανισμό μετάδοσης ροής πακέτων χωρίς καθυστερήσεις ανάδρασης. Βέβαια ένα από τα προβλήματα αυτού του πρωτοκόλλου είναι το ζήτημα της αξιόπιστης μετάδοσης δεδομένων. Με την κατάλληλη δρομολόγηση όμως της μετάδοσης των πακέτων και σε συνδυασμό με το γεγονός ότι η αναπαραγωγή πολυμέσων δεν επηρεάζεται σε μεγάλο βαθμό από τις απώλειες μικρών τμημάτων δεδομένων, περιορίζονται οι αρνητικές επιδράσεις από την εφαρμογή του UDP πρωτοκόλλου.

Λαμβάνοντας υπόψη τα παραπάνω η υπηρεσία ροής ενσωματώνει και τα δυο πρωτόκολλα TCP και UDP. Το πρώτο κυρίως εφαρμόζεται για την επικοινωνία ελέγχου μεταξύ των πελατών και του εξυπηρετητή λόγω της αξιοπιστίας που προσφέρει, ενώ το δεύτερο για την αποστολή των δεδομένων των αντικειμένων ροής (χωρίς βέβαια να αποκλείεται η χρήση του TCP πρωτοκόλλου για τον σκοπό αυτό, όπου οι συνθήκες το επιβάλλουν).

Στο Σχήμα 31 διακρίνεται η χρήση του αντικειμένου επικοινωνίας το οποίο έχει δημιουργηθεί και παρέχει ένα γενικό σύνολο λειτουργιών αδιαφάνες ως προς το πρωτόκολλο που χρησιμοποιείται. Έτσι σύμφωνα με αυτό το σχήμα είναι εύκολη η προσθήκη νέων πρωτοκόλλων μετάδοσης, όπως RTP (Real Time Protocol), τα οποία μπορούν να υλοποιηθούν κάτω από την αφηρημένη κλάση που αναπαριστά το αντικείμενο επικοινωνίας χωρίς άλλες αλλαγές στο υπόλοιπο σύστημα.

Το σημαντικότερο όμως ζήτημα που προκύπτει από την διαχείριση της μετάδοσης είναι η σωστή δρομολόγηση των πακέτων που αποστέλλονται σε πραγματικό χρόνο και εξετάζεται διεξοδικά στο επόμενο κεφάλαιο.

4.6 Δρομολόγηση εργασιών

Στις προηγούμενες ενότητες περιγράφηκαν τα τμήματα λογισμικού που αναπτύχθηκαν για την υποστήριξη της υπηρεσίας διαχείρισης δεδομένων ροής. Έγινε αναφορά στα υποσυστήματα διαχείρισης των δίσκων, της αποθήκευσης αντικειμένων, της προσωρινής μνήμης, των αιτήσεων και της μετάδοσης. Όπως προκύπτει από το Σχήμα 24, όλες αυτές οι εργασίες διαχείρισης διαφορετικών πόρων χρειάζονται έναν ολοκληρωμένο μηχανισμό δρομολόγησης που θα εγγυάται την βέλτιστη χρήση όλων των πόρων του συστήματος, εξασφαλίζοντας την ομαλή ροή των δεδομένων δια μέσου όλων των επιπέδων εργασιών από την ανάκτηση μέχρι την μετάδοση των δεδομένων στο δίκτυο.

Ο προτεινόμενος μηχανισμός δρομολόγησης της διατριβής αξιοποιεί την πολυνηματική τεχνολογία και εφαρμόζει τις αρχές προγραμματισμού πραγματικού χρόνου. Η περιγραφή και η λεπτομερής ανάλυση, αυτού του μηχανισμού δρομολόγησης, ακολουθεί στο 7ο κεφάλαιο.

4.7 Περίληψη

Σε αυτό το κεφάλαιο περιγράφηκαν τα υποσυστήματα λογισμικού που αναπτύχθηκαν για την υποστήριξη της υπηρεσίας διαχείρισης δεδομένων ροής. Ένα σύστημα διαχείρισης πολυμέσων χρειάζεται να έχει τον έλεγχο για ένα αριθμό από διαφορετικές εργασίες. Οι εργασίες αυτές αναφέρονται: στην διαχείριση των φυσικών μέσων αποθήκευσης, στην διαχείριση περιεχομένου (δηλαδή στην οργάνωση της αποθήκευσης και τοποθέτησης των αντικειμένων ροής), στην διαχείριση της προσωρινής αποταμίευσης δεδομένων, στην επεξεργασία των αιτήσεων των πελατών και στην μετάδοση των δεδομένων.

Το φυσικό επίπεδο αποθήκευσης της υπηρεσίας ροής αποτελείται από μαγνητικούς δίσκους τεχνολογίας SCSI. Η τεχνολογία αυτή μας επιτρέπει να χρησιμοποιούμε πολλούς δίσκους πάνω στον ίδιο δίαυλο, η επεκτασιμότητα όμως αυτού του σχήματος είναι περιορισμένη.

Ο σχεδιασμός της υπηρεσίας ροής προϋποθέτει τον πλήρη έλεγχο της πρόσβασης σε κάθε επίπεδο μνήμης, ώστε να είναι σε θέση να δρομολογήσει σωστά και σε πραγματικό χρόνο την εξυπηρέτηση των αιτήσεων ανάκτησης πολυμέσων. Έτσι για την ανάπτυξη του υποσυστήματος αποθήκευσης επιλέχθηκε η χρήση του raw interface που προσφέρει το λειτουργικό σύστημα Solaris, με το οποίο παρακάμπτουμε το σύστημα αρχείων και τους μηχανισμούς αποταμίευσης του λειτουργικού, με στόχο την ταχύτερη λειτουργία των δίσκων και την αποφυγή των περιορισμών του παραδοσιακού συστήματος αρχείων.

Στο κεφάλαιο περιγράφηκαν επίσης με αρκετή λεπτομέρεια η αρχιτεκτονική και οι δομές υποστήριξης του διαχειριστή των μέσων αποθήκευσης και του συστήματος αποθήκευσης των αντικειμένων ροής, δύο από τα βασικότερα τμήματα λογισμικού της υπηρεσίας ροής. Η ανάπτυξη του συστήματος αποθήκευσης έγινε με στόχο να είναι δυνατή η εφαρμογή οποιουδήποτε σχήματος τοποθέτησης δεδομένων πάνω σε πολλαπλούς δίσκους.

Δuo διαφορετικοί προσωρινοί αποταμιευτές χρησιμοποιούνται στην υπηρεσία ροής . Ο αποταμιευτής επικοινωνίας (CP) αποβλέπει στην υποστήριξη των διεργασιών ανάκτησης και μετάδοσης των αντικειμένων ροής, ενώ ο εσωτερικός αποταμιευτής (Heap) είναι απαραίτητος για την λειτουργία του συστήματος και την επεξεργασία των αιτήσεων (αναφέρεται στην ανάγκη διατήρησης στην κύρια μνήμη των κατάλληλων δομών).

Ο εκτελεστής (Executor) αποτελεί ένα βασικό αντικείμενο το συστήματος, οι μέθοδοι του οποίου καθορίζουν την επεξεργασία των εισερχόμενων αιτήσεων. Με βάση την πληροφορία που φέρει η κάθε αίτηση, ο εκτελεστής επιβάλλει και το κατάλληλο μονοπάτι εκτέλεσης μέσω της αντίστοιχης μεθόδου. Όπως περιγράφηκε ο εκτελεστής, κατά την διάρκεια της επεξεργασίας των αιτήσεων, συνεργάζεται με τα υποσυστήματα αποθήκευσης και αποταμίευσης.

Για την υποστήριξη των αναγκών επικοινωνίας και αποστολής δεδομένων έχει αναπτυχθεί το αντικείμενο επικοινωνίας, το οποίο παρέχει ένα γενικό σύνολο λειτουργιών αδιαφάνες ως προς το πρωτόκολλο που χρησιμοποιείται. Για την επίτευξη της επικοινωνίας (λήψη μηνυμάτων αιτήσεων πελατών) γίνεται εφαρμογή του πρωτοκόλλου TCP λόγω της αξιοπιστίας που προσφέρει, ενώ για την αποστολή των δεδομένων πολυμέσων χρησιμοποιείται το πρωτόκολλο UDP λόγω των πλεονεκτημάτων που εμφανίζει για ροές δεδομένων πραγματικού χρόνου.

Η ταυτόχρονη λειτουργία όλων των υποσυστημάτων υποστήριξης της υπηρεσίας ροής είναι ανάγκη να οδηγείται από τον κατάλληλο μηχανισμό δρομολόγησης, ο οποίος θα περιγραφεί στα επόμενα κεφάλαια.

➤ Υποσυστήματα Ανάκτησης και Μετάδοσης Δεδομένων



Σε αυτό το κεφάλαιο περιγράφονται τα υποσυστήματα υποστήριξης των ροών δεδομένων που καλείται να διαχειριστεί ο εξυπηρετητής. Ιδιαίτερη σημασία έχουν τα αποτελέσματα των πειραμάτων που έλαβαν χώρα. Τα πειράματα έχουν σαν στόχο την ανάδειξη των δυνατοτήτων, τόσο του εξυπηρετητή, όσο και των πελατών ανάλογα με τον εξοπλισμό που διαθέτουν. Εξετάζεται η απόδοση μετάδοσης δεδομένων του εξυπηρετητή ροής και οι απαιτήσεις που επιβάλλονται στο σύστημα από διαφορετικούς πελάτες, ενώ ιδιαίτερη σημασία έχει η συμπεριφορά του υποσυστήματος ανάκτησης δεδομένων σε συνθήκες υψηλού φόρτου εργασίας, η οποία καταγράφεται πειραματικά σε αυτό το κεφάλαιο.

5.1 Παράμετροι σχεδιασμού της υπηρεσίας ροής

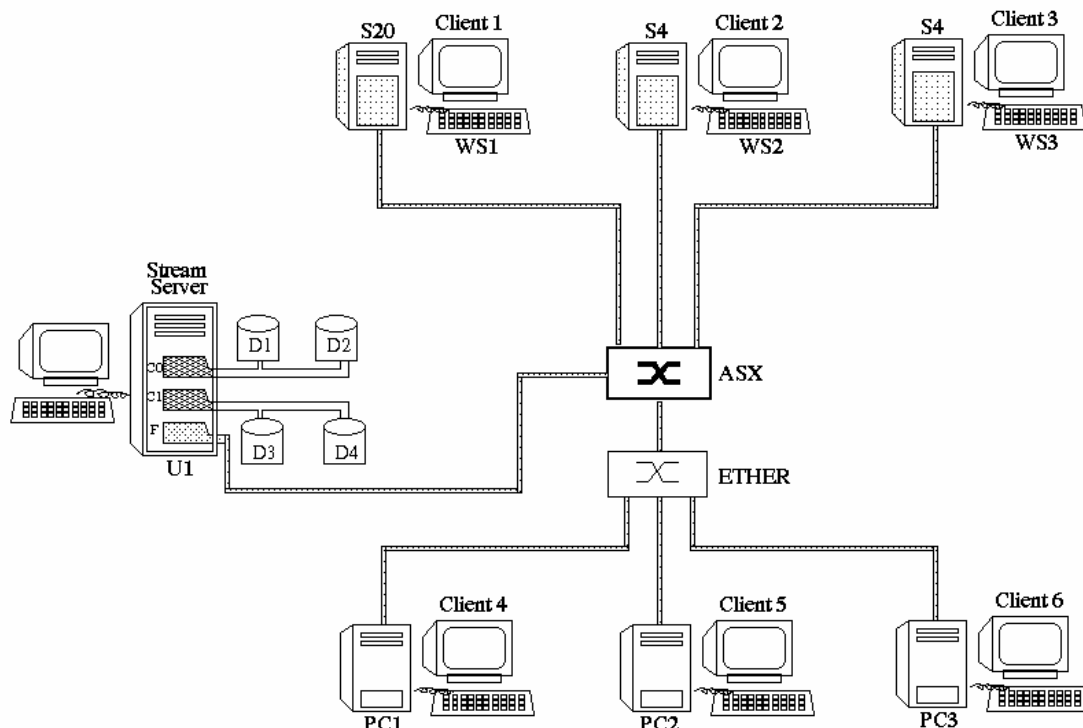
Πριν γίνουν τα πρώτα βήματα σχεδιασμού και ανάπτυξης της υπηρεσίας ροής είναι απαραίτητο να αναλυθούν οι απαιτήσεις που προκύπτουν κατά την λειτουργία ενός συστήματος εξυπηρέτησης δεδομένων συνεχόμενης ροής. Πρέπει να εντοπιστούν εκείνες οι παράμετροι που επηρεάζουν την απόδοση του συστήματος, άλλα και τυχόν προβλήματα που μπορούν να εμφανιστούν σε πραγματικές συνθήκες. Για όλα αυτά τα στοιχεία πρέπει να έχει γνώση ιδιαίτερα ο μηχανισμός δρομολόγησης, ώστε να είναι σε θέση να οδηγήσει αποτελεσματικά την λειτουργία του συστήματος, μέσα στα πλαίσια των φυσικών του δυνατοτήτων, αλλά και να ικανοποιεί τις διαφορετικές ανάγκες που πηγάζουν από την εξυπηρέτηση ετερογενών πελατών σε ανοιχτά περιβάλλοντα.

Στα πλαίσια αυτής της προσπάθειας δημιουργήθηκε ένα πειραματικό περιβάλλον το οποίο αποσκοπεί στην καταγραφή μετρήσεων και την ανάδειξη των παραγόντων που ο δρομολογητής πρέπει να λάβει υπόψη για την αποτελεσματικότερη λειτουργία του.

5.2 Η αρχιτεκτονική υλικού

Στο σχήμα Σχήμα 32 εμφανίζεται η αρχιτεκτονική υλικού που χρησιμοποιήθηκε για την διεξαγωγή όλων των πειραμάτων. Η υπηρεσία ροής συνήθως τρέχει σε μια Ultra-1 μηχανή

και έχει δύο προσαρμοστές SCSI-2, οι οποίοι ελέγχουν δύο συσκευές δίσκων ο καθένας (μέγιστης ονομαστικής απόδοσης 5Mbps). Η μετάδοση των δεδομένων ροής λαμβάνει χώρα μέσω μιας ATM κάρτας δικτύου μέγιστης ονομαστικής απόδοσης 155Mbps. Οι πελάτες αποτελούν άλλα έξι μηχανήματα (2 sparcstation4, 1 sparcstation20 και τρία PCs).



Σχήμα 32: Η αρχιτεκτονική υλικού

Το σχήμα υλικού που χρησιμοποιήθηκε για την διεξαγωγή των πειραμάτων. Η υπηρεσία αποθήκευσης δεδομένων ροής έχει εγκατασταθεί και τρέχει στο μηχανήμα U1 (Ultra 1), στο οποίο διακρίνονται οι προσαρμοστές SCSI (c0,c1), οι δίσκοι (D1-4), και η κάρτα δικτύου ATM(F). Σαν πελάτες χρησιμοποιούνται δύο sparc-4 (s4) και ένα sparc-20 (s20), τα οποία βρίσκονται συνδεδεμένα μέσω ενός ATM διακόπτη με τον εξυπηρετητή. Επίσης τρεις ακόμη προσωπικοί υπολογιστές (PC1-3) συμπεριλαμβάνονται στους πελάτες του διανεμημένου συστήματος, για τους οποίους επιτρέπεται η πρόσβαση στον εξυπηρετητή μέσω του διακόπτη Ethernet, που παρέχει σε κάθε υπολογιστή από ένα κανάλι μεταφοράς δεδομένων ονομαστικού εύρους 10Mbps

Οι προσωπικοί υπολογιστές φέρουν επεξεργαστές Pentium 166Mhz (PC1), Pentium Pro 200Mhz (PC2) και Pentium II 433Mhz (PC3) αντίστοιχα, ενώ διαθέτουν Ethernet κάρτες δικτύου ονομαστικής απόδοσης 10Mbps. Οι σταθμοί εργασίας Sparc είναι διασυνδεδεμένοι στο τοπικό ATM δίκτυο με την βοήθεια διακόπτη. Οι προσωπικοί υπολογιστές επίσης, όπως προκύπτει από το σχήμα, είναι διασυνδεδεμένοι μέσω ενός διακόπτη Ethernet, ο οποίος παρέχει από ένα κανάλι μεταφοράς δεδομένων εύρους 10Mbps σε κάθε υπολογιστή.

5.3 Πειράματα μετάδοσης δεδομένων

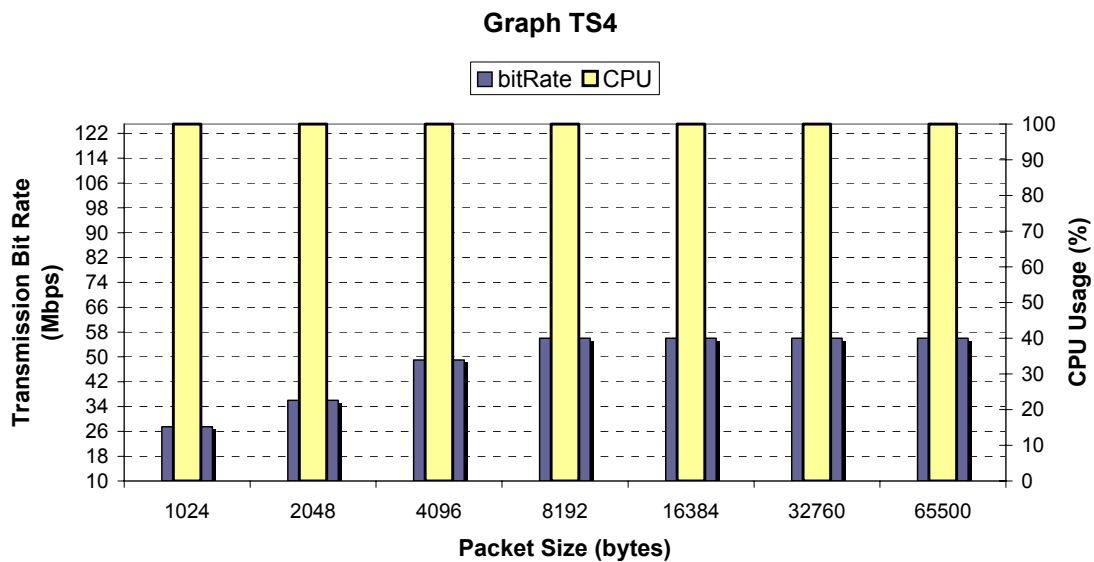
Σε αυτό το πείραμα εξετάζουμε την συμπεριφορά και τις δυνατότητες μετάδοσης δεδομένων, όταν ο εξυπηρετητής μετάδοσης τρέχει σε διαφορετικές υπολογιστικές μηχανές. Στόχος του πειράματος αποτελεί η ανάδειξη της μέγιστης απόδοσης μετάδοσης δεδομένων και οι απαιτήσεις επεξεργαστικής ισχύος που προκύπτουν. Στα πειράματα μετάδοσης

δεδομένων χρησιμοποιήθηκαν τρεις διαφορετικών δυνατοτήτων υπολογιστές στους οποίους έτρεχε αποκλειστικά ο εξυπηρετητής μετάδοσης, ενώ πρέπει να σημειωθεί ότι η αποστολή δεδομένων λαμβάνει χώρα με την χρήση της βασικής στοίβας πρωτοκόλλων UDP/IP. Ο εξυπηρετητής μετάδοσης θεωρεί ότι δεδομένα βρίσκονται συνεχώς διαθέσιμα στον αποταμιευτή επικοινωνίας, δημιουργεί UDP πακέτα τα οποία στέλνει συνεχώς σε αυθαίρετη διεύθυνση μέσω του τοπικού ATM δικτύου.

Σε όλους τους υπολογιστές του πειράματος χρησιμοποιείται η ίδια ATM κάρτα (Fore 200ASX) ονομαστικής ικανότητας μετάδοσης 155 Mbps.

Όλα τα διαγράμματα που ακολουθούν σε αυτή την ενότητα περιγράφουν τον μέγιστο ρυθμό μετάδοσης δεδομένων που ένα συγκεκριμένο υπολογιστικό σύστημα μπορεί να επιδώσει. Ο οριζόντιος άξονας φέρει τα μεγέθη των πακέτων που χρησιμοποιήθηκαν, ο πρωτεύων κάθετος άξονας αναπαριστά τον ρυθμό μετάδοσης δεδομένων και τέλος ο δευτερεύων κάθετος άξονας υποδεικνύει το ποσοστό χρήσης του επεξεργαστή που έλαβε χώρα κατά την διάρκεια του πειράματος μετάδοσης.

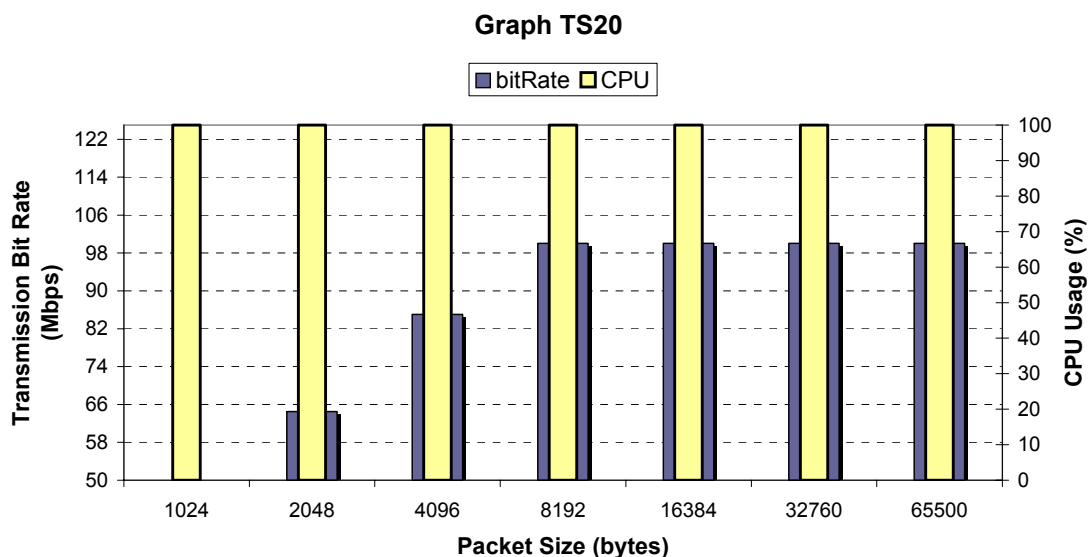
Στο Σχήμα 33 εμφανίζονται οι μετρήσεις που καταγράφηκαν για ένα υπολογιστή SparcStation 4. Παρατηρούμε ότι αυξάνοντας το μέγεθος του αρχικού πακέτου από 1024 σε 8192 επιτυγχάνουμε μία αύξηση στην μέγιστη απόδοση μετάδοσης από 27.5 σε 56 Mbps. Μεγαλύτερη αύξηση στο μέγεθος των πακέτων αποστολής δεν έχει καμία επίπτωση στην απόδοση. Επίσης πρέπει να σημειωθεί ότι σε όλες τις περιπτώσεις είχαμε πλήρη χρήση του επεξεργαστή (100%).



Σχήμα 33: Μέγιστος ρυθμός μετάδοσης δεδομένων. (Sparc4, Μετρήσεις Πειράματος)
Στο διάγραμμα αυτό εικονίζεται ο μέγιστος ρυθμός μετάδοσης δεδομένων που μπορεί να επιτευχθεί. Στο πείραμα αυτό χρησιμοποιήθηκε ένα SparcStation 4 με κάρτα δικτύου Fore 200ASX. Οι μετρήσεις δείχνουν μέγιστη ικανότητα μετάδοσης δεδομένων 56 Mbps με 100% χρήση του επεξεργαστή.

Ανάλογες μετρήσεις έγιναν σε ένα SparcStation 20 υπολογιστή και περιγράφονται στο ακόλουθο διάγραμμα (Σχήμα 34). Και σε αυτήν την περίπτωση όπως και στην προηγούμενη, η χρήση μικρού πακέτου δεδομένων μειώνει την μέγιστη απόδοση

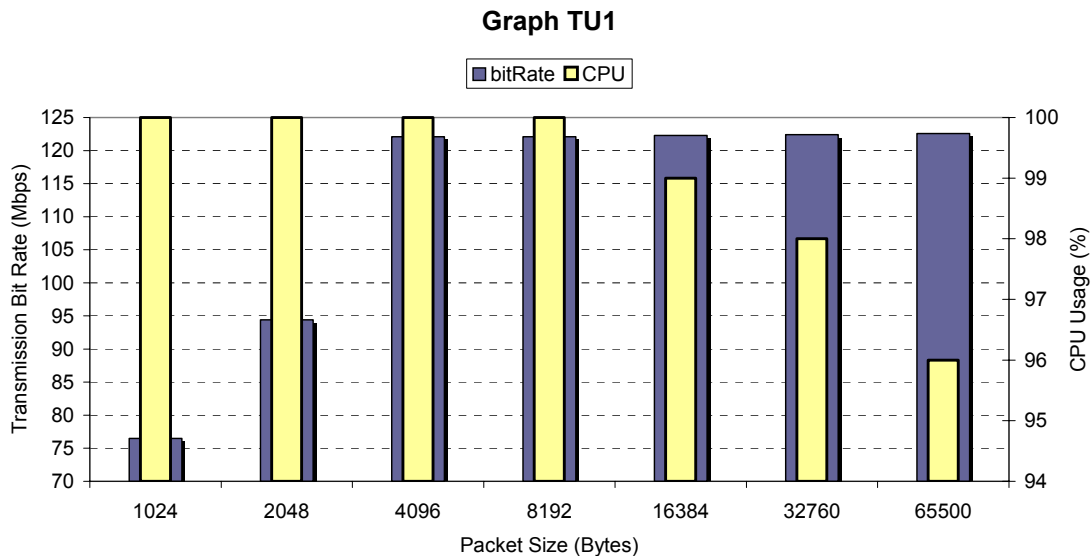
μετάδοσης, η οποία ανέρχεται στα 100 Mbps για πακέτα μεγαλύτερα από 8192 bytes. Μπορεί κανείς να παρατηρήσει ότι η μεγαλύτερη επεξεργαστική ικανότητα του Sparc20 υπολογιστή σε σύγκρισή με τον Sparc4, οδηγεί σε μεγαλύτερη απόδοση μετάδοσης, αλλά η απασχόληση του επεξεργαστή εξακολουθεί να παραμένει πλήρης (100%) για την εργασία αυτή.



Σχήμα 34: Μέγιστος ρυθμός μετάδοσης δεδομένων. (Sparc20- Μετρήσεις Πειράματος)
Στο διάγραμμα αυτό εικονίζεται ο μέγιστος ρυθμός μετάδοσης δεδομένων που μπορεί να επιτευχθεί. Στο πείραμα αυτό χρησιμοποιήθηκε ένα SparcStation 20 με κάρτα δικτύου Fore 200ASX. Οι μετρήσεις δείχνουν μέγιστη ικανότητα μετάδοσης δεδομένων 100 Mbps με 100% χρήσης του επεξεργαστή.

Στην προσπάθεια να αναδείξουμε τον ρόλο της επεξεργαστικής ισχύος στην μετάδοση δεδομένων, άλλα και της μέγιστης απόδοσης του υλικού δικτύου που χρησιμοποιούμε, τρέχουμε το ίδιο πείραμα σε ένα ακόμα ισχυρότερο επεξεργαστικά σύστημα.

Οι μετρήσεις που προέκυψαν από ένα Ultra One υπολογιστή καταγράφονται στο Σχήμα 35. Σε αντίθεση με τα προηγούμενα πειράματα σε αυτή την περίπτωση επιτυγχάνουμε μια καλή απόδοση 122 Mbps με σχετικά μικρό μέγεθος πακέτων (4096 bytes). Μεγαλύτερη αύξηση του μεγέθους των πακέτων αποστολής έχει πολύ μικρές θετικές επιπτώσεις στην συνολική απόδοση, φτάνοντας μέχρι 122,6 Mbps για πακέτα μεγέθους 65500 bytes. Αντίθετα αυτή η περαιτέρω αύξηση του μεγέθους των χρησιμοποιούμενων πακέτων επιφέρει μια μικρή, αλλά αξιοσημείωτη μείωση στο ποσοστό χρήσης του επεξεργαστή (μέχρι 96% για πακέτα μεγέθους 65500 bytes)



Σχήμα 35 Μέγιστος ρυθμός μετάδοσης δεδομένων. (Ultra One, Μετρήσεις Πειράματος)
 Στο διάγραμμα αυτό εικονίζεται ο μέγιστος ρυθμός μετάδοσης δεδομένων που μπορεί να επιτευχθεί. Στο πείραμα αυτό χρησιμοποιήθηκε ένας υπολογιστής Ultra One με κάρτα δικτύου Fore 200ASX. Οι μετρήσεις δείχνουν μέγιστη ικανότητα μετάδοσης δεδομένων 122.6 Mbps με 96% χρήσης του επεξεργαστή.

Το γεγονός ότι καταφέραμε να μετρήσουμε την μέγιστη απόδοση μετάδοσης δεδομένων χωρίς ο επεξεργαστής να είναι 100% απασχολημένος, πιθανότατα σημαίνει ότι πλησιάσαμε την μέγιστη απόδοση της κάρτας δικτύου. Η υπόθεση αυτή ενισχύεται περισσότερο, αν αναλογιστούμε ότι στην πραγματικότητα έχουμε ρυθμό μετάδοσης μεγαλύτερο του 122,6 Mbps, καθώς κατά την διαδικασία της μετάδοσης προσθέτονται στα ωφέλιμα δεδομένα και αρκετά ακόμα δεδομένα που αντιπροσωπεύουν επικεφαλίδες πρωτοκόλλων (ATM, IP, UDP headers).

Συμπερασματικά από τα προηγούμενα προκύπτει ότι η μετάδοση δεδομένων έχει αυξημένες επεξεργαστικές ανάγκες και συνεπώς η απόδοση του εξυπηρετητή, όπως είναι φυσικό, εξαρτάται από το υπολογιστικό σύστημα που το υποστηρίζει. Η πλήρης δέσμευση του επεξεργαστή για την επίτευξη της μέγιστης απόδοσης, αναδεικνύει τον κίνδυνο να αποτελεί το υποσύστημα μετάδοσης δεδομένων της υπηρεσίας ροής πιθανό σημείο συμφόρησης του συστήματος.

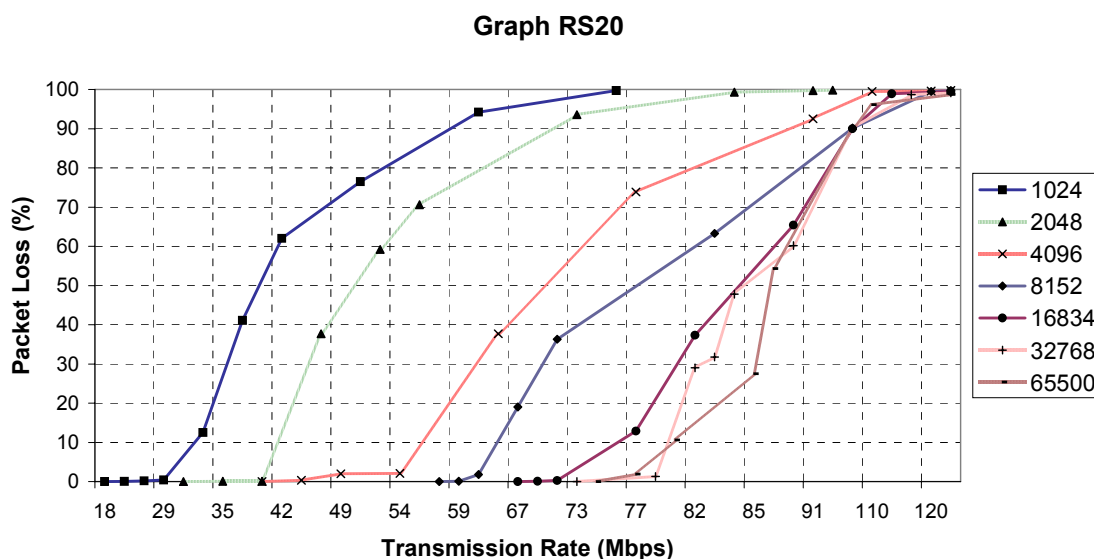
Στο επόμενο κεφάλαιο εφαρμόζουμε διάφορες πολιτικές δρομολόγησης της αποστολής δεδομένων στον εξυπηρετητή μετάδοσης. Στα πειράματα που διενεργούνται αντικειμενικός στόχος είναι η καταγραφή των αποτελεσμάτων και της ποιότητας των υπηρεσιών ροής στην πλευρά του πελάτη, ανάλογα με τις εφαρμοζόμενες τεχνικές δρομολόγησης στον εξυπηρετητή.

5.4 Πειράματα λήψης δεδομένων στον πελάτη

Σε αυτή την ενότητα περιγράφονται μια σειρά από πειράματα που έλαβαν χώρα σε συστήματα ετερογενών πελατών, ώστε να αναδείξουμε τις απαιτήσεις που προκύπτουν, για

την διατήρηση της ποιότητας των υπηρεσιών ροής σε ικανοποιητικά επίπεδα, στην πλευρά του πελάτη.

Τα πειράματα που ακολουθούν, έτρεξαν σε διαφορετικά υπολογιστικά συστήματα κάτω από διαφορετικές συνθήκες. Στόχος των πειραματικών εφαρμογών ήταν να συγκεντρώσουμε πληροφορία, η οποία θα πρέπει να ενσωματωθεί στον μηχανισμό δρομολόγησης του εξυπηρετητή μετάδοσης, ώστε να έχουμε καλύτερα αποτελέσματα από άλλες προσεγγίσεις. Η πληροφορία αυτή σχετίζεται με τον μέγιστο ρυθμό μετάδοσης δεδομένων, με τον οποίο ο εξυπηρετητής μετάδοσης μπορεί να αποστέλλει πακέτα, χωρίς να δημιουργούνται προβλήματα στον πελάτη. Και σε αυτά τα πειράματα χρησιμοποιήθηκε η αρχιτεκτονική υλικού που παρουσιάστηκε στην ενότητα 5.1.

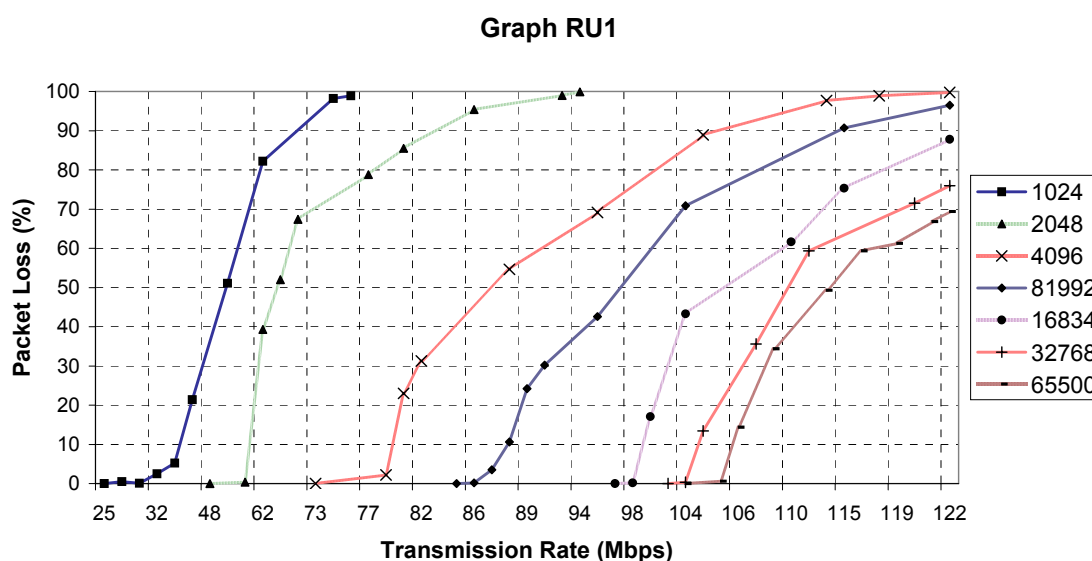


Σχήμα 36: Ρυθμός λήψης δεδομένων από τον πελάτη (Sparc20, Μετρήσεις Πειράματος)
Το παραπάνω διάγραμμα περιγράφει γραφικά τις μετρήσεις που έλαβαν χώρα σε υπολογιστή SparcStation 20, ο οποίος είχε τον ρόλο του πελάτη στο πείραμα. Μέσω του ATM δικτύου ο πελάτης ζητάει την αναπαραγωγή ενός τυχαίου αντικειμένου ροής. Ο εξυπηρετητής στέλνει τα δεδομένα με διάφορους ρυθμούς μετάδοσης και διαφορετικά μεγέθη πακέτων. Η καλύτερη απόδοση που καταγράφηκε στον πελάτη αντιστοιχεί σε ρυθμό αποστολής δεδομένων 75Mbps με πακέτα μεγέθους 65500 bytes και 0% απώλειες πακέτων.

Στο πρώτο πείραμα ο εξυπηρετητής τρέχει σε έναν Ultra One υπολογιστή. Δημιουργήθηκαν επτά διαφορετικά μεγέθη πακέτων δεδομένων που χρησιμοποιήθηκαν από τον εξυπηρετητή για την μετάδοση δεδομένων (από 1024-65500 bytes). Ο εξυπηρετητής επιλέγει το μικρότερο μέγεθος πακέτων και αρχίζει να αποστέλλει δεδομένα με το μέγιστο δυνατό ρυθμό μετάδοσης. Στην πλευρά του πελάτη καταγράφεται το ποσοστό απώλειας πακέτων κατά την διάρκεια του πειράματος. Στην συνέχεια ο εξυπηρετητής μετάδοσης επαναλαμβάνει την μετάδοση με μικρότερο ρυθμό αποστολής δεδομένων. Αυτή η διαδικασία συνεχίζεται μέχρι να βρεθεί ο κατάλληλος ρυθμός μετάδοσης για τον οποίο δεν παρατηρούνται απώλειες πακέτων στον πελάτη. Όλες οι παραπάνω ενέργειες επαναλαμβάνονται επίσης, για όλα τα μεγέθη των πακέτων που έχουν προκαθοριστεί. Στην συνέχεια της ενότητας το ίδιο πείραμα έλαβε χώρα για διαφορετικά είδη πελατών.

Στο Σχήμα 36 καταγράφονται τα αποτελέσματα των μετρήσεων σε ένα υπολογιστή SparcStation 20 που αποτέλεσε τον πελάτη. Ο κάθετος άξονας αναπαριστά το ποσοστό απώλειας πακέτων, ο οριζόντιος άξονας τους ρυθμούς μετάδοσης δεδομένων, ενώ κάθε καμπύλη αντιπροσωπεύει τα αποτελέσματα για διαφορετικού μεγέθους πακέτα. Όπως προκύπτει από το σχήμα, η καλύτερη απόδοση για τον συγκεκριμένο πελάτη-υπολογιστή σημειώνεται όταν ο εξυπηρετητής αποστέλλει πακέτα δεδομένων μεγέθους 65500 Bytes με ρυθμό μετάδοσης 75 Mbps.

Από το αποτέλεσμα αυτού του πειράματος μπορεί κανείς να εξηγήσει φαινόμενα απώλειας πακέτων, όταν δεν εφαρμόζεται η κατάλληλη δρομολόγηση στην αποστολή των δεδομένων (όπως τα παραδείγματα που θα ακολουθήσουν στις ενότητες 6.4.2, 6.4.3, 6.4.4).



Σχήμα 37: Ρυθμός λήψης δεδομένων στον πελάτη (Ultra One-Μετρήσεις Πειράματος)
Το σχήμα αυτό περιγράφει γραφικά τις μετρήσεις που έλαβαν χώρα σε υπολογιστή Ultra One, ο οποίος είχε τον ρόλο του πελάτη στο πείραμα. Μέσω του ATM δικτύου ο πελάτης ζητάει την αναπαραγωγή ενός τυχαίου αντικειμένου ροής. Ο εξυπηρετητής στέλνει τα δεδομένα με διάφορους ρυθμούς μετάδοσης και διαφορετικά μεγέθη πακέτων. Το καλύτερο αποτέλεσμα που καταγράφηκε στον πελάτη είναι ο ρυθμός αποστολής δεδομένων 104Mbps με πακέτα μεγέθους 65500 bytes καταγράφοντας 0% απώλειες πακέτων.

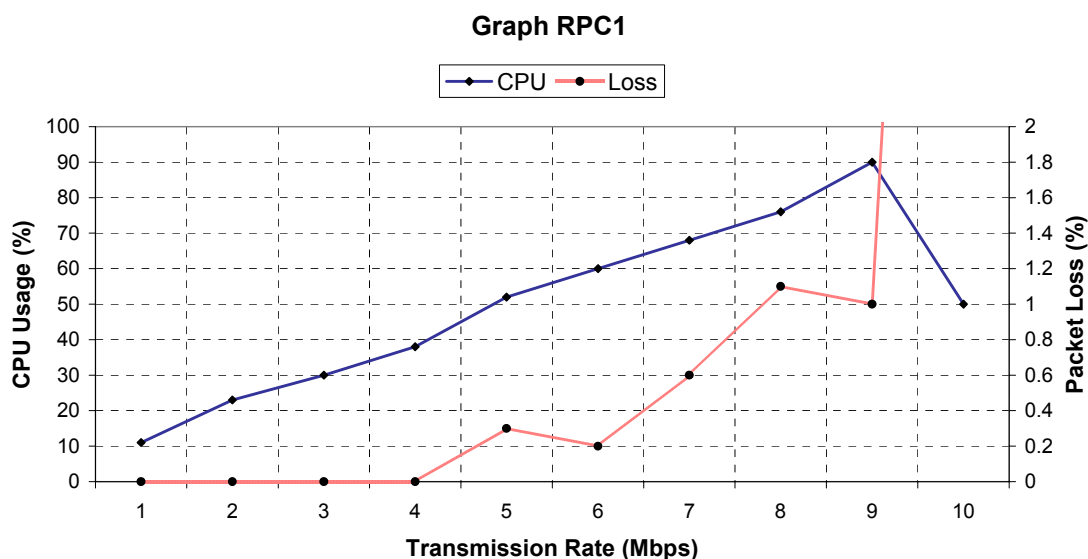
Στο Σχήμα 37 καταγράφουμε τα αποτελέσματα των μετρήσεων αλλάζοντας το υλικό του πελάτη που τώρα τρέχει σε ένα Ultra One υπολογιστή. Παρατηρώντας το διάγραμμα γίνεται εύκολα αντιληπτό ότι σε αυτό το πείραμα ο πελάτης καταφέρνει να δεχθεί δεδομένα με μεγαλύτερο ρυθμό, σε σύγκριση με τον προηγούμενο πελάτη (SparcStation 20), για όλα τα μεγέθη των πακέτων. Η μεγαλύτερη απόδοση εμφανίζεται για το μεγαλύτερο πακέτο, καθώς ο πελάτης επιτυγχάνει να δέχεται δεδομένα με ρυθμό 104Mbps χωρίς απώλειες.

Από τα προηγούμενα αναδεικνύεται η σχέση της επεξεργαστικής ισχύος στον πελάτη με την ικανότητα του για λήψη δεδομένων, καθώς καταγράφηκαν διαφορετικές επιδόσεις σε ετερογενείς πελάτες, διατηρώντας σταθερό το υλικό δικτύου, το λειτουργικό σύστημα και το λογισμικό του πελάτη. Αυτή η σχέση του υλικού του πελάτη με την απόδοσή του πρέπει να είναι γνωστή στον δρομολογητή του εξυπηρετητή πολυμέσων, ώστε να μπορεί να εγγυηθεί την ποιότητα υπηρεσιών σε όλους τους πελάτες. Ένα άλλο πλεονέκτημα αποτελεί το

γεγονός ότι ο δρομολογητής, έχοντας την πληροφορία των χαρακτηριστικών των ενεργών πελατών, θα είναι σε θέση να εκμεταλλευτεί ορισμένες δυνατότητές τους για λόγους καλύτερης απόδοσης, όπως θα δούμε σε επόμενα κεφάλαια.

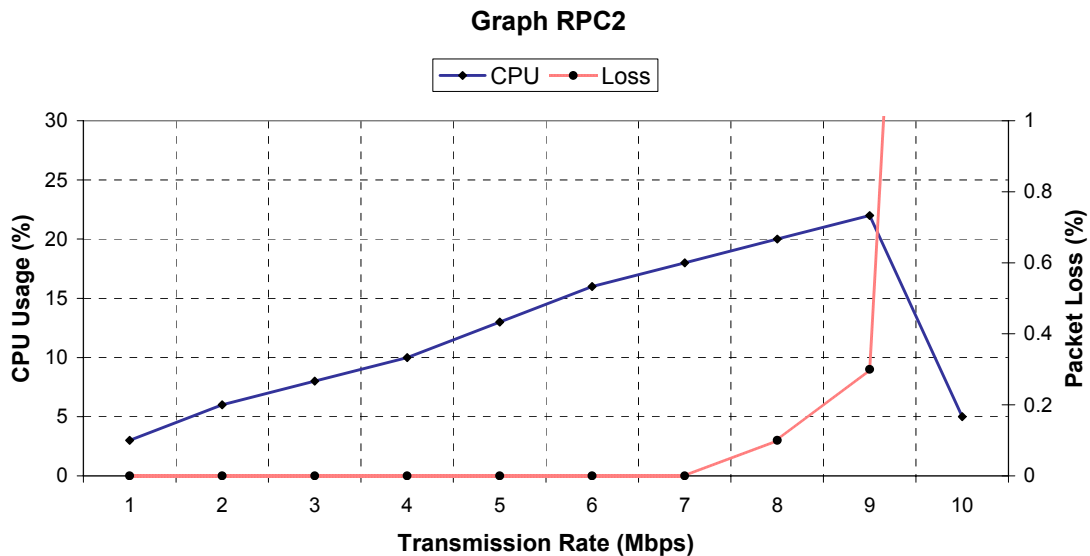
Για να μπορέσουμε να έχουμε μια καλύτερη εικόνα των διαφορετικών απαιτήσεων που προβάλλουν ετερογενείς πελάτες στην υπηρεσία ροής, τρέχουμε ανάλογα πειράματα σε διαφορετικούς προσωπικούς υπολογιστές, οι οποίοι φέρουν το λειτουργικό σύστημα Windows NT και είναι διασυνδεδεμένοι με τον εξυπηρετητή μέσω δύο διακοπών, ενός ATM και ενός ETHER διακόπτη. Από τα προηγούμενα αποτελέσματα, καταλήξαμε στο συμπέρασμα ότι η επεξεργαστικής ισχύς του πελάτη, επηρεάζει την απόδοσή του στην λήψη δεδομένων. Έτσι στα πειράματα που ακολουθούν καταγράφουμε μεταξύ άλλων και μετρήσεις σχετικές με την χρήση του επεξεργαστή.

Οι προσωπικοί υπολογιστές είναι εφοδιασμένοι με κάρτες δικτύου ονομαστικής ικανότητας 10Mbps. Στα πειράματα χρησιμοποιούνται πακέτα μεγέθους 8192 bytes και καταγράφονται ο ρυθμός μετάδοσης δεδομένων από τον εξυπηρετητή, η απώλεια πακέτων που παρατηρείται στην πλευρά του πελάτη και το ποσοστό χρήσης του επεξεργαστή από τις διεργασίες της λήψης δεδομένων.

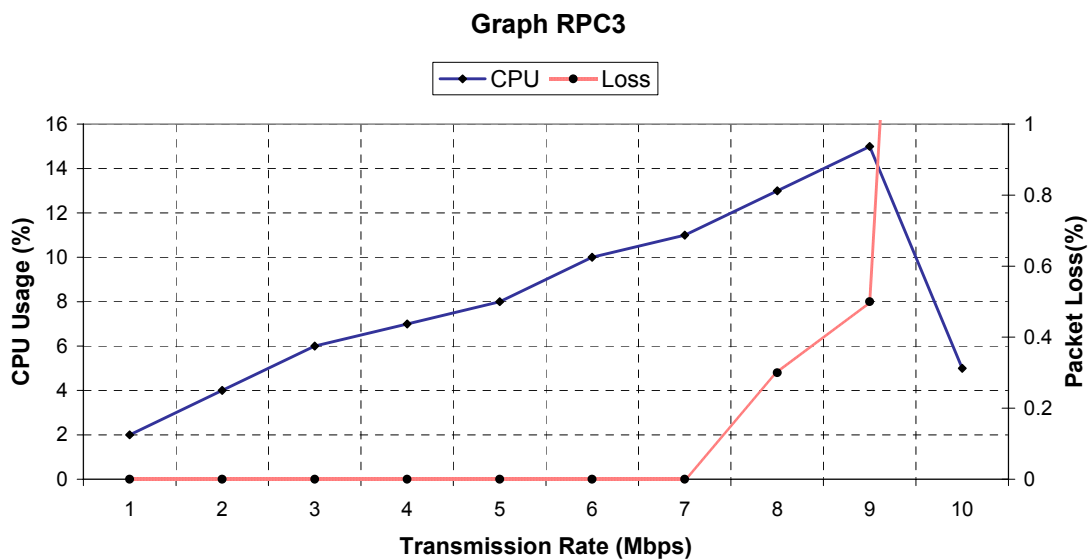


Σχήμα 38: Ρυθμός λήψης δεδομένων στον πελάτη (PC1, Μετρήσεις Πειράματος)
Στο πείραμα αυτό το λογισμικό του πελάτη έτρεξε σε ένα PC με επεξεργαστή Intel Pentium 166 Mhz. Μέσω των διακοπών δικτύου ATM και ETHER, ο πελάτης ζητάει την αναπαραγωγή ενός τυχαίου αντικειμένου ροής. Ο εξυπηρετητής στέλνει τα δεδομένα με διάφορους ρυθμούς μετάδοσης από τον μικρότερο προς τον μεγαλύτερο. Για κάθε εφαρμοζόμενο ρυθμό μετάδοσης καταγράφονται ταυτόχρονα το ποσοστό απώλειας πακέτων και το ποσοστό χρήσης του επεξεργαστή.

Στο Σχήμα 38 παρουσιάζονται τα αποτελέσματα του πειράματος όταν ο πελάτης τρέχει σε ένα PC με επεξεργαστή Intel Pentium 166 Mhz. Από το σχήμα βγαίνει το συμπέρασμα ότι οι διεργασίες λήψης δεδομένων δεσμεύουν σημαντικό ποσοστό της ισχύος του επεξεργαστή, ανάλογα με τον ρυθμό άφιξης των δεδομένων, το οποίο κυμαίνεται από 10% μέχρι 90%. Απώλειες πακέτων αρχίζουν για ρυθμούς μετάδοσης μεγαλύτερους από 4Mbps, οι οποίες όμως περιορίζονται σε χαμηλά επίπεδα.



Σχήμα 39: Ρυθμός λήψης δεδομένων στον πελάτη (PC2, Μετρήσεις Πειράματος)
 Στο πείραμα αυτό το λογισμικό του πελάτη έτρεξε σε ένα PC με επεξεργαστή Intel Pentium Pro. Μέσω των διακοπών δικτύου ATM και ETHER, ο πελάτης ζητάει την αναπαραγωγή ενός τυχαίου αντικειμένου ροής. Ο εξυπηρετητής στέλνει τα δεδομένα με διάφορους ρυθμούς μετάδοσης από τον μικρότερο προς τον μεγαλύτερο. Για κάθε εφαρμοζόμενο ρυθμό μετάδοσης καταγράφονται ταυτόχρονα το ποσοστό απώλειας πακέτων και το ποσοστό χρήσης του επεξεργαστή.



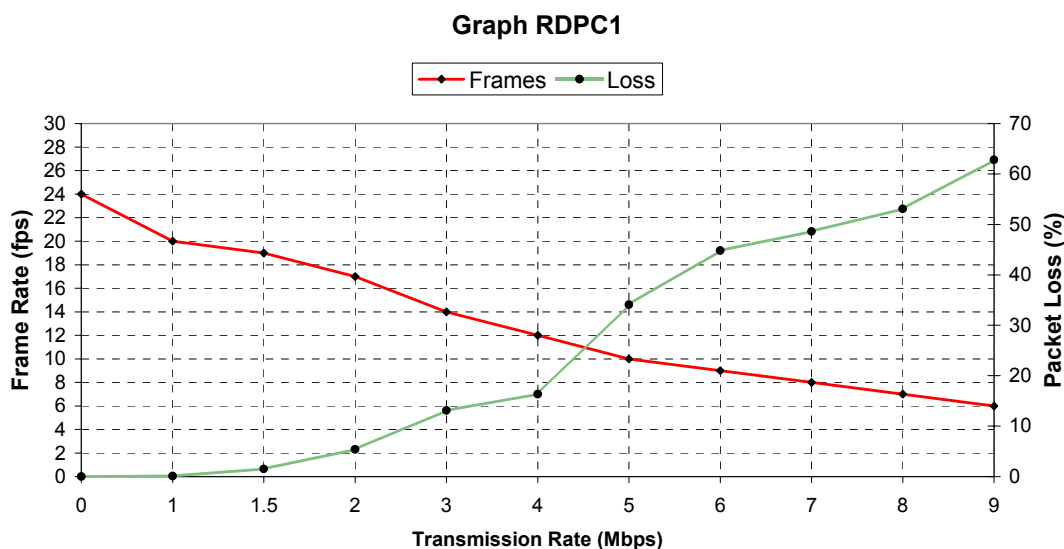
Σχήμα 40: Ρυθμός λήψης δεδομένων στον πελάτη (PC3, Μετρήσεις Πειράματος)
 Στο πείραμα αυτό το λογισμικό του πελάτη έτρεξε σε ένα PC με επεξεργαστή Intel Pentium II 433 Mhz. Ο εξυπηρετητής στέλνει τα δεδομένα με διάφορους ρυθμούς μετάδοσης από τον μικρότερο προς τον μεγαλύτερο. Για κάθε εφαρμοζόμενο ρυθμό μετάδοσης καταγράφονται ταυτόχρονα το ποσοστό απώλειας πακέτων και το ποσοστό χρήσης του επεξεργαστή. Στο τρέχων σύστημα για την προσέγγιση της μέγιστου ρυθμού λήψης δεδομένων, που του επιτρέπει η δικτυακή του υποδομή, απαιτείται δέσμευση του 15% περίπου της επεξεργαστικής ισχύος.

Στην συνέχεια επαναλαμβάνουμε την ίδια διαδικασία επιλέγοντας ένα PC με επεξεργαστή Intel Pentium Pro 200Mhz. Η μεγαλύτερη επεξεργαστική ικανότητα του υπολογιστή πελάτη που χρησιμοποιήθηκε, εμφανίζεται άμεσα στο Σχήμα 39, καθώς για τον μέγιστο δυνατό ρυθμό λήψης δεδομένων (9Mbps), ο επεξεργαστής απασχολείται σε

ποσοστό 22% και οι απώλειες πακέτων εμφανίζονται σε μεγαλύτερους ρυθμούς μετάδοσης και σε χαμηλότερα επίπεδα.

Από τα προηγούμενα πειράματα συμπεραίνουμε ότι κάθε πελάτης ανάλογα με το υλικό και την τεχνολογία την οποία διαθέτει έχει διαφορετικές απαιτήσεις και ικανότητες. Έτσι κάθε πελάτης έχει ένα μέγιστο όριο στον ρυθμό δεδομένων που μπορεί να δεχθεί χωρίς απώλειες πληροφορίας. Το όριο αυτό εξαρτάται από το είδος του δικτύου και της αντίστοιχης κάρτας, καθώς επίσης και από την επεξεργαστική ισχύ του υπολογιστικού συστήματος.

Σε όλα τα προηγούμενα αποτελέσματα ο επεξεργαστής ασχολούνταν αποκλειστικά με τις διεργασίες λήψης δεδομένων. Σε πραγματικές συνθήκες όμως, το λογισμικό στην πλευρά του πελάτη είναι υπεύθυνο και για άλλες εργασίες. Για παράδειγμα ή εφαρμογή του πελάτη ενός συστήματος video κατ' απαίτηση έχει τις ακόλουθες αρμοδιότητες. Ένα τμήμα του λογισμικού είναι υπεύθυνο για την λήψη των δεδομένων από το δίκτυο και την αποθήκευσή τους σε χώρους είτε στην κύρια μνήμη είτε σε τοπικούς δίσκους. Άλλο τμήμα λογισμικού προσπελαύνει τα αποθηκευμένα δεδομένα και εφαρμόζει τον αλγόριθμο αποκωδικοποίησης MPEG. Το τελευταίο τμήμα λογισμικού στην πλευρά του πελάτη δέχεται τα αποκωδικοποιημένα δεδομένα και αναλαμβάνει την παρουσίασή τους μέσω των συσκευών εξόδου. Είναι επομένως φυσικό ότι όλες αυτές οι διεργασίες δεσμεύουν επεξεργαστική ισχύ, ιδιαίτερα το τμήμα αποκωδικοποίησης MPEG, γεγονός που όπως είναι φυσικό επηρεάζει την τελική ικανότητα του πελάτη για λήψη δεδομένων.



Σχήμα 41: Ρυθμός λήψης και ρυθμός παρουσίαση καρέ (PC1, Μετρήσεις Πειράματος)
Στο σχήμα εικονίζονται τα αποτελέσματα όταν το λογισμικό του πελάτη τρέχει σε ένα PC με επεξεργαστή Intel Pentium 166Mhz. Χωρίς αποστολή δεδομένων επιτυγχάνεται η αποκωδικοποίηση και η παρουσίαση ενός πειραματικού video με ρυθμό 24 καρέ ανά δευτερόλεπτο (fps). Αρχίζοντας την μετάδοση δεδομένων ο ρυθμός παρουσίασης μειώνεται και πολύ γρήγορα εμφανίζονται σημαντικές απώλειες πακέτων.

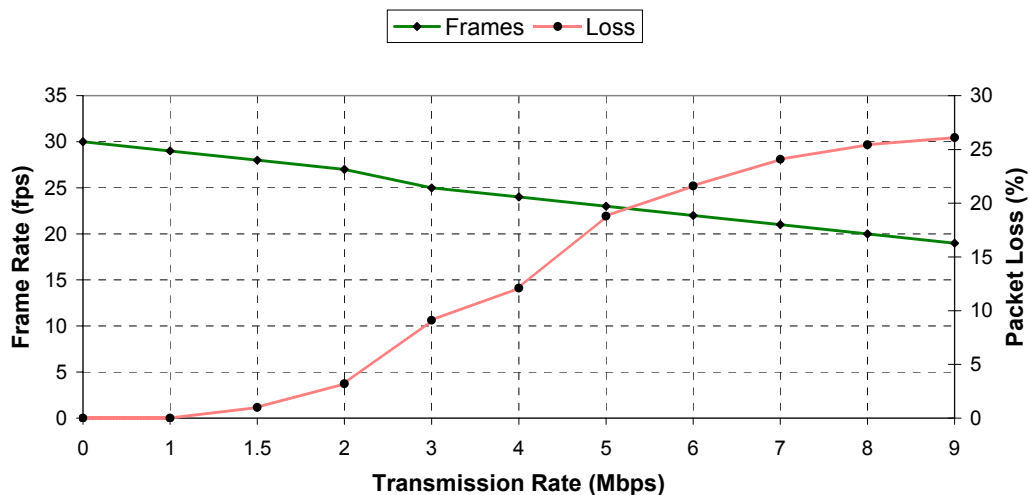
Στην συνέχεια της ενότητας ακολουθούν μερικά ενδεικτικά πειράματα που λαμβάνουν χώρα χρησιμοποιώντας το ίδιο υλικό πελατών με την διαφορά ότι τώρα στην πλευρά του πελάτη εκτός από την διεργασία λήψης δεδομένων, συνυπάρχουν οι διεργασίες

αποθήκευσης, αποκωδικοποίησης και παρουσίασης του αντικειμένου ροής. Τα διαγράμματα που παρουσιάζονται παρακάτω εικονίζουν τα αποτελέσματα των μετρήσεων. Ο οριζόντιος άξονας φέρει τους ρυθμούς μετάδοσης δεδομένων που εφαρμόστηκαν από τον εξυπηρετητή, ο πρωτεύων κάθετος άξονας αντιπροσωπεύει τον ρυθμό παρουσίασης των καρέ του video (frame rate) και τέλος ο δευτερεύων κάθετος άξονας αναπαριστά το ποσοστό απώλειας πακέτων στην πλευρά του πελάτη.

Το Σχήμα 41 παρουσιάζει τα αποτελέσματα στην πλευρά του πελάτη, όταν χρησιμοποιούμε ένα PC με επεξεργαστή Intel Pentium 166Mhz και κάρτα Ethernet. Το λογισμικό αποκωδικοποίησης MPEG και παρουσίασης επιτυγχάνει την αναπαραγωγή του μέσου με ρυθμό 24fps. Όταν ο εξυπηρετητής αποστέλλει δεδομένα με ρυθμό 1Mbps, τότε ο ρυθμός παρουσίασης μειώνεται στα 20fps. Αυξάνοντας συνεχώς τον ρυθμό μετάδοσης έχουμε αντίστοιχη μείωση στον ρυθμό παρουσίασης και εμφάνιση αρκετών απωλειών δεδομένων. Γίνεται επομένως εύκολα κατανοητό ότι όταν ο πελάτης είναι εξοπλισμένος με υλικό μικρών δυνατοτήτων ο αυξημένος ρυθμός μετάδοσης δεδομένων από την υπηρεσία ροής καταλήγει σε μείωση της ποιότητας υπηρεσιών στην πλευρά του πελάτη.

Τρέχοντας το ίδιο πείραμα σε ένα ταχύτερο υπολογιστή, ο οποίος επιτυγχάνει αποκωδικοποίηση και παρουσίαση με ρυθμό 30fps και εφαρμόζοντας στον εξυπηρετητή ρυθμό μετάδοσης 1.5 Mbps, έχουμε σαν αποτέλεσμα μια μικρή μείωση στον ρυθμό παρουσίασης (28Mbps) και απώλειες πακέτων της τάξης του 1% (Σχήμα 42).

Graph RDPC2

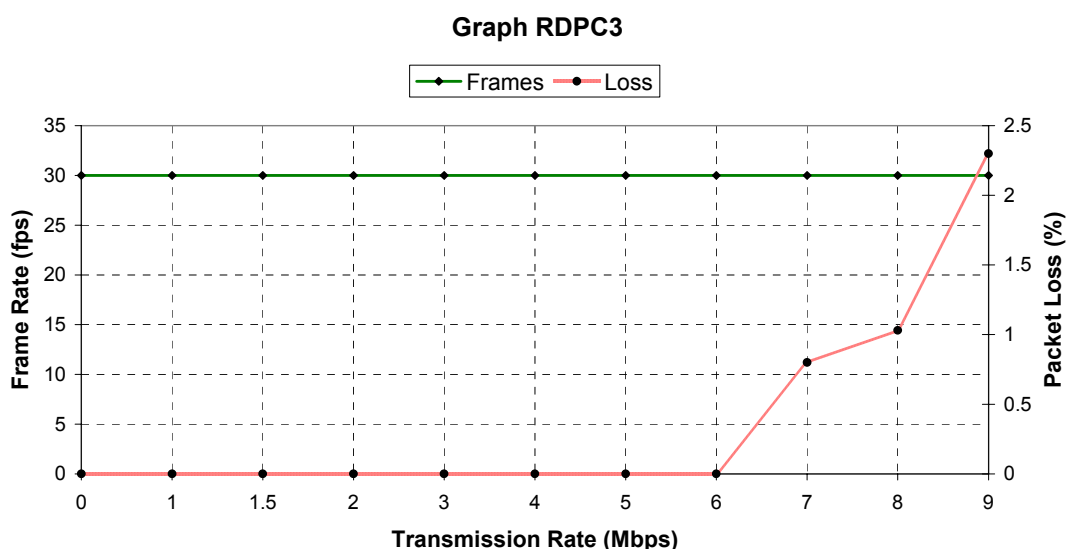


Σχήμα 42: Ρυθμός λήψης και ρυθμός παρουσίαση καρέ (PC2, Μετρήσεις Πειράματος)
Στο σχήμα εικονίζονται τα αποτελέσματα όταν το λογισμικό του πελάτη τρέχει σε ένα PC με επεξεργαστή Intel Pentium Pro 200Mhz. Χωρίς αποστολή δεδομένων επιτυγχάνεται η αποκωδικοποίηση και η παρουσίαση ενός πειραματικού video με ρυθμό 30 καρέ ανά δευτερόλεπτο (fps). Όταν ο εξυπηρετητής αποστέλλει δεδομένα με ρυθμό 1.5Mbps, τότε στην πλευρά του πελάτη παρατηρείται μείωση στο ρυθμό παρουσίασης (28fps) και απώλειες πακέτων σε ποσοστό 0.1%.

Για να αναδείξουμε τις διαφορετικές ανάγκες εξυπηρέτησης σε περιβάλλοντα ετερογενών πελατών χρησιμοποιούμε ένα ακόμα πελάτη που τρέχει σε PC με επεξεργαστή Intel Pentium II 433Mhz. Τα αποτελέσματα αυτής της ενέργειας εικονίζονται στο Σχήμα 43.

Η αυξημένη επεξεργαστική ισχύς του πελάτη επιτρέπει την αποκωδικοποίηση και παρουσίαση του μέσου με κανονικό ρυθμό 30fps, ακόμα και όταν ο εξυπηρετητής μεταδίδει δεδομένα με τον μέγιστο ρυθμό (9Mbps), με τον οποίο η δικτυακή υποδομή στον πελάτη μπορεί να δεχθεί, διατηρώντας τις απώλειες σε ποσοστά κάτω του 2.5%.

Πρέπει βέβαια να σημειωθεί ότι σε αυτήν την περίπτωση, οι απώλειες πακέτων που παρατηρούνται οφείλονται στις περιορισμένες δυνατότητες του δικτυακού εξοπλισμού του πελάτη.



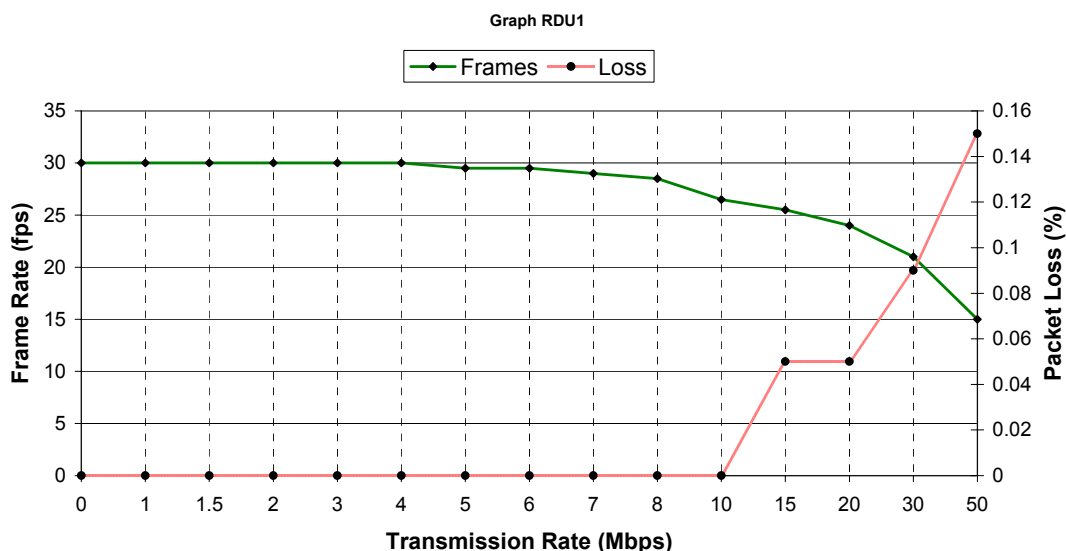
Σχήμα 43: Ρυθμός λήψης και ρυθμός παρουσίαση καρέ (PC3, Μετρήσεις Πειράματος)
Στο σχήμα εικονίζονται τα αποτελέσματα όταν το λογισμικό του πελάτη τρέχει σε ένα PC με επεξεργαστή Intel Pentium II 433Mhz. Χωρίς αποστολή δεδομένων επιτυγχάνεται η αποκωδικοποίηση και η παρουσίαση ενός πειραματικού video με ρυθμό 30 καρέ ανά δευτερόλεπτο (fps). Η αυξημένη επεξεργαστική ικανότητα του υπολογιστή επιτρέπει την διατήρηση του ρυθμού παρουσίασης στα 30fps ακόμα και για το μέγιστο ρυθμό λήψης δεδομένων, εμφανίζοντας μικρές απώλειες δεδομένων για ρυθμούς μετάδοσης μεγαλύτερους των 6Mbps.

Αλλάζοντας την δικτυακή υποδομή, το υπολογιστικό και λειτουργικό σύστημα του πελάτη καταγράφουμε τις μετρήσεις, οι οποίες παρουσιάζονται γραφικά στο Σχήμα 44. Στην περίπτωση αυτή το λογισμικό του πελάτη τρέχει σε υπολογιστή Ultra One με λειτουργικό σύστημα Solaris 2.5.1, συνδεδεμένος με τον εξυπηρετητή μέσω του τοπικού δικτύου ATM. Το υλικό επιτρέπει την αποκωδικοποίηση και παρουσίαση του μέσου με ρυθμό 30fps, όταν ο επεξεργαστής δεν έχει άλλες εργασίες. Με τις συγκεκριμένες ρυθμίσεις ο πελάτης επιτυγχάνει να δέχεται δεδομένα με ρυθμούς μέχρι 10Mbps, χωρίς απώλειες, αλλά με μικρή μείωση του ρυθμού παρουσίασης (26.5fps). Για ακόμα μεγαλύτερους ρυθμούς μετάδοσης (μέχρι 50 Mbps) παρατηρούμε δραστική μείωση του ρυθμού παρουσίασης αλλά οι απώλειες πακέτων εξακολουθούν να είναι πολύ μικρές (< 0.15%).

Έχοντας υπόψη τα αποτελέσματα που προέκυψαν από την μελέτη των προηγούμενων πειραμάτων, προσπαθούμε να εφαρμόσουμε μια στρατηγική δρομολόγησης στον εξυπηρετητή μετάδοσης, η οποία θα πρέπει να γνωρίζει τις δυνατότητες και τον τρόπο λειτουργίας των πελατών που ζητούν εξυπηρέτηση. Ο μηχανισμός δρομολόγησης της μετάδοσης δεδομένων σχεδιάζεται στο 6ο κεφάλαιο με βασικό γνώμονα να είναι ικανός να

οδηγεί πολλαπλές ροές αποστολής δεδομένων προς ετερογενείς πελάτες, με προδιαγραφές που περιλαμβάνουν σταθερό ρυθμό αποστολής δεδομένων και μειωμένο jitter μετάδοσης, έτσι ώστε να εξασφαλίζεται η ποιότητα υπηρεσιών στην πλευρά του πελάτη.

Στο 7ο κεφάλαιο παρουσιάζονται συμπληρωματικές εργασίες που οδηγούν σε μια ολοκληρωμένη αντιμετώπιση της δρομολόγησης των πολλαπλών ροών δεδομένων, η οποία λαμβάνει υπόψη της τα ιδιαίτερα χαρακτηριστικά των πελατών, με απώτερο σκοπό την καλύτερη αξιοποίηση των διαθέσιμων πόρων και την αύξηση της συνολικής απόδοσης του συστήματος.



Σχήμα 44: Ρυθμός λήψης και ρυθμός παρουσίασης καρέ (Ultra One, Μετρήσεις Πειράματος)

Στο σχήμα εικονίζονται τα αποτελέσματα όταν το λογισμικό του πελάτη τρέχει σε Ultra One υπολογιστή. Χωρίς αποστολή δεδομένων επιτυγχάνεται η αποκωδικοποίηση και η παρουσίαση ενός πειραματικού video με ρυθμό 30 καρέ ανά δευτερόλεπτο (fps). Ο εξυπηρετητής αποστέλλει πακέτα δεδομένων μέσω του τοπικού δικτύου ATM. Το σύστημα του πελάτη επιτυγχάνει να παρουσιάζει το video στον κανονικό ρυθμό και να δέχεται ταυτόχρονα χωρίς απώλειες δεδομένα με ρυθμό 4Mbps. Για μεγαλύτερους ρυθμούς μετάδοσης αρχίζει να μειώνεται ο ρυθμός παρουσίασης, ενώ αργότερα εμφανίζονται μικρά ποσοστά απώλειας δεδομένων.

5.5 Πειράματα απόδοσης στο υποσύστημα ανάκτησης δεδομένων

Στις προηγούμενες ενότητες μελετήσαμε την συμπεριφορά του υποσυστήματος της μετάδοσης δεδομένων. Εξετάσαμε τις διαφορετικές απαιτήσεις που εισάγουν στην υπηρεσία ροής ετερογενείς πελάτες και τις μέγιστες επιδόσεις μετάδοσης δεδομένων που μπορούν να επιτευχθούν.

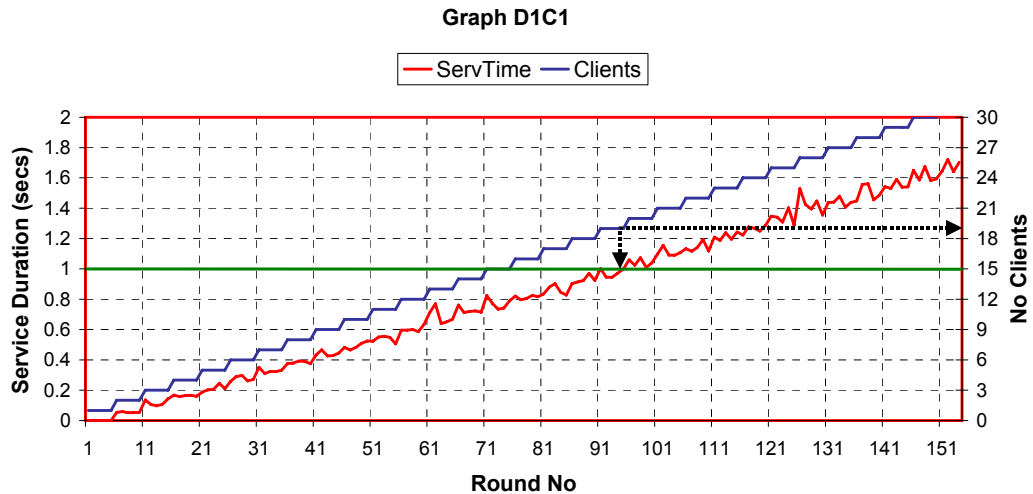
Αυτή η ενότητα εστιάζεται στο υποσύστημα ανάκτησης δεδομένων, απομονώνοντας την λειτουργία της μετάδοσης. Στόχος της ενότητας αποτελεί η παρατήρηση της συμπεριφοράς και η καταγραφή των επιδόσεων των μέσων αποθήκευσης που διαθέτουμε, πάνω στα οποία θα στηριχθεί η λειτουργία της υπηρεσίας ροής. Το υποσύστημα ανάκτησης περιγράφεται με λεπτομέρεια στο επόμενο κεφάλαιο. Αποτελείται από τους εξυπηρετητές

δίσκου, οι οποίοι υποστηρίζονται από τον διαχειριστή συσκευών και η λειτουργία τους συντονίζεται με την βοήθεια του κεντρικού δρομολογητή. Το σύστημα οδηγεί την ανάκτηση δεδομένων από τους δίσκους με κύκλους εξυπηρέτησης, έτσι ώστε σε κάθε κύκλο να ανακτάται ένα ή περισσότερα τμήματα δεδομένων για κάθε πελάτη. Σε αυτή την ενότητα επιβάλλουμε σταθερή κυκλική σειρά στην εξυπηρέτηση των πελατών.

Στα πειράματα που ακολουθούν χρησιμοποιούμε ένα υπολογιστή Ultra One, ο οποίος είναι εξοπλισμένος με δύο προσαρμοστές SCSI-2 διαύλου και τέσσερις μαγνητικούς δίσκους (2 για κάθε SCSI δίαυλο , μέγιστης ονομαστικής απόδοσης 5MBps).

Για τις ανάγκες των πειραμάτων θεωρούμε ότι έχουν αποθηκευτεί στο υποσύστημα ανάκτησης τα τμήματα των αντικειμένων ροής σε τυχαίες θέσεις στους δίσκους. Το μέγεθος του τμήματος ανάκτησης που χρησιμοποιείται για κάθε πελάτη είναι 192K, το οποίο αντιστοιχεί σε 1 sec παρουσίασης του μέσου (MPEG1-1.5Mbps). Κατά την διάρκεια του πειράματος καταγράφουμε την διάρκεια εξυπηρέτησης όλων των πελατών από τους δίσκους. Ταυτόχρονα εισάγουμε σταδιακά νέους χρήστες στο σύστημα και παρατηρούμε την αύξηση του χρόνου εξυπηρέτησης. Η διαδικασία εισαγωγής νέων πελατών συνεχίζεται μέχρι να καταγραφούν σημαντικές επιδράσεις στην ποιότητα υπηρεσιών, σε μια προσπάθεια εύρεσης του μέγιστου αριθμού χρηστών που μπορούν να εξυπηρετηθούν με σταθερή ποιότητα. Οι δίσκοι καταφέρνουν να ανταποκριθούν στην εξυπηρέτηση των ενεργών πελατών, όταν ο χρόνος εξυπηρέτησης για ένα κύκλο κυμαίνεται σε επίπεδα κάτω του 1 sec. Στο Σχήμα 45 παρουσιάζονται τα αποτελέσματα των μετρήσεων όταν όλα τα αντικείμενα ροής έχουν αποθηκευτεί σε ένα δίσκο. Ο κάθετος πρωτεύων άξονας αναπαριστά την διάρκεια εξυπηρέτησης, ενώ ο δευτερεύων κάθετος άξονας τον αριθμό των ενεργών πελατών που εξυπηρετούνται από το σύστημα. Η οριζόντια έντονη γραμμή στο επίπεδο 1 sec εμφανίζει το μέγιστο επιθυμητό όριο των χρονικών καθυστερήσεων ανάκτησης. Όταν το γράφημα του χρόνου εξυπηρέτησης υπερβεί ανοδικά το όριο αυτό, σημαίνει ότι οι συσκευές αποθήκευσης δεν μπορούν να ικανοποιήσουν τις απαιτήσεις πραγματικού χρόνου των πελατών λόγω υψηλού φόρτου εργασίας.

Από το σημείο τομής της γραμμής του ορίου στο Σχήμα 45 μπορούμε να συμπεράνουμε ότι ο δίσκος που εξετάζουμε μπορεί να επεξεργαστεί ικανοποιητικά την ανάκτηση μέχρι 19 πελατών. Όσο συνεχίζουμε να εισάγουμε νέους χρήστες πέρα από το όριο αυτό, η τιμή καθυστέρησης αρχίζει να αυξάνει αισθητά, γεγονός που όπως είναι φυσικό οδηγεί σε μείωση του ρυθμού ανάκτησης δεδομένων σε επίπεδα κάτω από τα απαιτούμενα για την αναπαραγωγή του αντικειμένου ροής.

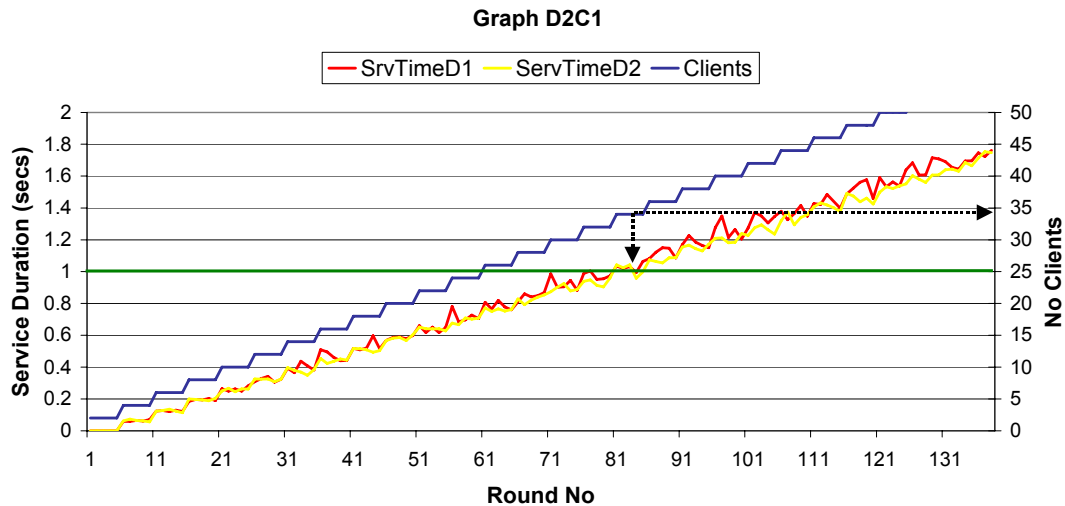


Σχήμα 45: Απόδοση των μέσων αποθήκευσης (α)

Όλα τα αντικείμενα ροής του πειράματος έχουν αποθηκευτεί σε ένα δίσκο. Το παραπάνω διάγραμμα παρουσιάζει τα αποτελέσματα των μετρήσεων για την καθυστέρηση επεξεργασίας των υποαιτήσεων ανάκτησης όλων των πελατών σε ένα κύκλο. Ταυτόχρονα προβάλλεται ο αριθμός των ενεργών πελατών που εξυπηρετούνται σε κάθε χρονική στιγμή από το σύστημα. Από το σχήμα προκύπτει ότι η υπηρεσία ροής με τις τρέχουσες ρυθμίσεις μπορεί να εξυπηρετήσει ικανοποιητικά την αναπαραγωγή 19 αντικειμένων ροής

Στην συνέχεια αλλάζουμε τον τρόπο αποθήκευσης των αντικειμένων ροής και επαναλαμβάνουμε το ίδιο πείραμα. Σε αυτή την περίπτωση χρησιμοποιούμε δύο δίσκους, τους οποίους συνδέουμε στον ίδιο δίαυλο SCSI και θεωρούμε ότι αποθηκεύουμε σε αυτούς τα αντικείμενα ροής του πειράματος, εφαρμόζοντας gross-striping τεχνική.

Τα αποτελέσματα του πειράματος εμφανίζονται στο Σχήμα 46. Παρατηρώντας την γραμμή της διάρκειας εξυπηρέτησης, καταλήγουμε στο συμπέρασμα ότι με τις ρυθμίσεις του τρέχοντος πειράματος η υπηρεσία ροής μπορεί να εξυπηρετήσει την ανάκτηση 34 πελατών με τον κατάλληλο ρυθμό. Ο μέγιστος αριθμός πελατών που προκύπτει από το πείραμα είναι μικρότερος από τον αναμενόμενο. Κι' αυτό γιατί χρησιμοποιώντας δύο δίσκους θα περίμενε κανείς τον διπλασιασμό του συνολικού ρυθμού ανάκτησης του συστήματος (δηλ. 38 πελάτες). Είναι εμφανές όμως ότι ο κοινός δίαυλος SCSI, στον οποίο βρίσκονται προσαρμοσμένοι οι δύο δίσκοι, αποτελεί σημείο συμφόρησης (bottleneck) και μειώνει την μέγιστη απόδοση που οι δίσκοι μπορούν να προσφέρουν



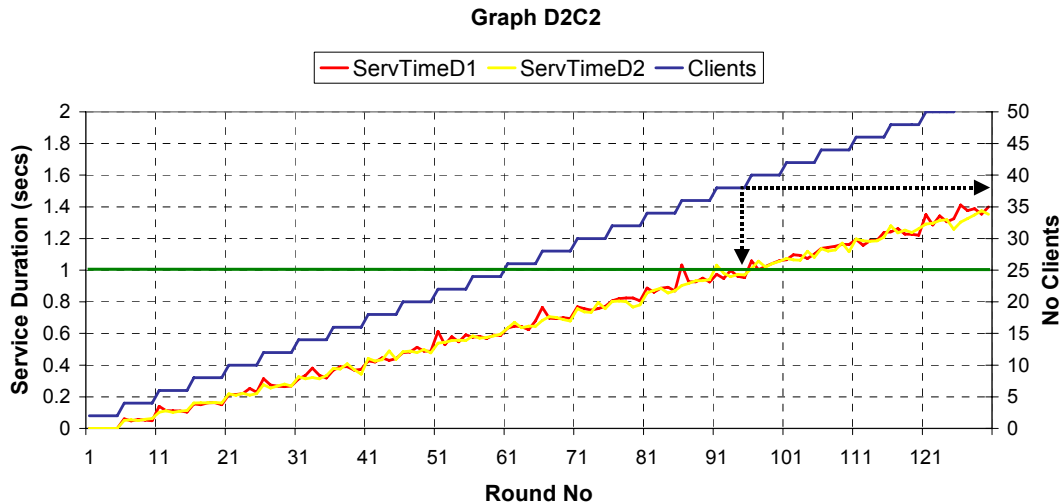
Σχήμα 46: Απόδοση των μέσων αποθήκευσης (β)

Όλα τα αντικείμενα ροής του πειράματος έχουν αποθηκευτεί σε δύο δίσκους, οι οποίοι βρίσκονται προσαρμοσμένοι στον ίδιο δίαυλο SCSI. Στο σχήμα εμφανίζονται οι χρόνοι εξυπηρέτησης των εργασιών ενός κύκλου και για τους δύο δίσκους, από τους οποίους προκύπτει ότι η υπηρεσία ροής με τις τρέχουσες ρυθμίσεις μπορεί να εξυπηρετήσει ικανοποιητικά την αναπαραγωγή 34 αντικειμένων ροής

Για να αποδείξουμε το γεγονός αυτό τρέχουμε το ίδιο πείραμα με την διαφορά ότι σε αυτή την περίπτωση χρησιμοποιούμε διαφορετικό δίαυλο SCSI για κάθε δίσκο. Οι μετρήσεις που καταγράφηκαν εικονίζονται στο Σχήμα 47. Το γράφημα επιβεβαιώνει το αναμενόμενο αποτέλεσμα, ότι δηλαδή η χρήση δύο δίσκων διπλασιάζει σχεδόν την ικανότητα εξυπηρέτησης πελατών της υπηρεσίας ροής, καθώς όπως προκύπτει από το διάγραμμα μπορούν να ικανοποιηθούν οι απαιτήσεις ανάκτησης 38 πελατών ταυτόχρονα.

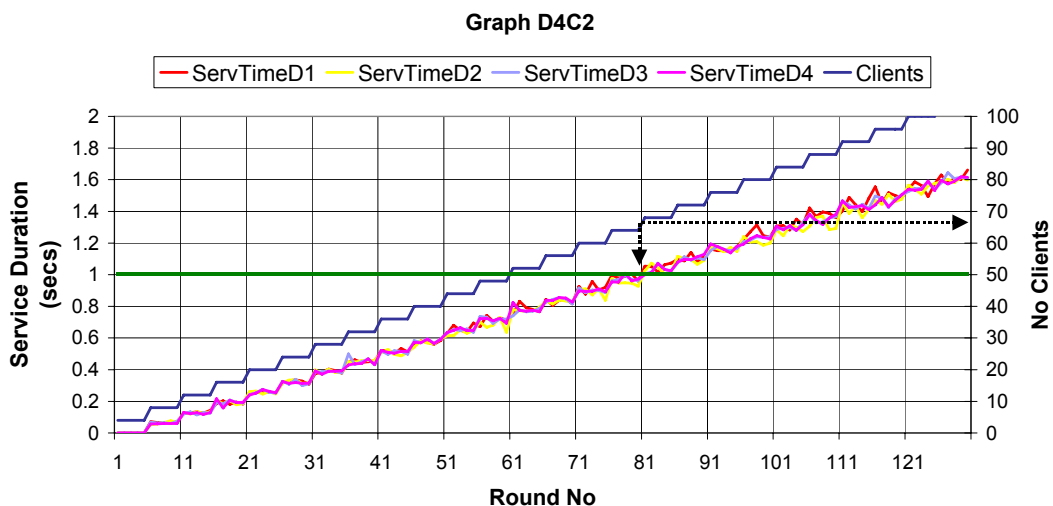
Στην συνέχεια επαναλαμβάνουμε το πείραμα για 4 δίσκους, οι οποίοι προσαρμόζονται ανά δύο σε δύο διαφορετικούς δίαυλους SCSI, σε μια προσπάθεια να καταγράψουμε την μέγιστη απόδοση του υποσυστήματος ανάκτησης δεδομένων της υπηρεσίας ροής για το υλικό που διαθέτουμε.

Από το Σχήμα 48 προκύπτει ότι 68 πελάτες μπορούν να εξυπηρετηθούν από την υπηρεσία ροής χωρίς παραβίαση των απαιτήσεων ανάκτησης δεδομένων που επιβάλλει η αναπαραγωγή των αντικειμένων ροής.



Σχήμα 47: Απόδοση των μέσων αποθήκευσης (γ)

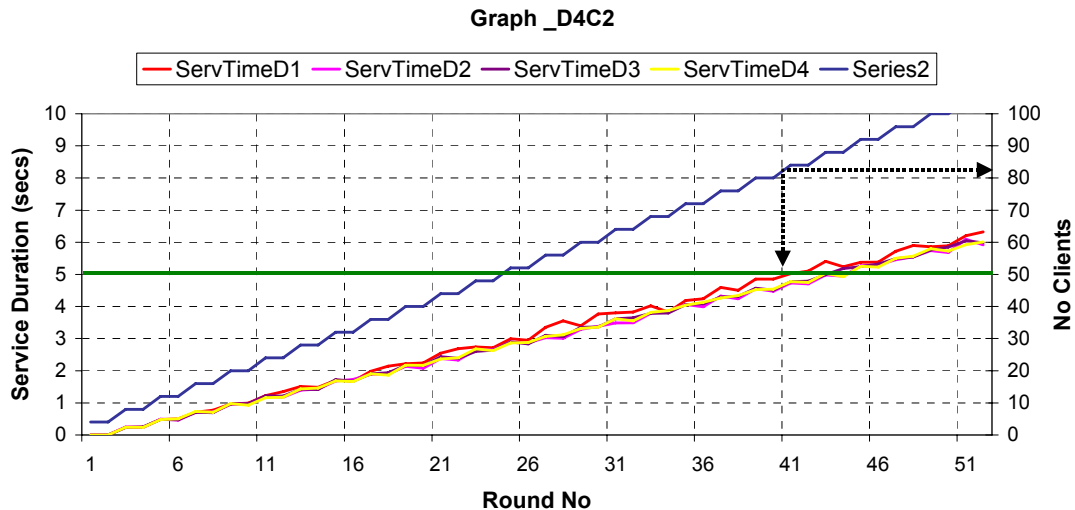
Όλα τα αντικείμενα ροής του πειράματος έχουν αποθηκευτεί σε δύο δίσκους, οι οποίοι βρίσκονται προσαρμοσμένοι σε διαφορετικό δίαυλο SCSI. Από το σχήμα προκύπτει ότι η χρήση διαφορετικού διαύλου SCSI για κάθε δίσκο έχει ως αποτέλεσμα την πλήρη αξιοποίηση των δίσκων και συνεπώς τον διπλασιασμό της απόδοσης για το σύστημα που τώρα υποστηρίζει 38 αντικείμενα ροής.



Σχήμα 48: Απόδοση των μέσων αποθήκευσης (δ)

Όλα τα αντικείμενα ροής του πειράματος έχουν αποθηκευτεί σε 4 δίσκους, οι οποίοι βρίσκονται προσαρμοσμένοι σε δύο διαφορετικούς διαύλους SCSI ανά δύο. Από το σχήμα προκύπτει ότι η υπηρεσία ροής με τις τρέχουσες ρυθμίσεις μπορεί να εξυπηρετήσει ικανοποιητικά την αναπαραγωγή 68 αντικειμένων ροής

Το μέγεθος του τμήματος ανάκτησης που χρησιμοποιείται για κάθε πελάτη, επηρεάζει σημαντικά την απόδοση των μέσων αποθήκευσης. Είναι λογικό ότι μεγαλώνοντας το μέγεθος του τμήματος αυξάνεται και η απόδοση καθώς περιορίζονται οι επιδράσεις από το κόστος αναζήτησης. Έτσι αν αυξήσουμε το μέγεθος του τμήματος που χρησιμοποιήθηκε στα προηγούμενα πειράματα από 192K σε 960K, έχουμε σαν αποτέλεσμα μια αξιοσημείωτη αύξηση των πελατών που μπορούν να υποστηρίξουν τα μέσα αποθήκευσης, ένα γεγονός όμως που έχει αρνητικό αντίκτυπο στις απαιτήσεις αποταμίευσης (όπως θα δούμε σε επόμενο κεφάλαιο) και στον χρόνο απόκρισης.



Σχήμα 49: Απόδοση των μέσων αποθήκευσης (ε)

Αυξάνοντας το τμήμα ανάκτησης για κάθε πελάτη σε ένα κύκλο εξυπηρέτησης, επιτυγχάνουμε να αυξήσουμε την απόδοση των μέσων αποθήκευσης. Όπως προκύπτει από τις πειραματικές μετρήσεις το υποσύστημα ανάκτησης (4 δίσκοι / 2 SCSI), με μέγεθος τμήματος 960K και κύκλο εξυπηρέτησης 5 sec, υποστηρίζει πάνω από 80 πελάτες.

5.6 Περίληψη

Στόχος του κεφαλαίου αυτού ήταν να εντοπιστούν όλοι εκείνοι οι παράγοντες που επηρεάζουν την απόδοση του συστήματος, αλλά και τυχόν προβλήματα που μπορούν να εμφανιστούν σε πραγματικές συνθήκες, ώστε να συμπεριληφθούν τα στοιχεία αυτά στον σχεδιασμό της υπηρεσίας διαχείρισης δεδομένων συνεχόμενης ροής.

Στα πειράματα μετάδοσης του εξυπηρετητή, που έλαβαν χώρα σε σταθμούς εργασίας Sparc4, Sparc20 και Ultra One, καταλήξαμε στο συμπέρασμα ότι ο μέγιστος ρυθμός μετάδοσης δεδομένων, που μπορεί να επιτευχθεί για συγκεκριμένη δικτυακή υποδομή, εξαρτάται σε μεγάλο βαθμό από τις επεξεργαστικές δυνατότητες του υπολογιστικού συστήματος. Επίσης ενδεχόμενη χρήση πακέτων αποστολής μικρού μεγέθους έχει αρνητική επίδραση στην συνολική απόδοση του εξυπηρετητή. Οι υπολογιστές Sparc4 και Sparc20 αποδείχθηκαν, ότι δεν είναι αρκετά ισχυροί ώστε να υποστηρίξουν την μέγιστη απόδοση της συσκευής δικτύου ATM. Ακόμα και στο σύστημα Ultra One χρειάστηκε η δέσμευση του 96% της επεξεργαστικής ισχύος για να παρατηρήσουμε την μέγιστη απόδοση 122.6Mbps. Το γεγονός αυτό αναδεικνύει τον κίνδυνο να αποτελεί ο επεξεργαστής πιθανό σημείο συμφόρησης σε ένα εξυπηρετητή πολυμέσων.

Στο κεφάλαιο αυτό έγιναν επίσης πολλά πειράματα λήψης δεδομένων από διαφορετικούς πελάτες. Στόχος των πειραμάτων αυτών ήταν να συγκεντρωθεί πληροφορία, η οποία θα πρέπει να ενσωματωθεί στον μηχανισμό δρομολόγησης του εξυπηρετητή ροής, ώστε να έχουμε τα καλύτερα αποτελέσματα. Ο υπολογιστής Sparc20 παρουσίασε απόδοση λήψης δεδομένων χωρίς απώλειες με ρυθμό 75Mbps, ενώ ο Ultra One παρουσίασε αντίστοιχη μέγιστη απόδοση 104Mbps. Αντίστοιχα πειράματα σε προσωπικούς υπολογιστές

διασυνδεδεμένους με δίκτυο Ethernet, έδειξαν ανάλογα αποτελέσματα. Χαρακτηριστικό είναι επίσης ότι ο μέγιστος ρυθμός λήψης δεδομένων περιορίζεται ακόμα περισσότερο, όπως είναι φυσικό, όταν ο πελάτης αναλαμβάνει και άλλες εργασίες όπως είναι η αποταμίευση των δεδομένων, η αποκωδικοποίησή τους με χρήση λογισμικού και η παρουσίασή τους στις συσκευές εξόδου.

Συμπερασματικά, τα αποτελέσματα των πειραμάτων στους πελάτες αναδεικνύουν την σχέση του υλικού και της τεχνολογίας στον πελάτη (τύπος επεξεργαστή, εξοπλισμός δικτύου, τύπος αποκωδικοποιητή MPEG, μέσα αποθήκευσης) με την ικανότητα του για λήψη δεδομένων, καθώς καταγράφηκαν διαφορετικές επιδόσεις σε ετερογενείς πελάτες. Συνεπώς ισχυροί πελάτες είναι σε θέση να αξιοποιήσουν σε μεγάλο βαθμό τις δυνατότητες του δικτύου, ώστε να λαμβάνουν δεδομένα με υψηλό ρυθμό. Το γεγονός αυτό είναι ιδιαίτερα θετικό, καθώς μπορεί να το εκμεταλλευτεί ο εξυπηρετητής ροής για την αύξηση της απόδοσής του, είτε μέσω μηχανισμών προανάκτησης είτε μέσω καλύτερης διαχείρισης της προσωρινής μνήμης (κεφάλαιο 7).

Στην συνέχεια του κεφαλαίου απομονώθηκε το υποσύστημα ανάκτησης δεδομένων για να μελετήσουμε πειραματικά την συμπεριφορά του. Για τις ανάγκες του πειράματος χρησιμοποιείται ένας υπολογιστής Ultra One, ο οποίος είναι εξοπλισμένος με δύο προσαρμοστές SCSI και τέσσερις μαγνητικούς δίσκους (δύο για κάθε SCSI δίαυλο). Εξετάστηκε η απόδοση για διαφορετικό αριθμό δίσκων και διαφορετικά σχήματα προσαρμογής των δίσκων στους διαθέσιμους SCSI διαύλους. Στο υποσύστημα ανάκτησης επιβάλλουμε πολιτική εξυπηρέτησης σε κύκλους. Τα αποτελέσματα των πειραμάτων έδειξαν απόδοση (throughput) 19 πελατών για 1 δίσκο, 34 πελατών για 2 δίσκους σε κοινό δίαυλο, 38 πελατών για δύο δίσκους σε διαφορετικό δίαυλο και τέλος 68 πελατών σε 4 δίσκους προσαρμοσμένους ανά δύο σε δύο SCSI διαύλους. Από τα αποτελέσματα είναι εμφανές ότι η πολύπλεξη δίσκων σε κοινό δίαυλο SCSI μειώνει σε ένα βαθμό την απόδοση που οι δίσκοι μεμονωμένα είναι σε θέση να επιτύχουν. Μεγαλύτερη ανάλυση στην απόδοση των δίσκων και περισσότερα πειράματα γίνονται στο 7ο κεφάλαιο.

➤ Μηχανισμός δρομολόγησης της ροής δεδομένων



Σε αυτό το κεφάλαιο περιγράφεται ο σχεδιασμός και η ανάπτυξη του μηχανισμού δρομολόγησης που οδηγεί την υπηρεσία διαχείρισης δεδομένων ροής και αποτελεί την ουσιαστική συνεισφορά της παρούσας διατριβής. Παρουσιάζονται τα χαρακτηριστικά της πολυνηματικής τεχνολογίας και οι αρχές ανάπτυξης λογισμικού πραγματικού χρόνου, ενώ εμφανίζεται η ανάγκη υποστήριξης αυτών από έναν εξυπηρετητή αποθήκευσης δεδομένων ροής, προκειμένου να καταστεί εφικτή η δυνατότητα εξυπηρέτησης μεγάλου αριθμού πελατών με εγγυήσεις ποιότητας πραγματικού χρόνου. Ο προτεινόμενος μηχανισμός δρομολόγησης αποτελεί μια ολοκληρωμένη αντιμετώπιση του προβλήματος υποστήριξης χρονικά ευαίσθητων δεδομένων, καθώς αναλαμβάνει τον συντονισμό και την σωστή εκτέλεση διεργασιών που λαμβάνουν χώρα από την έναρξη της ανάκτησης μέχρι την μετάδοσή των δεδομένων στους αιτούμενους πελάτες. Σημαντικό τμήμα του κεφαλαίου αποτελούν αρκετά πειράματα που έγιναν με στόχο την επιβεβαίωση άλλα και την βελτίωση του προτεινόμενου μηχανισμού δρομολόγησης.

6.1 Η ροή δεδομένων πολυμέσων και η υπηρεσία διαχείρισης

Η αποτελεσματική αντιμετώπιση των ψηφιακών δεδομένων πολυμέσων είναι μια αρκετά πολύπλοκη υπόθεση που πηγάζει από την ανάγκη διαχείρισης μεγάλου όγκου δεδομένων και τις απαιτήσεις που προκύπτουν από την συνεχόμενη φύση των πολυμέσων. Ο σχεδιασμός και η ανάπτυξη μιας υπηρεσίας λογισμικού, η οποία θα οργανώνει την αποθήκευση σε μεγάλης κλίμακας συστήματα και θα παρέχει κατ' απαίτηση δεδομένα ροής σε πολλούς πελάτες, προϋποθέτει τον διαχωρισμό των επιμέρους ανεξάρτητων εργασιών. Οι εργασίες αυτές αναφέρονται στις ανάγκες ανάκτησης δεδομένων από τα μέσα αποθήκευσης, αποταμίευσης στην προσωρινή μνήμη και μετάδοσης των δεδομένων μέσω δικτύου.

Ιδιαίτερη δυσκολία εμφανίζεται στην εύρεση ενός μηχανισμού συντονισμού και δρομολόγησης όλων των εργασιών, ώστε να επιτύχουμε την μέγιστη δυνατή απόδοση του συστήματος, αλλά και να ακολουθήσουμε τους περιορισμούς πραγματικού χρόνου με τους οποίους συνοδεύεται η χρήση των μέσων συνεχόμενης ροής.

Ολόκληρο το πέμπτο κεφάλαιο ασχολείται με την υποστήριξη, σχεδιασμό και ανάπτυξη ενός μηχανισμού δρομολόγησης που θα ενσωματωθεί μέσα στην υπηρεσία διαχείρισης δεδομένων ροής.

6.2 Η Πολυνηματική Τεχνολογία

Ο σχεδιασμός της υπηρεσίας ροής προϋποθέτει χρήση της πολυνηματικής τεχνολογίας. Στην επόμενη ενότητα παρουσιάζεται το μοντέλο, οι αρχές του πολυνηματικού προγραμματισμού και τα κύρια πλεονεκτήματα του.

6.2.1 Γενικά

Η ανάγκη για την παροχή της παράλληλης επεξεργασίας από τα λειτουργικά συστήματα αποτελεί μια αρκετά παλαιά ιδέα. Πολλές εφαρμογές οδηγήθηκαν στην χρήση πολλαπλών διεργασιών για την εκτέλεση μιας εργασίας που λογικά αποτελούσε μια διεργασία. Αυτές οι προσεγγίσεις κατέληξαν στην χρήση μη αποδοτικών μηχανισμών επικοινωνίας μεταξύ των διεργασιών (IPC) [Ritc78]. Ο όρος πολυεπεξεργασία (multiprocessing) περιγράφει την ικανότητα του λειτουργικού συστήματος να διαχειρίζεται πολλαπλούς επεξεργαστές για την εκτέλεση των διεργασιών [SunM92]. Αυτές οι συντρέχουσες διεργασίες εμφανίζουν φυσική παράλληλη εκτέλεση. Συντρέχουσες (concurrent) θεωρούνται οι διεργασίες που οι εκτέλεσή τους συμπίπτει χρονικά.

Δύο είναι οι βασικές αρχιτεκτονικές υλικού για την σχεδίαση πολυεπεξεργαστικών μηχανών: Η ασύμμετρη και η συμμετρική αρχιτεκτονική. Σε μια ασύμμετρη αρχιτεκτονική κάθε επεξεργαστής έχει την δικιά του μνήμη, αντίθετα στο συμμετρικό σχήμα όλοι οι επεξεργαστές μοιράζονται μια κοινή μνήμη, γι' αυτό και αλλιώς καλούνται πολυεπεξεργαστές διαμοιραζόμενης μνήμης (shared memory multiprocessors). Οι τεχνολογίες λογισμικού που χρησιμοποιούνταν στα λειτουργικά συστήματα ήταν ελλιπής ως προς την υποστήριξη της πολυεπεξεργασίας. Τα λειτουργικά συστήματα αναγκαζόταν να διατηρούν πληροφορία σε δύο οι περισσότερους επεξεργαστές. Αυτή η επιπρόσθετη εργασία αναφέρεται ως κόστος πολυεπεξεργασίας (multiprocessing overhead). Σε αυτό το κόστος περιλαμβάνονται τεχνικές context-switching και δρομολόγησης. Οι αρχικές προσεγγίσεις χαρακτηρίστηκαν ως πολυεπεξεργαστικά λειτουργικά συστήματα υψηλού επιπέδου (coarse grained multiprocessing OS), καθώς παρείχαν επιπρόσθετες επεξεργαστικές δυνατότητες χωρίς να χρειάζονται αλλαγές στα προγράμματα των εφαρμογών. Επίσης παρείχαν διαφάνεια επεξεργασίας, με την έννοια ότι δεν ήταν γνωστό ποια διεργασία έτρεχε σε ποιόν επεξεργαστή, κάτι που δεν ήταν απαραίτητο άλλωστε. Οι διεργασίες που έτρεχαν πάνω σε αυτά τα λειτουργικά μπορούσαν να εκτελούνται σε φυσικό επίπεδο παράλληλα και ανεξάρτητα ή και ασύγχρονα με κάποια απαιτούμενη μορφή συγχρονισμού (π.χ. μηχανισμοί επικοινωνίας διαμοιραζόμενης μνήμης μεταξύ των διεργασιών). Η εκτέλεση μιας διεργασίας λάμβανε χώρα σε έναν επεξεργαστή. Αν και ο χρόνος εκτέλεσης των ανεξάρτητων διεργασιών δεν ήταν όπως είναι φυσικό ταχύτερος από αυτόν σε ένα μονοεπεξεργαστικό σύστημα, παρόλα αυτά βελτιωνόταν η συνολική παραγωγικότητα.

Όσο το κόστος του υλικού συνέχιζε να μειώνεται, τόσο αυξανόταν η τάση προς τις λεγόμενες διεργασίες μικρού βάρους (lightweight) και την παράλληλη επεξεργασία [MK85][Fren86]. Υπάρχουν αρκετοί στην βιβλιογραφία που προτείνουν την πολυνηματική τεχνολογία ως απαραίτητη υπόθεση για την ανάδειξη της συνεχούς προόδου στις

επεξεργαστικές επιδόσεις, ενώ άλλη την αντιμετωπίζουν ως μια μέθοδο για να εκμεταλλευτούμε πλήρως την επεξεργαστική ισχύ μιας υπολογιστικής μηχανής.

Ο [Bart87] στην προσπάθειά του να παράσχει ένα μοντέλο παράλληλου προγραμματισμού αναδεικνύει δύο σημεία που αποτελούσαν εμπόδιο στην επίτευξη της ικανοποιητικής απόδοσης: την επικοινωνία μεταξύ ανεξάρτητων διεργασιών και το κόστος διαχείρισης των διεργασιών μιας εφαρμογής. Το κόστος πολυεπεξεργασίας που εξαρτάται από τις λειτουργίες -δημιουργία, τερματισμός, αναστολή εκτέλεσης, επανεργοποίηση εκτέλεσης, δρομολόγηση και μηχανισμός context-switch- στις διεργασίες, μπορούν να έχουν σημαντικές επιπτώσεις στην συνολική απόδοση. Αυτά τα μειονεκτήματα έρχεται να προσπεράσει ο προγραμματισμός βάσει νημάτων.

Στο [Gape] αναγνωρίζονται τέσσερις μέθοδοι για την αύξηση της απόδοσης σε υπολογιστικά συστήματα. Η πρώτη αναφέρεται στο επίπεδο του υλικού για ταχύτερα κυκλώματα, η δεύτερη απευθύνεται στον τομέα των μεταγλωττιστών, η τρίτη εξετάζει νέους παράλληλους αλγόριθμους και τέλος η τέταρτη δίνει έμφαση σε νέες προσεγγίσεις όπως μοντέλα ροής δεδομένων. Η πολυνηματική τεχνολογία εμπίπτει στην τρίτη κατηγορία παρέχοντας τον τρόπο για ένα πρόγραμμα να αλληλοκαλύπτει τις επεξεργαστικές εργασίες με αυτές για I/O σε μονοεπεξεργαστικά συστήματα και να αξιοποιεί την ισχύ πολλών επεξεργαστών σε πολυεπεξεργαστικά συστήματα. Στο [Birr89] αναφέρονται επιπρόσθετα κέρδη από την χρήση νημάτων:

1. Σαν επέκταση των μοντέλου διεργασιών, οι εφαρμογές δεν χρειάζονται πλέον να αποτελούνται από πολλαπλές διεργασίες που τρέχουν σε διαφορετικές περιοχές μνήμης (address space).
2. Ευελιξία στην διαχείριση αργών συσκευών I/O, όπως οδηγοί δίσκων και μέσα δικτύου.
3. Δυνατότητα ανάπτυξης εφαρμογών με αυξημένες ανάγκες αλληλεπίδρασης
4. Ευκολία στην ανάπτυξη εξυπηρετητών με πολλούς πελάτες που παρέχει πολλές υπηρεσίες
5. Αύξηση της απόδοσης ακόμη και σε μονοεπεξεργαστικά συστήματα με την μείωση του χρόνου απόκρισης

Το βάρος μιας διεργασίας καθορίζεται από δύο παραμέτρους. Η πρώτη είναι το ποσό πληροφορίας που είναι απαραίτητο για την λειτουργία του μηχανισμού context switch για την διεργασία. Όσο μεγαλύτερο ποσό πληροφορίας χρειάζεται, τόσο μεγαλύτερο είναι ο χρόνος εναλλαγής εκτέλεσης των διεργασιών. Η δεύτερη παράμετρος εξαρτάται από τον βαθμό διαμοιρασμού της μνήμης. τρεις είναι οι γενικά παραδεκτές κατηγορίες που χαρακτηρίζουν το βάρος των διεργασιών. Η πρώτη κατηγορία είναι οι διεργασίες μεγάλου βάρους (Heavyweight). Η δεύτερη κατηγορία είναι οι διεργασίες μεσαίου μεγέθους (Mediumweight) και τελευταία είναι οι διεργασίες μικρού βάρους. Πιο αναλυτικά οι κατηγορίες αυτές περιγράφονται στην συνέχεια.

1. *Διεργασίες μεγάλου βάρους*: Το παραδοσιακό μοντέλο διεργασιών στο UNIX μπορεί να χαρακτηριστεί ως μεγάλου βάρους, κυρίως γιατί δεν επιτρέπει τον διαμοιρασμό των πόρων μεταξύ διαφορετικών διεργασιών (εκτός ίσως του μηχανισμού IPC διαμοιρασμού χώρων μνήμης). Το μοντέλο αυτό χρησιμοποιεί μία σειριακή ροή εκτέλεσης, στην οποία υπάρχει ένα μόνο νήμα ελέγχου. Κάθε διεργασία εκτελείται στον δικό της χώρο διευθύνσεων και αναπαριστάται από μοναδικές δομές δεδομένων στο πυρήνα.
2. *Διεργασίες μεσαίου βάρους*: Σε μια υλοποίηση που ακολουθεί το μοντέλο αυτό υπάρχουν αρκετά νήματα ελέγχου. Κάθε νήμα έχει την δική του στοίβα εκτέλεσης σε επίπεδο χρήστη. Αυτό έχει σαν αποτέλεσμα την καλύτερη δια-νηματική ασφάλεια σε σύγκριση με τις διεργασίες μικρού βάρους, ενώ ταυτόχρονα επιτρέπει την αποδοτική επικοινωνία με τον διαμοιρασμό των υπόλοιπων πόρων. Σε αυτό το μοντέλο όπως και στο επόμενο βασικό πλεονέκτημα είναι ότι οι χρόνοι που μεσολαβούν κατά την δημιουργία, εναλλαγή εκτέλεσης και συγχρονισμό των νημάτων είναι εξαιρετικά μικροί σε σύγκριση με αυτούς που ισχύουν στην πρώτη κατηγορία.
3. *Διεργασίες μικρού βάρους*: Σύμφωνα με αυτό το μοντέλο ο χρόνος που απαιτείται για την δημιουργία μιας διεργασίας ή για να εκτελεστεί ένα context switch είναι αισθητά μικρός. Στο [Gosc 91] η διεργασία μικρού βάρους ορίζεται σαν η διεργασία η οποία περιέχει πολλά νήματα που όλα χρησιμοποιούν τα ίδια δεδομένα και τις ίδιες στοίβες εκτέλεσης με την σημείωση ότι μερικές φορές οι διεργασίες μικρού βάρους καλούνται και ως νήματα. Αυτός ο διαμοιρασμός των πόρων σε υψηλό βαθμό από όλα τα νήματα, μειώνει αισθητά την ασφάλεια. Ένα γεγονός το οποίο είναι εύκολα αντιληπτό αν αναλογιστεί κανείς τα προβλήματα που θα παρουσιαστούν από μια λανθασμένη πρόσβαση στην κοινή στοίβα εκτέλεσης των νημάτων.

Στην επόμενη ενότητα περιγράφονται οι αρχές και τα βασικά χαρακτηριστικά του μοντέλου χρήσης νημάτων.

6.2.2 Το μοντέλο νημάτων

Το μοντέλο νημάτων αποτελεί μια επέκταση του παραδοσιακού μοντέλου διεργασιών στο λειτουργικό σύστημα UNIX. Ένα νήμα είναι απλά μια ροή εκτέλεσης μέσα σε μια διεργασία. Στο παραδοσιακό μοντέλο διεργασιών ένα και μοναδικό νήμα εκτέλεσης αρχίζει με την πρώτη εντολή ακολουθεί την λογική της εφαρμογής και τερματίζεται όταν οι εργασίες της διεργασίας ολοκληρωθούν. Σε αυτό το μοντέλο επίσης περιλαμβάνονται οι έννοιες του χώρου διευθύνσεων (address space), έτσι ώστε κάθε διεργασία να εκτελείται σε διαφορετικούς χώρους διευθύνσεων. Το γεγονός αυτό προσφέρει ένα σχήμα προστασίας και εγγυάται ότι μια διεργασία δεν πρόκειται να αναμιχθεί στην διαδικασία εκτέλεσης μιας άλλης διεργασίας. Στο παραδοσιακό μοντέλο προβλέπονται και οι πόροι μιας διεργασίας, όπως φαίνεται στο Σχήμα 50.

Execution Stream	Resources	Address Space
------------------	-----------	---------------

Σχήμα 50: Παραδοσιακή μορφή διεργασίας

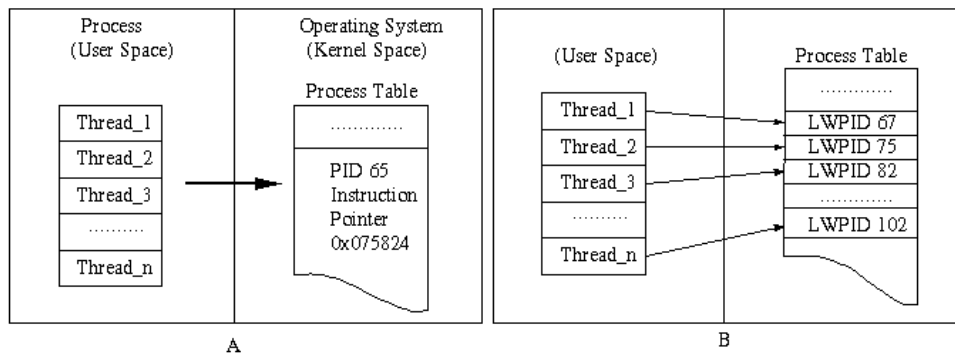
Στο μοντέλο νημάτων μια εφαρμογή έχει πλέον την δυνατότητα να περιλαμβάνει ένα ή περισσότερα μονοπάτια εκτέλεσης, μέσα στην ίδια διεργασία. Κάθε μονοπάτι εκτέλεσης αποτελεί ένα νήμα ελέγχου ή απλά ένα νήμα. Έτσι ένα νήμα διαχωρίζει το σειριακό μονοπάτι εκτέλεσης μιας διεργασίας από τους υπόλοιπους πόρους της. Σε ένα πολυεπεξεργαστικό σύστημα, δύο ή περισσότερα νήματα μέσα στην ίδια διεργασία μπορούν να εκτελούνται ταυτόχρονα σε διαφορετικούς επεξεργαστές. Τα νήματα επεξεργάζονται σαν συντρέχουσες ροές εκτέλεσης, οι οποίες διαμοιράζονται τον ίδιο χώρο διευθύνσεων και αναλαμβάνουν εργασίες που συνδέονται με τις επιθυμητές υπηρεσίες.

Για να μπορεί να υποστηριχθούν πολλά διαφορετικά μονοπάτια εκτέλεσης, κάθε νήμα χρειάζεται ένα τμήμα εκτελέσιμου κώδικα μαζί με μια στοίβα και δεδομένα. Αυτή η στοίβα που βρίσκεται στο επίπεδο χρήστη παρέχει την διαχείριση των τοπικών μεταβλητών του νήματος. Από την άποψη του λειτουργικού συστήματος κάθε ένα από τα νήματα που εκτελούνται πρέπει να έχουν ξεχωριστή στοίβα πυρήνα (kernel stack) και περιβάλλον διεργασίας (process context). Αυτά επιτρέπουν στα νήματα να εκτελούν κλήσεις του συστήματος, και να εξυπηρετούν διακοπές (interrupts) και σφάλματα σελίδας (page faults) ανεξάρτητα. Συμπερασματικά δηλαδή ένα νήμα είναι στην ουσία ένα μονοπάτι εκτέλεσης με την δική του στοίβα, τοπικές μεταβλητές και μετρητή προγράμματος, το οποίο διαμοιράζεται πόρους με άλλα νήματα που εκτελούνται μέσα στον ίδιο χώρο διευθύνσεων. Τα νήματα που δημιουργούνται μέσα σε μια διεργασία καλούνται νήματα παιδιά-αδέρφια (sibling). Ο χαρακτηρισμός αυτός έχει ιδιαίτερη σημασία καθώς απομακρυνόμαστε από την παραδοσιακή σχέση πατέρα-παιδιού των διεργασιών που υποδηλώνει ένα είδος εξάρτησης και περνάμε στην ομότιμη σχέση αδέρφια.

6.2.3 Υλοποιήσεις νημάτων

Τα νήματα μπορούν να υλοποιηθούν ώστε να αποτελέσουν μια αφαίρεση διεργασίας στο επίπεδο του πυρήνα, μια αφαίρεση στο επίπεδο του χρήστη ή σαν συνδυασμό των δύο προηγούμενων. Τα νήματα που υποστηρίζονται από τον πυρήνα (Kernel) έχουν ανάγκη από δομές δεδομένων του πυρήνα, όπως για παράδειγμα μια εγγραφή στον πίνακα των διεργασιών και στον πίνακα χρηστών, ώστε να είναι δυνατή η αναπαράσταση του νήματος. Από την άλλη πλευρά τα νήματα που υλοποιούνται σε επίπεδο χρήστη αναπαριστώνται από δομές δεδομένων μέσα στον χώρο διευθύνσεων της διεργασίας. Τα τελευταία νήματα δεν έχουν ανάγκη από την άμεση υποστήριξη από το λειτουργικό σύστημα. Όμως ένα νήμα σε επίπεδο χρήστη έχει σαν προϋπόθεση για να εκτελεστεί να συνδεθεί με μια διεργασία του πυρήνα. Η υλοποίηση συνεπώς πρέπει να πολυπλέξει ένα νήμα χρήστη πάνω σε μια διεργασία πυρήνα για εκτέλεση και στην συνέχεια πρέπει να υπάρχει η δυνατότητα να διακοπεί η εκτέλεση του τρέχοντος νήματος για να συνεχίσει η διεργασία πυρήνα με την εξυπηρέτηση ενός άλλου «αδερφού» νήματος.

Ένα πλεονέκτημα αυτής της υλοποίησης είναι ότι τα περισσότερα νήματα δεν χρειάζονται αλλαγή φάσης λειτουργίας, δηλαδή από την φάση του χρήστη στην φάση του πυρήνα [ABLL91]. Αυτό επιτρέπει στις λειτουργίες δημιουργίας νημάτων, δρομολόγησης και τερματισμού να ολοκληρωθούν με καλύτερη απόδοση από την ισοδύναμη υλοποίηση υποστηριζόμενη από τον πυρήνα.



Σχήμα 51:Είδη νημάτων

Στα παραπάνω σχήματα φαίνονται δυο διαφορετικές προσεγγίσεις λειτουργίας των νημάτων. Στο σχήμα A επιτρέπεται η πολύπλεξη των νημάτων επιπέδου χρήστη πάνω σε μία διεργασία του πυρήνα. Στο σχήμα B κάθε νήμα επιπέδου χρήστη είναι συνδεδεμένο και υποστηρίζεται από μια και μοναδική διεργασία του πυρήνα.

Ένα δεύτερο πλεονέκτημα είναι ότι τα νήματα επιπέδου χρήστη είναι επεκτάσιμα. Επιπρόσθετη πληροφορία μπορεί να εισαχθεί στο επίπεδο χρήστη χωρίς να είναι αναγκαίο να έχουμε καινούργια έκδοση του λειτουργικού συστήματος. Αντίθετα τα νήματα υποστηριζόμενα από τον πυρήνα έχουν το μειονέκτημα ότι το μέγεθος του πυρήνα αυξάνεται για κάθε νήμα που προστίθεται σε κάθε διεργασία σε επίπεδο χρήστη.

Μερικές υλοποιήσεις χρησιμοποιούν ένα νήμα πυρήνα για κάθε νήμα χρήστη. Το νήμα πυρήνα είναι στην ουσία μια διεργασία μικρού βάρους (lightweight process) και συμβολίζεται LWP. Ένα νήμα επιπέδου χρήστη είναι “δεμένο” (bound) με 1-1 αντιστοίχιση με ένα νήμα πυρήνα. Η δρομολόγηση του συστήματος, η αποστολή εκτέλεσης (dispatching) και η εκτέλεση των διεργασιών μικρού βάρους του πυρήνα καταλήγουν στην εκτέλεση του αντίστοιχου δεμένου νήματος χρήστη. Το μειονέκτημα εδώ είναι ότι κάθε νήμα επιπέδου χρήστη χρειάζεται την δημιουργία μιας LWP πυρήνα και κατά συνέπεια καταναλώνει χρόνο επεξεργασίας στον πυρήνα. Με τον ίδιο τρόπο όταν ένα νήμα επιπέδου χρήστη πρέπει να τερματιστεί, αναγκαστικά τερματίζεται και η αντίστοιχη LWP. Επίσης ένα άλλο μειονέκτημα είναι ότι το νήμα πυρήνα δεν μπορεί να εκτελεί εργασίες εφόσον η εκτέλεση του νήματος επιπέδου χρήστη ανασταλεί για λόγους συγχρονισμού.

Τέλος άλλες και πιο πρόσφατες υλοποιήσεις προσφέρουν και νήματα που μπορεί να πολυπλεχθούν στον πυρήνα αλλά και δεμένα νήματα. Ο αριθμός των LWP στον πυρήνα που είναι διαθέσιμα για την πολύπλεξη των νημάτων καθορίζεται από τη εκάστοτε υλοποίηση αλλά μπορεί να ρυθμιστεί και από την εφαρμογή. Τα νήματα με δυνατότητα πολύπλεξης αποτελούν αφαίρεση σε επίπεδο χρήστη και συνεπώς δεν έχουν άμεση αναπαράσταση στον πυρήνα. Αντίθετα τα δεμένα νήματα τυγχάνουν της άμεσης υποστήριξης του πυρήνα.

6.2.4 Διασυνδέσεις νημάτων

Η προκαθορισμένη συμπεριφορά της ενέργειας δημιουργίας νήματος προϋποθέτει την δημιουργία ενός νήματος με δυνατότητα πολύπλεξης και εκχωρεί μια δομή δεδομένων στον χώρο διευθύνσεων της διεργασίας προκειμένου να αναπαραστήσει το νέο-δημιουργημένο νήμα. Ο μηχανισμός δρομολόγησης των νημάτων πολύπλεξης θα αντιστοιχίσει ένα νήμα σε

κάποια διαθέσιμη LWP πυρήνα για εκτέλεση. Το λειτουργικό σύστημα θα δρομολογήσει την LWP βασισμένο στα χαρακτηριστικά του αντίστοιχου νήματος. Ένα σύνολο από LWP συνήθως διατηρείται ενεργό στον πυρήνα. Με αυτήν την τεχνική μια διεργασία μπορεί να έχει περισσότερα νήματα στο επίπεδο χρήστη από τις LWP που υπάρχουν στον πυρήνα. Η υλοποίηση είναι υπεύθυνη για την πολύπλεξη των νημάτων, την πολιτική δρομολόγησης και την προτεραιότητα εκτέλεσης. Όλες αυτές οι ενέργειες λαμβάνουν χώρα στο επίπεδο χρήστη, εναλλακτικά η μέθοδος δημιουργίας νήματος δεσμεύει μια μοναδική LWP για το νέο νήμα. Σε αυτή την περίπτωση το νήμα είναι δεμένο στην αντίστοιχη LWP. Ένα δεμένο νήμα δεν μπορεί να εκτελεστεί χρησιμοποιώντας κάποια άλλη LWP. Όμοια δεν μπορεί να αντιστοιχηθεί ένα νήμα πολύπλεξης με μία LWP που ήδη έχει δεθεί.

THREADS	LWP	BINDING Rel.
1	1	1 $\rightarrow$ 1
M	1	M $\rightarrow$ 1
1	N	—
M	N	M $\rightarrow$ N

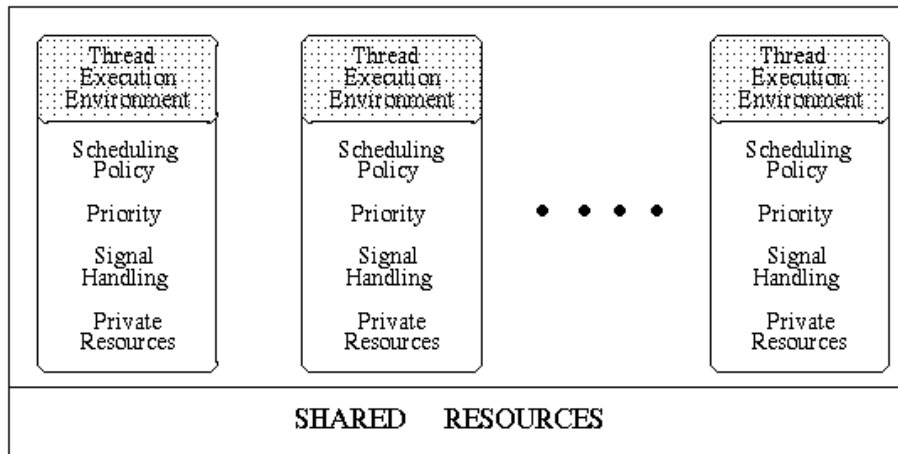
Πίνακας 3: Διασυνδέσεις νημάτων και διεργασιών πυρήνα μικρού βάρους

Στον παραπάνω πίνακα εμφανίζεται η σχέση μεταξύ νημάτων και διεργασιών πυρήνα. Στην τρίτη γραμμή, ο πίνακας δείχνει την κατάσταση στην οποία έχουμε 1 νήμα και M LWP. Αυτή η περίπτωση δεν προσφέρει τίποτα και γενικά δεν υποστηρίζεται. Πιο συγκεκριμένα μπορεί να επιτραπεί να εμφανιστεί αυτή η κατάσταση προσωρινά και μετά σιωπηλά να αφαιρεθούν οι πλεονάζουσες LWP. Κάτι ανάλογο συμβαίνει στην τελευταία γραμμή του πίνακα στην περίπτωση που το M υπερβεί σε κάποια φάση το N.

Ένα παράδειγμα χρήσης και των δύο προσεγγίσεων (νήματα πολύπλεξης και δεμένα νήματα) αποτελεί μια εφαρμογή που ακολουθεί το σχήμα πελάτη-εξυπηρετητή. Η κύρια ρουτίνα του εξυπηρετητή μπορεί να εκτελείται ως ξεχωριστό δεμένο νήμα. Όταν ο πελάτης ζητήσει μια συγκεκριμένη υπηρεσία, η κύρια ρουτίνα μπορεί να δημιουργήσει ένα νέο νήμα για την εξυπηρέτηση του πελάτη. Αυτό το νέο νήμα θα πολυπλεχθεί σε κάποια διαθέσιμη LWP στον πυρήνα.

Στα νήματα μπορούν να εφαρμοστούν ιδιότητες για την περιγραφή των χαρακτηριστικών τους. Οι δύο κύριες ιδιότητες που συνήθως υποστηρίζονται είναι : η ιδιότητα δαίμονας (daemon), σύμφωνα με την οποία διαφοροποιείται ο τρόπος τερματισμού του νήματος και η ιδιότητα απόσπασης (detached) που σχετίζεται με την διαθεσιμότητα της κατάστασης (τιμή εξόδου) κατά τον τερματισμό του νήματος.

Τα νήματα διαμοιράζονται συγκεκριμένους πόρους που σχετίζονται με την διεργασία. Οι πόροι στους οποίους επιτρέπεται η διαμοιραζόμενη πρόσβαση συμπεριλαμβάνονται η μνήμη, οι περιγραφείς αρχείων, τα αναγνωριστικά διεργασίας κ.α. .



Σχήμα 52: Το περιβάλλον εκτέλεσης των νημάτων σε μια διεργασία

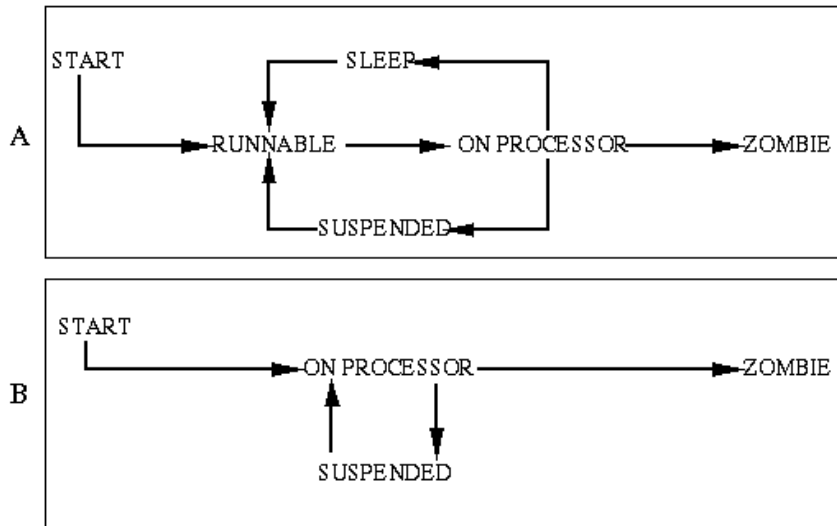
Τα νήματα που δημιουργούνται μέσα σε μια διεργασία διαμοιράζονται τους περισσότερους πόρους (μνήμη, περιγραφείς αρχείων, μεταβλητές περιβάλλοντος). Διατηρούν όμως ξεχωριστή στοίβα εκτέλεσης, μετρητή προγράμματος και ορισμένους πόρους (αναγνωριστικό νήματος, μετρητές χρόνου, καταχωρητές) που χαρακτηρίζονται αποκλειστικοί (private). Ακολουθούν μια πολιτική δρομολόγησης με την δική τους τιμή προτεραιότητας και διαχειρίζονται ανεξάρτητα τα σήματα.

Τα νήματα εκτελούνται μέσα στον ίδιο χώρο διευθύνσεων και δεν τυγχάνουν περιορισμούς ασφαλείας. Όταν για παράδειγμα ένα νήμα αλλάζει την τιμή μιας μεταβλητής στην κοινή μνήμη, η αλλαγή αυτή είναι άμεσα ορατή από όλα τα νήματα. Φυσικά υπάρχουν και πόροι που δεν μπορούν να διαμοιραστούν και έχουν σχέση με την λειτουργία των νημάτων.

6.2.5 Εκτέλεση νημάτων

Μια τιμή που υποδηλώνει μια κατάσταση είναι διασυνδεδεμένη με κάθε νήμα. Το διάγραμμα μεταβάσεων κατάστασης κατά την εκτέλεση των νημάτων φαίνεται στο Σχήμα 53.

Η μέθοδος δημιουργίας νήματος η οποία πρόκειται να δημιουργήσει ένα νήμα πολύπλεξης, αρχικοποιεί το νήμα στην κατάσταση RUNNABLE (έτοιμο για εκτέλεση). Το νέο νήμα μεταπηδά στην κατάσταση ON PROCESSOR (επεξεργασίας) μόλις αντιστοιχηθεί με μια LWP πυρήνα. Τότε ο πυρήνας αποστέλλει για εκτέλεση την LWP. Κατά την διάρκεια επεξεργασίας, η εκτέλεση του νήματος μπορεί να ανασταλεί για λόγους συγχρονισμού και μεταβαίνει στην κατάσταση SLEEP (ανενεργή) με ταυτόχρονη απελευθέρωση της αντιστοιχίας του με την LWP. Όταν η συνέχιση της εκτέλεσης καταστεί δυνατή, το νήμα αποκτά πάλι την RUNNABLE κατάσταση πάνω σε μία διαθέσιμη LWP. Φυσικά όπως φαίνεται και στο σχήμα η εκτέλεση του νήματος μπορεί να ανασταλεί κατ' απαίτηση της εφαρμογής, οπότε και περνά στην κατάσταση SUSPEND (αναστολής). Η υλοποίηση του μηχανισμού υποστήριξης νημάτων παρέχει δρομολόγηση βασισμένη σε σήματα και δύναται να διακόψει την εκτέλεση ενός νήματος, αν ένα άλλο νήμα που πολυπλέκεται στην ίδια LWP έχει μεγαλύτερη προτεραιότητα και βρίσκεται στην RUNNABLE κατάσταση (preemption).



Σχήμα 53: Διάγραμμα καταστάσεων κατά την επεξεργασία νημάτων
 Στο σχήμα Α παρουσιάζεται το διάγραμμα καταστάσεων για τα νήματα με δυνατότητα πολύπλεξης, ενώ στο σχήμα Β έχουμε το αντίστοιχο διάγραμμα για δεμένα νήματα το οποίο είναι αρκετά διαφορετικό.

Όπως προκύπτει από τα προηγούμενα κάθε νήμα έχει την δική του τιμή προτεραιότητας που σχετίζεται με μία πολιτική δρομολόγησης. Αν και τα νήματα πολύπλεξης περιορίζονται αναγκαστικά στην πολιτική διαμοιρασμού χρόνου (time-sharing), δεν ισχύει ανάλογος περιορισμός για τα δεμένα νήματα στα οποία μπορεί να εφαρμοστεί οποιαδήποτε πολιτική δρομολόγησης διαθέσιμη από το λειτουργικό σχήμα.

6.3 Επισκόπηση αρχιτεκτονικής της υπηρεσίας ροής

Στην ενότητα 3.5 είχαν επισημανθεί τα πιο σημαντικά προβλήματα στις τεχνικές ανάπτυξης λογισμικού για την αντιμετώπιση των δεδομένων συνεχόμενης ροής. Η ύπαρξη αρκετών διεργασιών, το κόστος συγχρονισμού και επικοινωνίας αυτών των διεργασιών, και το μονοπάτι της ροής δεδομένων μέσω των υποσυστημάτων, είναι επιγραμματικά τα ζητήματα που χρειάζονται αντιμετώπιση και διαφορετική προσέγγιση. Έχοντας υπόψη τα θετικά στοιχεία και τα βασικά πλεονεκτήματα που μας προσφέρει η πολυνηματική τεχνολογία, ακολουθεί ο σχεδιασμός της νέας υπηρεσίας αποθήκευσης δεδομένων συνεχόμενης ροής.

Η υπηρεσία αυτή αποτελεί στην ουσία ένα εξυπηρετητή που έχει να επεξεργαστεί αιτήσεις πολλών πελατών για αναπαραγωγή πολυμέσων. Οι βασικές εργασίες που πρέπει να αναληφθούν από την υπηρεσία ροής και είναι λογικά διαχωρίσιμες είναι οι ακόλουθες. Η εργασία διαχείρισης των αιτήσεων θα αποτελέσει την πύλη εισόδου σε αυτό το υποσύστημα λογισμικού. Σε αυτή την κατηγορία συμπεριλαμβάνεται η ανάπτυξη λογισμικού για την υποδοχή μηνυμάτων, την αποκωδικοποίησή τους και την αποθήκευσή τους στις κατάλληλες δομές. Κάθε νέα υπηρεσία που ζητά ένας πελάτης αποτελεί μια ξεχωριστή εργασία και έτσι πρέπει να αντιμετωπίζεται από το σύστημα και να αναπαριστάται με τους κατάλληλους μηχανισμούς λογισμικού. Μια άλλη εμφανή ανεξάρτητη κατηγορία εργασιών

συνδέεται με τα μέσα αποθήκευσης. Η εργασία που επιτελείται για παράδειγμα από έναν οδηγό δίσκου είναι εύκολα κατανοητό ότι αποτελεί μια αυτόνομη ροή εκτέλεσης, ή καλύτερα από μια διαφορετική οπτική γωνία το τμήμα λογισμικού που είναι υπεύθυνο για την διαχείριση ενός μέσου αποθήκευσης αποκτά την έννοια ενός μικρο-εξυπηρετητή. Τέλος πρέπει να αναφερθεί η σημαντική εργασία που πρέπει να λάβει χώρα για την ομαλή και άμεση μετάδοση των δεδομένων στο εξωτερικό περιβάλλον του εξυπηρετητή αποθήκευσης.

Η ύπαρξη μιας κεντρικής οντότητας λογισμικού η οποία θα αναλαμβάνει να οδηγεί και να συντονίζει τις επιμέρους εργασίες όπως αυτές αναφέρθηκαν παραπάνω είναι αναγκαία για λόγους που περιγράφονται στην συνέχεια της ενότητας.

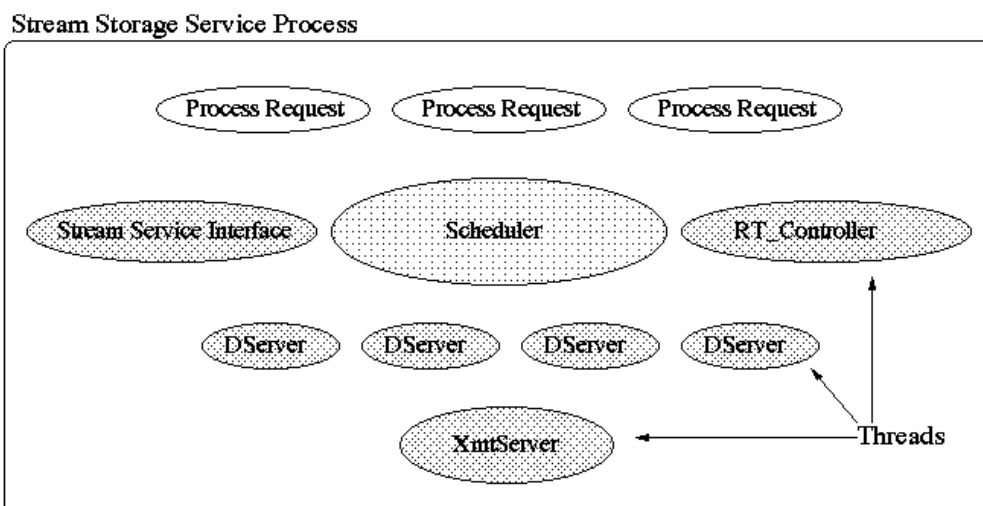
Μια κεντρική οντότητα δρομολογητή είναι απαραίτητη για:

1. Να δρομολογεί κρίσιμους κοινούς πόρους, όπως είναι οι αποταμιευτές και οι επικοινωνία με το εξωτερικό περιβάλλον. Για παράδειγμα ο χώρος των αποταμιευτών μπορεί να μην είναι αρκετά μεγάλος για όλους τους δίσκους σε ένα σύστημα παράλληλο σε υψηλό βαθμό, συνεπώς χρειάζεται ένας κοινός διαχειριστής του χώρου αυτού, όπως επίσης ένας κοινός διαχειριστής απαιτείται για την δρομολόγηση της επικοινωνίας μέσω των γραμμών μεταφοράς δεδομένων (transfer links).
2. Να δρομολογεί αιτήσεις ανάκτησης δεδομένων από πολλαπλούς δίσκους όπως αιτήσεις ανάκτησης αντικειμένων (π.χ. video), τα οποία έχουν διασπαστεί σε διαφορετικούς δίσκους
3. Να συντονίζει και συγχρονίζει τους διαφορετικούς δίσκους και να χρησιμοποιεί βέλτιστα των χώρο των προσωρινών αποταμιευτών, ένα γεγονός που έχει ιδιαίτερη σημασία καθώς η νέα υπηρεσία διαχειρίζεται δεδομένα συνεχόμενης ροής.
4. Να καθοδηγεί την εκτέλεση των επιμέρους εργασιών ώστε να ελέγχεται χρονικά η πορεία επεξεργασίας, έτσι ώστε να ακολουθούνται οι περιορισμοί πραγματικού χρόνου. Η τελευταία αυτή υπευθυνότητα αποτελεί την σημαντικότερη πρόκληση στην ανάπτυξη της υπηρεσίας και ταυτόχρονα το πιο κρίσιμο σημείο επιτυχίας.

Αν αντιστοιχίσουμε κάθε μία από τις λογικά διαχωρισμένες εργασίες σε μια ροή εκτέλεσης δηλαδή σε ένα νήμα ελέγχου, καταλήγουμε στο πρώτο βήμα του πολυνηματικού σχεδιασμού το οποίο εμφανίζεται γραφικά στο Σχήμα 54 και περιγράφεται με μεγαλύτερη λεπτομέρεια στην επόμενη υποενότητα. Το επόμενο στάδιο στον σχεδιασμό είναι ο προσεκτικός συντονισμός και ο κατάλληλος συγχρονισμός της εκτέλεσης των επιμέρους νημάτων, έχοντας πάντα υπόψη την επίτευξη ορισμένων αρχών, καθοριστικών για την απόδοση της υπηρεσίας. Για παράδειγμα η ανταπόκριση του κάθε νήματος στις δημιουργούμενες ανάγκες επεξεργασίας πρέπει να είναι η κατάλληλη, ανάλογα με την αξιολόγηση της κρισιμότητας εκτέλεσης κάθε εργασίας. Επίσης με τις κατάλληλες τεχνικές συγχρονισμού των νημάτων θα πρέπει να εξασφαλίζεται ότι η δέσμευση πόρων και κυρίως του επεξεργαστή γίνεται μόνο όταν είναι απαραίτητο, αποφεύγοντας με αυτόν τον τρόπο φαινόμενα μονοπώλησης πόρων σε μια προσπάθεια προσέγγισης του ζητήματος βελτιστοποίησης χρήσης των διαθέσιμων πόρων του συστήματος.

Το σημαντικότερο πλεονέκτημα που προσφέρει η πολυνηματική τεχνολογία, το οποίο αναφέρεται στην παράλληλη επεξεργασία με δυνατότητα διαμοιρασμού των πόρων του συστήματος, προσπαθούμε να εκμεταλλευτούμε σε αυτήν την περίπτωση για να εκμηδενίσουμε το κόστος επικοινωνίας μεταξύ διεργασιών, να ελαττώσουμε την χρονική επιβάρυνση των μηχανισμών context switch και τέλος να εγκαταστήσουμε ένα αποδοτικό μονοπάτι ροής δεδομένων, παρέχοντας άμεση διασύνδεση στις εργασίες ανάκτησης δεδομένων από τα μέσα αποθήκευσης και μετάδοσής αυτών μέσω δικτύου στους αντίστοιχους πελάτες.

Ο σχεδιασμός της υπηρεσίας αποθήκευσης είχε σαν στόχο την δημιουργία ενός συστήματος λογισμικού ικανό να αξιοποιήσει πλήρως τους πόρους ενός υπολογιστικού συστήματος για την καλύτερη εξυπηρέτηση χρηστών που ζητούν πρόσβαση σε δεδομένα πολυμέσων. Ιδιαίτερη προσοχή έχει δοθεί κατά την ανάπτυξη του συστήματος, τόσο στις τεχνικές υλοποίησης όσο και στην ποιότητα του λογισμικού. Έγινε πλήρη χρήση των πλεονεκτημάτων που προσφέρει η πολυνηματική τεχνολογία στα σύγχρονα λειτουργικά συστήματα, ενώ στόχος αποτελούσε επίσης η εκμετάλλευση των υπηρεσιών πραγματικού χρόνου που προσφέρονται από το λειτουργικό σύστημα για να αντιμετωπίσουμε τις χρονικές απαιτήσεις των πολυμέσων.



Σχήμα 54: Η πολυνηματική υπηρεσία αποθήκευσης δεδομένων ροής
Κάθε νήμα αναλαμβάνει ένα σαφώς καθορισμένο τομέα εργασιών. Οι εξυπηρετητές δίσκων (Dservers) διαχειρίζονται την ανάκτηση δεδομένων. Τα νήματα επεξεργασίας των αιτήσεων αναλύουν τις απαιτήσεις και συγκεντρώνουν απαραίτητη πληροφορία. Το σύστημα ζεύξης (Stream Service Interface) παρακολουθεί τα μηνύματα από το εξωτερικό περιβάλλον. Τα νήματα του δρομολογητή και ελεγκτή πραγματικού χρόνου οδηγούν την εκτέλεση, ενώ τέλος ο εξυπηρετητής μετάδοσης είναι υπεύθυνος για την αποστολή των δεδομένων

Στο σχήμα η έλλειψη αντιπροσωπεύει ένα νήμα (thread) που εκτελεί εξειδικευμένη εργασία. Ακολουθεί περιληπτική περιγραφή των επιμέρους τμημάτων του συστήματος:

- **Σύστημα ζεύξης και επαφής - Stream Service Interface (SSI):** Αποτελεί τον κόμβο εισόδου της υπηρεσίας. Το νήμα αυτό βρίσκεται μόνιμα σε κατάσταση αναμονής για να δεχθεί τις αιτήσεις υποσυστημάτων ανώτερου επιπέδου που ζητούν ανάκτηση

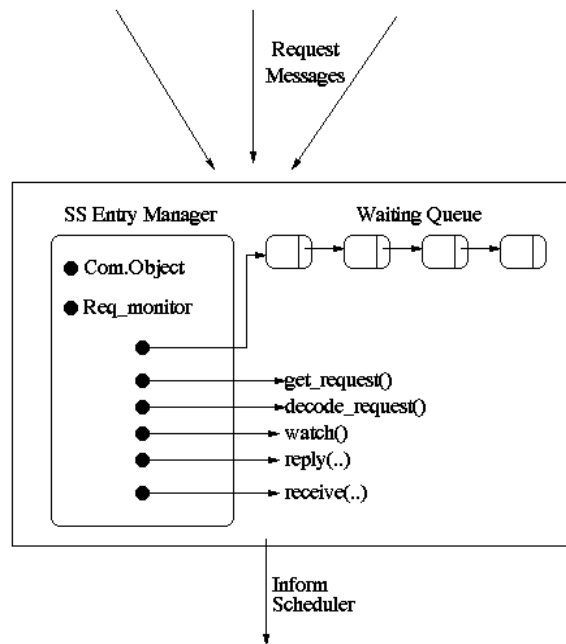
δεδομένων ροής. Επιπρόσθετα αποκωδικοποιεί τα εισερχόμενα μηνύματα και αναλαμβάνει την απαραίτητη ανταλλαγή πληροφορίας με τον αιτούμενο πελάτη.

- **Κεντρικός δρομολογητής - Scheduler:** Αυτό το νήμα έχει την ευθύνη της κεντρικής επεξεργασίας για την δρομολόγηση των αιτήσεων. Συγκεντρώνει την απαραίτητη πληροφορία, ελέγχει την ικανότητα του συστήματος να εξυπηρετήσει νέες αιτήσεις, ενώ είναι ενήμερο κάθε στιγμή για την εξέλιξη της εξυπηρέτησης των αιτήσεων. Ο δρομολογητής συνεργάζεται με άλλα κομμάτια λογισμικού που χρειάζονται για την υποστήριξη της δρομολόγησης και την συγκέντρωση της πληροφορίας. Επίσης για να εξυπηρετήσει μια αίτηση, προκαλεί την δημιουργία ενός νέου νήματος εργασίας που αναλαμβάνει την επεξεργασία της αίτησης (Process Request Thread). Το νέο αυτό νήμα εργάζεται ανεξάρτητα και στο τέλος της επεξεργασίας επιστρέφει τα αποτελέσματα στον κεντρικό δρομολογητή.
- **Ελεγκτής πραγματικού χρόνου - RT-Controller:** Ο ελεγκτής πραγματικού χρόνου αποτελεί ξεχωριστό νήμα και μοναδική του υπευθυνότητα είναι να παίρνει τακτικά χρονικά σημεία (time stamps) από το σύστημα, να τα παρακολουθεί και να οδηγεί την δρομολόγηση. Ταυτόχρονα συγχρονίζει την λειτουργία των μέσων αποθήκευσης καθορίζοντας την άμεση έναρξη της ανάκτησης των δεδομένων και οδηγεί την λειτουργία του εξυπηρετητή μετάδοσης (XmtServer) παρέχοντας του χρονική πληροφορία.
- **Εξυπηρετητές δίσκου – Disk Servers:** Τα νήματα αυτά είναι καθένα υπεύθυνο για την διαχείριση ενός μέσου αποθήκευσης. Όπως αναφέρθηκε ο RT-Controller ειδοποιεί τους εξυπηρετητές δίσκου για την έναρξη του κύκλου εξυπηρέτησης. Τα ανακτώμενα δεδομένα μεταφέρονται στην ειδική μνήμη μετάδοσης για να τα αναλάβει στην συνέχεια ο εξυπηρετητής μετάδοσης.
- **Εξυπηρετητής μετάδοσης - XmtServer:** Εδώ ο εξυπηρετητής-νήμα με τον κατάλληλο μηχανισμό δρομολόγησης αναλαμβάνει την αποστολή των δεδομένων ροής μέσω του δικτύου στους χρήστες που αιτήθηκαν την ανάκτηση. Σε συνεργασία με τον RT-Controller για την παροχή χρονικής πληροφορίας, δρομολογεί την μετάδοση των κατάλληλων πακέτων.

Ο πολυνηματικός σχεδιασμός και η ανάπτυξη του συστήματος επιτρέπει την δρομολόγηση των επιμέρους νημάτων με βάση προτεραιότητες. Στην επόμενη ενότητα περιγράφονται με μεγαλύτερη λεπτομέρεια οι υπευθυνότητες του κάθε νήματος, τα τμήματα λογισμικού υποστήριξης και οι δομές δεδομένων, καθώς και η αλληλεπίδραση και ο συγχρονισμός των νημάτων που αποτελούν την υπηρεσία διαχείρισης δεδομένων συνεχόμενης ροής.

6.3.1 Η υπηρεσία διαχείρισης δεδομένων συνεχόμενης ροής

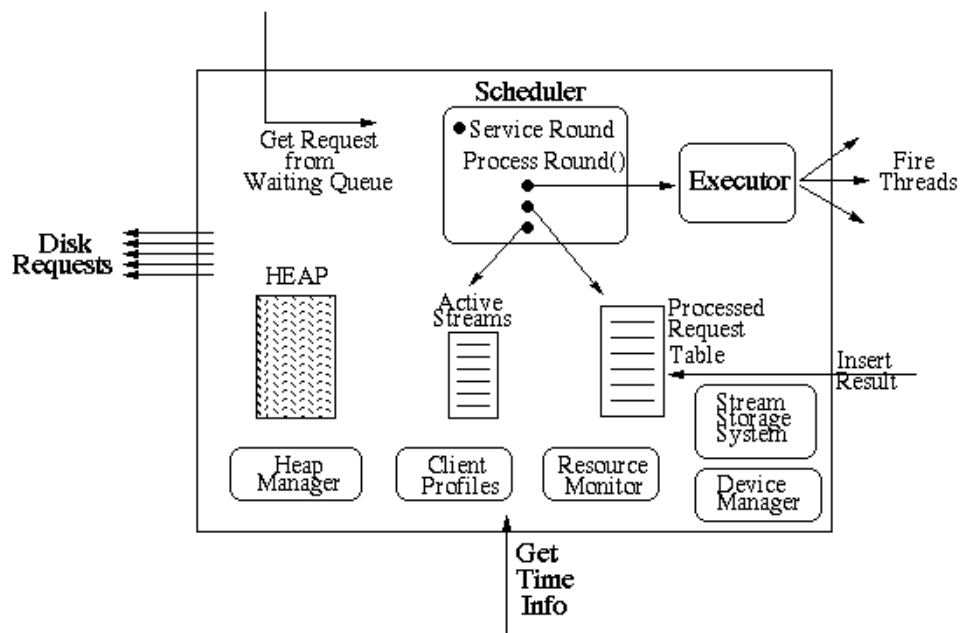
Η υπηρεσία διαχείρισης δεδομένων συνεχόμενης ροής καλείται να επιτελέσει ένα πολύπλοκο έργο με πολλές εργασίες οι οποίες πρέπει να λαμβάνουν χώρα ταυτόχρονα και επιπρόσθετα πρέπει να ακολουθούν χρονικούς περιορισμούς. Κάθε νήμα της υπηρεσίας αναλαμβάνει ένα σαφή τομέα εργασιών.



Σχήμα 55: Το νήμα ζεύξης και επαφής της υπηρεσίας ροής

Το Νήμα Ζεύξης και επαφής της υπηρεσίας ροής: Το νήμα αυτό αναλαμβάνει την επικοινωνία με το εξωτερικό περιβάλλον. Κύρια αρμοδιότητά του είναι να λαμβάνει τα μηνύματα αιτήσεων για ανάκτηση αντικειμένων ροής. Τα μηνύματα-αιτήσεις μπορεί να προέρχονται κατευθείαν από τους απομακρυσμένους πελάτες ή από τα υπόλοιπα υποσυστήματα που τρέχουν στην ίδια υπολογιστική μηχανή. Στην πρώτη περίπτωση χρησιμοποιείται το πρωτόκολλο TCP/IP και internet sockets για την επικοινωνία, ενώ στην δεύτερη για λόγους απόδοσης γίνεται χρήση των UNIX domain sockets. Όπως φαίνεται στο σχήμα ένας κεντρικός διαχειριστής εισόδου (SSEntry Manager) οδηγεί την εκτέλεση του νήματος. Αυτός ο διαχειριστής ελέγχει τον τρόπο επικοινωνίας, αντιστοιχεί σε κάθε εισερχόμενη αίτηση ένα μοναδικό αναγνωριστικό και παρέχει βασικές μεθόδους λειτουργίας. Το αναγνωριστικό επιτρέπει την παρακολούθηση όλων των ενεργειών των πελατών, ενώ με βάση το αναγνωριστικό αυτό επιτρέπονται λειτουργίες πάνω σε προηγούμενες αιτήσεις. Για παράδειγμα οι λειτουργίες τερματισμού/παύσης των αιτήσεων ανάκτησης αντικειμένων ροής, χρησιμοποιούν το αναγνωριστικό της αίτησης για να διευκρινίσουν στο σύστημα ποια αίτηση αναπαραγωγής πρέπει να τερματιστεί/διακοπεί. Ο διαχειριστής εισόδου διατηρεί επίσης μια ουρά αναμονής στην οποία αποθηκεύονται οι αποκωδικοποιημένες αιτήσεις, ώστε να είναι προσβάσιμες από τον δρομολογητή (μέθοδος get_request) όπως θα περιγραφεί στην συνέχεια του κειμένου.

Ο Δρομολογητής της υπηρεσίας ροής: Ο δρομολογητής αποτελεί την καρδιά της υπηρεσίας ροής δεδομένων. Το τμήμα αυτό συγκεντρώνει και οργανώνει την πληροφορία από όλα τα άλλα τμήματα λογισμικού. Στο Σχήμα 56 εμφανίζεται η γενική αρχιτεκτονική του δρομολογητή με τα επιμέρους τμήματα λογισμικού που υποστηρίζουν την λειτουργία του.



Σχήμα 56: Το νήμα δρομολογητής της υπηρεσίας ροής

Το νήμα-δρομολογητής αποτελεί τον κεντρικό οργανωτή της υπηρεσίας ροής. Δέχεται πληροφορία από άλλα νήματα και την επεξεργάζεται. Η λειτουργία του δρομολογητή υποστηρίζεται από ανεξάρτητα τμήματα λογισμικού για την διαχείριση της μνήμης, των μέσων αποθήκευσης και των αποθηκευμένων αντικειμένων. Για την εξυπηρέτηση των αιτήσεων δημιουργεί νέα νήματα και διατηρεί πληροφορία για τις αιτήσεις που ήδη εξυπηρετούνται αλλά και για αυτές που έχουν επεξεργαστεί. Η εξυπηρέτηση του δρομολογητή είναι βασισμένη σε κύκλους, στην διάρκεια των οποίων στέλνονται οι απαραίτητες υπο-αιτήσεις δίσκου.

Σημαντικό ρόλο έχει ο διαχειριστής μνήμης (Heap manager). Ο διαχειριστής μνήμης δεσμεύει αρχικά ένα τμήμα της κύριας μνήμης, το οποίο προορίζεται για τις ανάγκες επεξεργασίας της υπηρεσίας ροής. Ο διαχειριστής έχει την πλήρη ευθύνη της διαχείρισης του χώρου της δεσμευμένης μνήμης, παρέχοντας δύο βασικές λειτουργίες για εκχώρηση και απελευθέρωση (allocation/deallocation) τμημάτων μνήμης. Η μνήμη αυτή (heap) δεν έχει καμία σχέση με τις ανάγκες προσωρινής αποταμίευσης των δεδομένων ροής. Προσπαθεί να καλύψει αποδοτικά τις ανάγκες προσωρινής μνήμης για την υποστήριξη δυναμικών δομών δεδομένων κατά την εκτέλεση του λογισμικού της υπηρεσίας ροής. Η ανάπτυξη ενός τέτοιου μηχανισμού διαχείρισης μνήμης έχει τα εξής πλεονεκτήματα: Πρώτον παρέχεται αυστηρός έλεγχος στις λειτουργίες εκχώρησης μνήμης, ώστε να αποφεύγονται μακροπρόθεσμα φαινόμενα σταδιακής αύξησης της μνήμης που χρησιμοποιεί η διεργασία και δεύτερον και πιο ουσιαστικό, με τον διαχειριστή μνήμης όλες οι λειτουργίες εκχώρησης/απελευθέρωσης αντιμετωπίζονται άμεσα και γρήγορα μέσα στον χώρο διευθύνσεων της διεργασίας χωρίς την μεσολάβηση του λειτουργικού συστήματος που θα οδηγούσε σε μια πιο χρονοβόρα προσέγγιση.

Το σύστημα αποθήκευσης αντικειμένων ροής (Stream Storage System), είναι ένα άλλο σημαντικό τμήμα λογισμικού, το οποίο αναλαμβάνει την οργάνωση της αποθήκευσης των αντικειμένων ροής στο σύστημα. Το σύστημα αυτό, σε συνεργασία με τον διαχειριστή μέσων αποθήκευσης (Device Manager) καθορίζει τον τρόπο αποθήκευσης, όπως επίσης διατηρεί και διαχειρίζεται πληροφορία σχετική με την τοποθέτηση των αντικειμένων ροής στα πολλαπλά μέσα αποθήκευσης. Τόσο ο δρομολογητής όσο και τα νήματα που

δημιουργούνται για την εξυπηρέτηση νέων δεδομένων εξάγουν την απαραίτητη πληροφορία από το σύστημα αποθήκευσης, για την περιοδική διαδικασία ανάκτησης δεδομένων (Βλ. κεφ. 4).

Ο δρομολογητής διατηρεί δύο βασικές δομές δεδομένων, τον πίνακα των αιτήσεων που έχουν επεξεργαστεί (Processed Request Table) και τον πίνακα των ενεργών ροών (Active Streams Table). Τα νήματα επεξεργασίας των αιτήσεων εισάγουν εγγραφές στον πρώτο πίνακα για να ενημερώσουν τον δρομολογητή. Ο πίνακας ενεργών ροών περιέχει το σχήμα εξυπηρέτησης των αιτήσεων. Με τα δεδομένα που υπάρχουν στον πίνακα αυτό, παρακολουθείται η κατάσταση εξυπηρέτησης κάθε αίτησης και παράγονται οι υποαιτήσεις προσπέλασης δίσκου σε κάθε κύκλο.

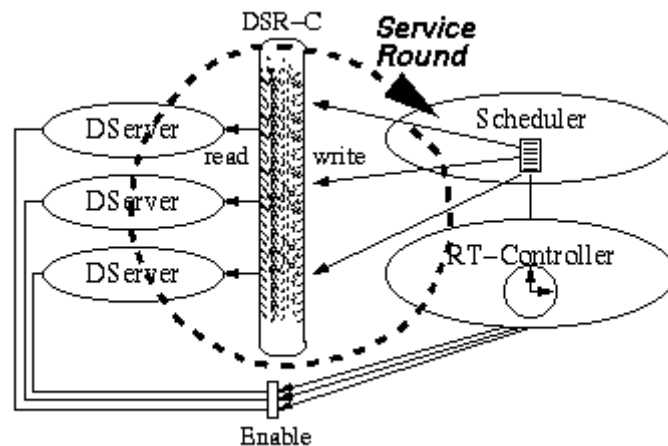
Πολλές δομές (όπως ο πίνακας επεξεργασμένων αιτήσεων) προσπελαύνονται από περισσότερα από ένα νήματα. Το γεγονός αυτό μπορεί να εμφανίσει φαινόμενα ασυνέπειας στην πληροφορία που αναπαριστούν οι δομές. Για τον λόγο αυτό χρησιμοποιούνται κλειδώματα αποκλειστικής εκτέλεσης (mutex locks) ώστε να μην αλλοιωθεί η πληροφορία, που διατηρείται σε διάφορες δομές, από την ταυτόχρονη προσπέλαση διαφορετικών νημάτων.

Επίσης ο σχεδιασμός προβλέπει την ύπαρξη του ελεγκτή των πόρων του συστήματος (Resource Monitor) και του διαχειριστή των πελατών (Client Profile Manager). Υπευθυνότητα του ελεγκτή είναι να δέχεται τα αποτελέσματα της επεξεργασίας μιας νέας αίτησης (που προκύπτουν από τα νήματα επεξεργασίας με την συνεργασία του διαχειριστή περιεχομένου) και αξιοποιώντας την πληροφορία που διατηρεί σχετικά με τον φόρτο των συσκευών και την διαθεσιμότητα άλλων πόρων, να ενημερώνει τον δρομολογητή με την κατάσταση του συστήματος, ώστε ο τελευταίος να αποφασίσει αν η νέα αίτηση μπορεί να εξυπηρετηθεί με εγγυήσεις ποιότητας υπηρεσιών, χωρίς αρνητικές επιδράσεις στην εξυπηρέτηση των υπόλοιπων αιτήσεων. Περισσότερες λεπτομέρειες για τον ρόλο του ελεγκτή των πόρων, καθώς και του διαχειριστή πελατών (Client Profile Manager) υπάρχουν στο επόμενο κεφάλαιο.

Η μηχανισμός δρομολόγησης που αναπτύχθηκε ακολουθεί εξυπηρέτηση των αιτήσεων σε κύκλους, ακολουθώντας με αυτόν τον τρόπο την περιοδική φύση των δεδομένων συνεχόμενης ροής. Σύμφωνα με την πολιτική αυτή σε κάθε κύκλο εξυπηρέτησης τμήματα από κάθε αντικείμενο ροής ανακτώνται από τα μέσα αποθήκευσης και αποστέλλονται μέσω δικτύου στους πελάτες. Η χρονική διατήρηση των κύκλων γίνεται με την βοήθεια του ελεγκτή πραγματικού χρόνου, ο οποίος ενημερώνει τον δρομολογητή για την έναρξη κάθε κύκλου εξυπηρέτησης. Με την βοήθεια του πίνακα ενεργών ροών, ο δρομολογητής γνωρίζει τι εργασίες πρέπει να λάβουν χώρα για τον τρέχοντα κύκλο. Σε αυτή την φάση επεξεργάζονται οι ενεργές αιτήσεις και παράγονται οι υποαιτήσεις προσπέλασης δίσκου. Οι αιτήσεις αυτές γράφονται σε ένα ειδικό χώρο μέσα στον χώρο διευθύνσεων της διεργασίας, ο οποίος καλείται κανάλι επικοινωνίας εξυπηρετητών δίσκου (DSR_Channel) και περιγράφεται σε επόμενη υποενότητα. Με τον τρόπο αυτό επιτυγχάνεται η άμεση μεταφορά πληροφορίας στα νήματα διαχείρισης των μέσων αποθήκευσης.

Ο κύκλος του συστήματος είναι σταθερός και κοινός για όλους τους εξυπηρετητές δίσκων. Για λόγους απόδοσης αποθηκεύουμε τα αντικείμενα ροής σε συνεχόμενα τμήματα με το κατάλληλο μέγεθος που υπαγορεύεται από τον ρυθμό κατανάλωσης ($Cr \cdot SYS_ROUND$), ώστε να έχουμε μια ενέργεια αναζήτησης στον δίσκο για κάθε αντικείμενο ροής σε κάθε κύκλο εξυπηρέτησης. Αν όμως ο διαχειριστής τοποθέτησης δεδομένων κρίνει ότι το μέγεθος αυτό είναι μικρό, τότε μπορεί να επιλέξει την αποθήκευση πολλαπλάσιου μεγέθους ($x \cdot Cr \cdot SYS_ROUND$), ώστε το συγκεκριμένο αντικείμενο ροής να εξυπηρετείται κάθε x κύκλους του συστήματος.

Το παραπάνω μοντέλο λειτουργίας αποτελεί μια γενική αντιμετώπιση, επιτρέποντας με αυτό τον τρόπο την εφαρμογή σχημάτων coarse αλλά και fine striping, καθώς ο δρομολογητής καθορίζει τον αριθμό των υποαιτήσεων και πότε πρέπει να εκτελεστούν για όλους τους δίσκους. Επίσης εφικτή είναι η υιοθέτηση διαφόρων σχημάτων προτεραιοτήτων, αναδιατάσσοντας με τον κατάλληλο αλγόριθμο τις υποαιτήσεις δίσκου στο κανάλι επικοινωνίας. Επίσης αξίζει να σημειωθεί ότι τεχνικές προανάκτησης δεδομένων είναι άμεσα εφαρμόσιμες ανάλογα με τις αποφάσεις του κεντρικού δρομολογητή.



Σχήμα 57: Η κυκλική δρομολόγηση των εργασιών στην υπηρεσία ροής

Στο σχήμα εικονίζεται με γραφικό τρόπο ο μηχανισμός δρομολόγησης των επιμέρους εργασιών στην υπηρεσία δεδομένων συνεχόμενης ροής. Η δρομολόγηση βασίζεται στην περιοδική φύση των δεδομένων ροής και εξυπηρετεί ταυτόχρονα τους πελάτες του συστήματος, οδηγώντας την εκτέλεση με κύκλους. Σε κάθε κύκλο ένας αριθμός από τμήματα δεδομένων πρέπει να ανακτηθεί για κάθε πελάτη. Ο ελεγκτής πραγματικού χρόνου αναλαμβάνει τον χρονικό συντονισμό των εξυπηρετητών δίσκου, παρέχοντας τους πληροφορία μέσω του κεντρικού δρομολογητή και καθορίζοντας την έναρξη των εργασιών κάθε κύκλου.

Πρόκειται για ένα σχήμα που σαφώς εμφανίζει πλεονεκτήματα σε σύγκριση με την προηγούμενη προσέγγιση που είχε χρησιμοποιηθεί, στην οποία οι διαχειριστές των δίσκων αποτελούσαν διαφορετικές διεργασίες και επικοινωνούσαν με τεχνικές IPC με το υπόλοιπο σύστημα. Κάθε νήμα εξυπηρετητής δίσκου έχει στην διάθεσή του ένα τμήμα του καναλιού επικοινωνίας, το οποίο και προσπελαύνει κυκλικά κατά την διάρκεια της λειτουργίας του. Ο ελεγκτής πραγματικού χρόνου αποτελεί ξεχωριστό νήμα. Η αποκλειστική του απασχόληση είναι ο καθορισμός των κύκλων εξυπηρέτησης με ακρίβεια. Για να πετύχει τον στόχο αυτό ζητά με περιοδικό τρόπο και αρκετά συχνά χρονικά σημεία από το ρολόι του συστήματος. Ο μέσος όρος των διαστημάτων μεταξύ δυο διαφορετικών στιγμών λήψεως χρονικών σημείων

αποτελεί υποπολλαπλάσιο του καθορισμένου χρόνου των κύκλων εξυπηρέτησης. Όταν ληφθεί ένα χρονικό σημείο, το οποίο είναι πολύ κοντά (ανάλογα με την χρονική ακρίβεια που παρέχει το υλικό) ή έχει ξεπεράσει για λίγο τον χρόνο έναρξης του επόμενου κύκλου, τότε ο ελεγκτής πραγματικού χρόνου μέσω του δρομολογητή αρχίζει άμεσα και κατά προτεραιότητα την εγγραφή των νέων υποαιτήσεων προσπέλασης δίσκου στο κανάλι επικοινωνίας. Μόλις η εγγραφή όλων των υποαιτήσεων για τον συγκεκριμένο κύκλο τελειώσει, ο ελεγκτής πραγματικού χρόνου προσθέτει μια ειδική εγγραφή σε κάθε τμήμα στο κανάλι επικοινωνίας που αντιστοιχεί σε ένα εξυπηρετητή δίσκου που υποδηλώνει την λήξη των εργασιών για τον συγκεκριμένο κύκλο, ενώ ταυτόχρονα ενεργοποιεί την λειτουργία των εξυπηρετητών δίσκου.

Όπως θα δούμε και στην συνέχεια του κεφαλαίου ο ελεγκτής πραγματικού χρόνου έχει σημαντικό ρόλο και στην λειτουργία του εξυπηρετητή μετάδοσης για τον χρονικό έλεγχο της αποστολής πακέτων δεδομένων. Η συνεχής παρακολούθηση των χρονικών σημείων επιτρέπει στην υπηρεσία ροής να διατηρεί μια σημαντική χρονική ακρίβεια κατά την λειτουργία της, επιτρέποντας μικρές αποκλίσεις στον καθορισμό της έναρξης των κύκλων, χωρίς σοβαρές παραβιάσεις προθεσμιών. Μία ακρίβεια, η οποία έχει όπως είναι φυσικό αντίκτυπο στον καθορισμό των κύκλων εξυπηρέτησης και κατά συνέπεια στην ποιότητα εξυπηρέτησης των αιτήσεων. Η προσέγγιση αυτή είναι προτιμότερη από την χρήση μετρητών για την διατήρηση της πολιτικής EDF που καθόριζαν την πιο πρόσφατη προθεσμία στη προηγούμενη έκδοση του συστήματος. Κι' αυτό γιατί τυχόν λάθη ακρίβειας συσσωρεύονται με την πάροδο του χρόνου με ευνόητες συνέπειες.

Εξυπηρετητές δίσκου: Τα νήματα-εξυπηρετητές δίσκου αναλαμβάνουν την διαχείριση ενός μέσου αποθήκευσης το καθένα. Η μοναδική μέριμνά τους είναι να εκτελέσουν όσο γίνεται πιο γρήγορα τις υποαιτήσεις δίσκου, όταν τους δοθεί η σχετική άδεια. Όπως αναφέρθηκε στην προηγούμενη υποενότητα, όταν ο ελεγκτής πραγματικού χρόνου διαπιστώσει ότι πρέπει να αρχίσει ένας κύκλος εξυπηρέτησης, εγγράφει μέσω του δρομολογητή τις κατάλληλες υποαιτήσεις δίσκου στο κανάλι επικοινωνίας DSR-Channel και στην συνέχεια ενεργοποιεί τους εξυπηρετητές δίσκου να αρχίσουν την εκτέλεση. Για να επιτευχθεί ο μηχανισμός αυτός, η λειτουργία των εξυπηρετητών δίσκου συγχρονίζεται με την χρήση σημαφόρων. Έτσι ο ελεγκτής πραγματικού χρόνου κλειδώνει όλες τις δυαδικές σημαφόρους, αναστέλλοντας με αυτόν τον τρόπο την λειτουργία των εξυπηρετητών δίσκου. Όταν διαπιστώσει ότι πρέπει να αρχίσει ο επόμενος κύκλος εξυπηρέτησης, απελευθερώνει τις σημαφόρους με αποτέλεσμα αυτόματα να ενεργοποιούνται τα νήματα των εξυπηρετητών δίσκου και να αρχίζει η επεξεργασία των υποαιτήσεων ανάκτησης δεδομένων του επόμενου κύκλου εξυπηρέτησης.

Οι εξυπηρετητές δίσκου εκτελούν σειριακά τις υποαιτήσεις δίσκου, που διαβάζουν από τις εγγραφές στο κανάλι επικοινωνίας, μέχρι να συναντήσουν την ειδική εγγραφή, που υποδηλώνει το τέλος των εργασιών του τρέχοντος κύκλου, οπότε και απενεργοποιούνται. Για την προσωρινή αποταμίευση των δεδομένων που ανακτώνται σε κάθε κύκλο, χρησιμοποιείται ένας ειδικός προσωρινός αποταμιευτής που ονομάζεται αποταμιευτής επικοινωνίας -Com Pool- (εφόσον διαμοιράζεται με τον εξυπηρετητή μετάδοσης για την αποστολή των δεδομένων) και διαχειρίζεται από τον αντίστοιχο διαχειριστή (Com Pool

6.3.2 Το κανάλι επικοινωνίας των εξυπηρετητών δίσκου (DSR-Channel)

[illegible]

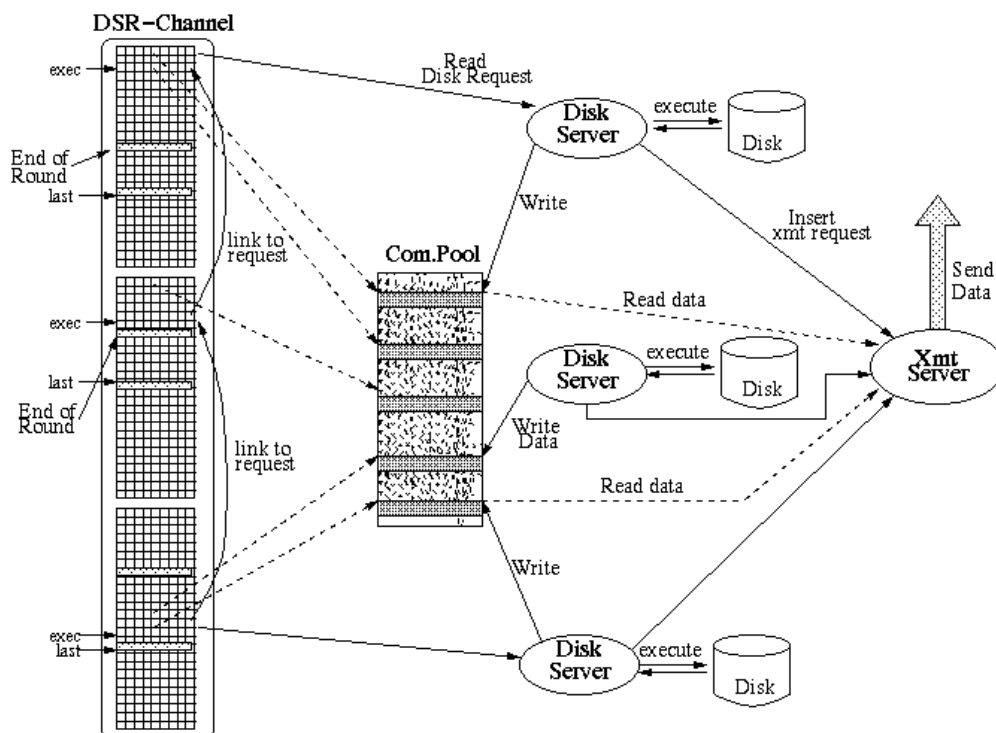
Κάθε τμήμα αποσκοπεί στην μεταφορά των υποαιτήσεων ανάκτησης δεδομένων σε ένα μοναδικό εξυπηρετητή δίσκου. Στην ουσία το κανάλι επικοινωνίας αποτελεί ένα πίνακα, ο οποίος περιέχει εγγραφές σταθερού μεγέθους. Κάθε εγγραφή αντιπροσωπεύει μια αίτηση προσπέλασης ενός συγκεκριμένου δίσκου. Η δομή των εγγραφών περιγράφεται γραφικά στο Σχήμα 58. Από το σχήμα μπορεί κανείς να παρατηρήσει ότι η υποαίτηση-εγγραφή περιέχει το αναγνωριστικό της αίτησης του πελάτη στην οποία ανήκει, το είδος της ενέργειας που πρέπει να λάβει χώρα (εγγραφή/ανάκτηση), την φυσική θέση των δεδομένων

στον δίσκο, καθώς επίσης και μια διεύθυνση στον αποταμιευτή επικοινωνίας για τις ανάγκες προσωρινής μνήμης. Επιπρόσθετα, η εγγραφή μεταφέρει πληροφορία σχετική με την μετάδοση των δεδομένων, όπως είναι οι διευθύνσεις δικτύου και τα πρωτόκολλα που πρέπει να χρησιμοποιηθούν.

Τα δύο τελευταία πεδία της εγγραφής συνιστούν ένα δείκτη σύνδεσης της τρέχουσας υποαίτησης με μία άλλη που βρίσκεται σε διαφορετικό τμήμα του καναλιού επικοινωνίας. Αυτό το σχήμα διασύνδεσης των υποαίτησεων, χρησιμεύει στην υποστήριξη της ανάκτησης αντικειμένων ροής στα οποία έχουν εφαρμοστεί τεχνικές διασκορπισμένης αποθήκευσης (fine striping).

Η αλυσίδα διασύνδεσης υποδεικνύει στον εξυπηρετητή μετάδοσης την λογική σειρά των τμημάτων που έχουν αποθηκευτεί σε διαφορετικές συσκευές δίσκου, ώστε να εξασφαλιστεί η σωστή ροή δεδομένων. Έτσι για να μπορέσει να αρχίσει η μετάδοση των δεδομένων μιας υποαίτησης δίσκου, που έχει ήδη ανακτήσει δεδομένα, θα πρέπει απαραίτητα η προηγούμενη αίτηση με την οποία είναι συνδεδεμένη, να έχει εκτελεστεί και τα αντίστοιχα δεδομένα της να έχουν αποσταλεί.

Κάθε τμήμα του καναλιού επικοινωνίας δεσμευμένο για ένα δίσκο, ακολουθεί τον μηχανισμό του κυκλικού πίνακα, ο οποίος οδηγείται από δύο δείκτες. Ο πρώτος δείκτης δείχνει την τελευταία υποαίτηση που έχει εκτελεστεί από το μέσο αποθήκευσης και ο δεύτερος την τελευταία εισαγωγή εγγραφής.



Σχήμα 59: Ο ρόλος του καναλιού επικοινωνίας των εξυπηρετητών δίσκου.

Η εγγραφή εκείνη που φέρει την ενδεικτική τιμή “ROUND” στο πεδίο που υποδηλώνει την ενέργεια της υποαίτησης, ενημερώνει τον αντίστοιχο εξυπηρετητή δίσκου για το τέλος των εργασιών του τρέχοντος κύκλου εξυπηρέτησης.

Στο Σχήμα 59 παρουσιάζεται ο ρόλος του καναλιού επικοινωνίας και η αλληλεπίδραση των τμημάτων του συστήματος μέσω αυτού με μεγαλύτερη λεπτομέρεια. Οι υποαίτησεις δίσκου εγγράφονται στο κανάλι επικοινωνίας από τον δρομολογητή της υπηρεσίας ροής, αυξάνοντας ταυτόχρονα τους δείκτες της τελευταίας εγγραφής. Οι εξυπηρετητές δίσκου αρχίζουν την επεξεργασία, διαβάζοντας την επόμενη εγγραφή της από την θέση που δείχνει ο δείκτης τελευταίας εκτέλεσης. Αναλύοντας τα πεδία της εγγραφής ο εξυπηρετητής δίσκου αρχίζει άμεσα την εκτέλεση. Αρχικά δεσμεύει χώρο στον αποταμιευτή επικοινωνίας, εκτελεί την απαραίτητη προσπάθεια στον δίσκο και τοποθετεί τα ανακτημένα δεδομένα στον κατάλληλο χώρο στον αποταμιευτή, ενημερώνοντας ταυτόχρονα το αντίστοιχο πεδίο της εγγραφής για την διεύθυνση του δεσμευμένου τμήματος.

Μόλις τα προηγούμενα πραγματοποιηθούν, ο εξυπηρετητής δίσκου δημιουργεί μια αίτηση μετάδοσης την οποία εισάγει στις κατάλληλες δομές που διαχειρίζεται ο εξυπηρετητής μετάδοσης. Η αίτηση μετάδοσης, όπως θα περιγραφεί στην συνέχεια του κειμένου, περιέχει το χρονικό σημείο εισαγωγής και ένα δείκτη στην υποαίτηση στο κανάλι επικοινωνίας, η οποία όπως περιγράφηκε περιέχει απαραίτητη πληροφορία σχετικά με τις διευθύνσεις τα πρωτόκολλα μετάδοσης και την θέση των δεδομένων, που πρέπει να αποσταλούν, στον αποταμιευτή επικοινωνίας. Στην συνέχεια αυξάνει τον δείκτη της τελευταίας εκτέλεσης και συνεχίζει την λειτουργία του όπως προηγουμένως. Η ίδια διαδικασία επαναλαμβάνεται με ένα κυκλικό μηχανισμό μέσα στο κανάλι, μέχρι ο εξυπηρετητής δίσκου να διαβάσει την ειδική εγγραφή που αναπαριστά το τέλος των εργασιών του κύκλου. Σε αυτό το σημείο το νήμα του εξυπηρετητή αδρανοποιείται και περιμένει την ενεργοποίησή του για τον επόμενο κύκλο.

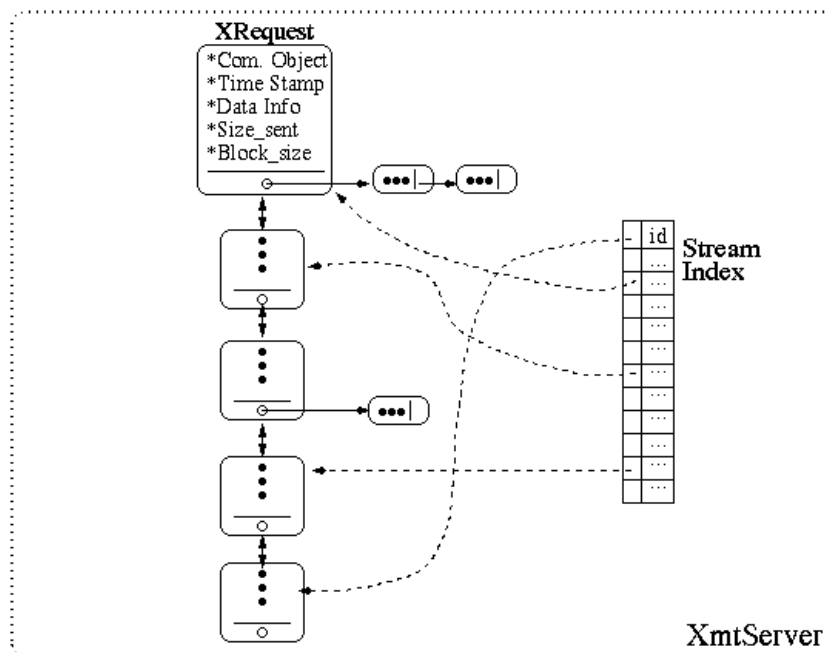
6.4 Ο μηχανισμός μετάδοσης δεδομένων συνεχόμενης ροής

Στην ενότητα αυτή εξετάζεται ο σχεδιασμός και η ανάπτυξη του εξυπηρετητή μετάδοσης της υπηρεσίας ροής. Λαμβάνοντας υπόψη τα αποτελέσματα των πειραμάτων του 4ου κεφαλαίου και μέσα από μια σειρά πειραματικών εφαρμογών διαφόρων πολιτικών δρομολόγησης της αποστολής δεδομένων που ακολουθούν στις επόμενες υποενότητες, παρουσιάζεται αναλυτικά ο προτεινόμενος μηχανισμός δρομολόγησης του εξυπηρετητή μετάδοσης.

6.4.1 Ο εξυπηρετητής μετάδοσης δεδομένων

Ένας από τους πιο σημαντικούς στόχους ενός εξυπηρετητή αποθήκευσης είναι να παρέχει ποιότητα στις υπηρεσίες του, ενώ ταυτόχρονα να μεγιστοποιεί την απόδοση του συστήματος, έτσι ώστε να ικανοποιεί μεγάλο αριθμό χρηστών. Συνεπώς η ικανότητα του εξυπηρετητή να ικανοποιεί τις απαιτήσεις των πελατών και να αποφεύγει τυχόν αρνητικές επιπτώσεις της εξυπηρέτησης των αιτήσεων στην πλευρά του πελάτη, αναδεικνύει την αξία

και την χρησιμότητα του εξυπηρετητή. Σύμφωνα με τα προηγούμενα η διαδικασία μετάδοσης των ανακτώμενων δεδομένων έχει ιδιαίτερη σημασία, καθώς έχει άμεσο αντίκτυπο στην ποιότητα της υπηρεσίας στον πελάτη. Ο εξυπηρετητής δεδομένων ροής πρέπει να μεταδίδει δεδομένα στον πελάτη με τον κατάλληλο ρυθμό μετάδοσης και να αποφεύγει εκρηκτικές μεταδόσεις (burst), ώστε να έχουμε θετικά αποτελέσματα.



Σχήμα 60: Οι δομές υποστήριξης του εξυπηρετητή μετάδοσης δεδομένων

Οι δομές υποστήριξης του εξυπηρετητή μετάδοσης-XmtServer, που αποτελεί τμήμα της υπηρεσίας ροής, φαίνονται στο Σχήμα 60. Η δομή Xrequest περιέχει δεδομένα σχετικά με την μετάδοση ενός αντικειμένου ροής. Στην δομή αυτή περιέχονται το αντικείμενο επικοινωνίας, που υλοποιεί την μετάδοση των δεδομένων μέσω δικτύου, ένα χρονικό σημείο, ένα δείκτη στο κανάλι επικοινωνίας των εξυπηρετητών δίσκου (που περιέχει την πληροφορία μετάδοσης και προσπέλασης των δεδομένων στον αποταμιευτή) και τέλος μεταβλητές που αναπαριστούν το μέγεθος των δεδομένων που έχουν αποσταλεί και το μέγεθος του τρέχοντος τμήματος.

Το χρονικό σημείο αντιπροσωπεύει προθεσμία και ο ρόλος του θα περιγραφεί στην συνέχεια του κειμένου. Οι εξυπηρετητές δίσκου κάθε φορά που ένα τμήμα του αντικειμένου είναι διαθέσιμο, εισάγουν τα νέα στοιχεία μετάδοσης στην XRequest δομή αν το προηγούμενο τμήμα έχει πλήρως μεταδοθεί, ενώ διαφορετικά δημιουργείται μια νέα δομή που ενώνεται με την μορφή λίστας στην αρχική αίτηση. Η δομή Stream index αποτελεί ένα πίνακα κατακερματισμού (hashing table) με δείκτες πάλι σε δομές XRequest και χρησιμοποιείται για γρήγορη προσπέλαση της αίτησης μετάδοσης δεδομένων που αντιστοιχεί σε κάποιο αντικείμενο ροής με βάση το αναγνωριστικό του. Η προσπέλαση μιας αίτησης μετάδοσης δεδομένων γίνεται κάθε φορά που ένας εξυπηρετητής δίσκου ανακτά ένα τμήμα δεδομένων κάποιου αντικειμένου και χρειάζεται ενημέρωση της παλαιότερης αίτησης για την συνέχιση της μετάδοσης.

Στα πειράματα που ακολουθούν εφαρμόστηκαν διάφορες πολιτικές στον μηχανισμό μετάδοσης δεδομένων και καταγράφηκαν τα αποτελέσματα που είχε η εφαρμογή τους στην πλευρά του πελάτη. Απώτερος στόχος είναι η εύρεση του κατάλληλου μηχανισμού δρομολόγησης αποστολής δεδομένων, ο οποίος θα έχει τα καλύτερα αποτελέσματα στην πλευρά του πελάτη.

Αναφορά για τα πρωτόκολλα μετάδοσης έγινε στο 4ο κεφάλαιο, ενώ σε όλα τα πειράματα της διατριβής η μετάδοση των δεδομένων ροής γίνεται με χρήση των UDP/IP πρωτοκόλλων.

6.4.2 Πείραμα αποστολής δεδομένων σε τμήματα

Για τις ανάγκες του πειράματος έχουν αποθηκευτεί διάφορα αντικείμενα ροής στον εξυπηρετητή αποθήκευσης. Στόχος είναι να καταγράψουμε τα αποτελέσματα στην πλευρά ενός χρήστη, όταν χρησιμοποιούμε την υπηρεσία αποθήκευσης δεδομένων ροής που σχεδιάστηκε, ενώ ειδικά ο εξυπηρετητής μετάδοσης-XmtServer αποστέλλει τα δεδομένα κάθε ανακτώμενου τμήματος συνεχόμενα ακολουθώντας FIFO πολιτική σε όλους τους πελάτες.

Στα πειράματα της ενότητας αλλά και στα υπόλοιπα που θα ακολουθήσουν και είναι σχετικά με την δρομολόγηση της μετάδοσης δεδομένων, έχει απομονωθεί το υποσύστημα των δίσκων ώστε να μην επηρεάζει σε αυτή την φάση τα αποτελέσματα. Η απομόνωση αυτή έχει σαν συνέπεια να θεωρούμε ότι τα κατάλληλα δεδομένα είναι έγκαιρα διαθέσιμα στον αποταμιευτή επικοινωνίας για κάθε αντικείμενο ροής που εξυπηρετείται. Όλα τα πειράματα του κεφαλαίου έλαβαν χώρα στο πειραματικό περιβάλλον που περιγράφηκε στην ενότητα 5.2.

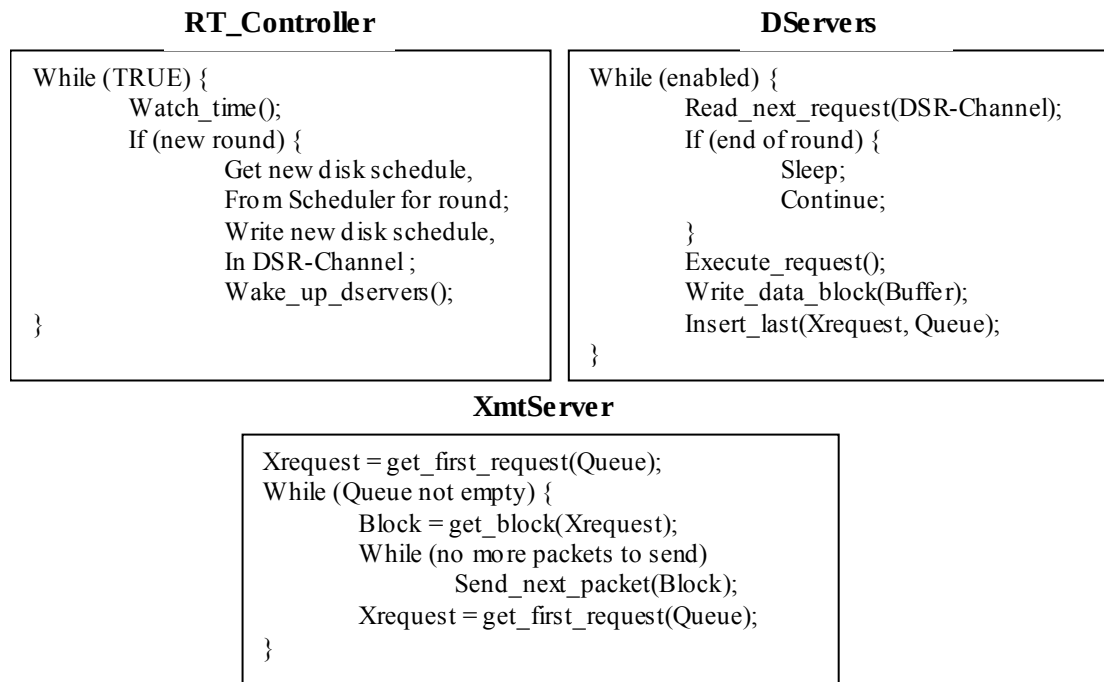
Ο χρήστης που παρακολουθούμε στην αρχή ζητά την αναπαραγωγή ενός αντικειμένου ροής. Μετά 10 δεύτερα αρχίζει η σταδιακή εισαγωγή νέων χρηστών με διαφορά ενός δευτέρου, με στόχο την παρακολούθηση της επίδρασης της σταδιακής εισαγωγής νέων απαιτήσεων στο σύστημα στην εξυπηρέτηση της πρώτης αίτησης. Νέες αιτήσεις αναπαραγωγής εξακολουθούν να εισάγονται στο σύστημα ανά δεκάδα με μεσοδιαστήματα δέκα δευτερολέπτων, μέχρι τον τελικό αριθμό των 90 χρηστών. Με αυτό τον τρόπο επιθυμούμε να καταγράψουμε την συμπεριφορά του συστήματος σε καταστάσεις υψηλού φόρτου εργασίας.

Σε αυτό το σημείο πρέπει να σημειωθεί ότι ο εξυπηρετητής μετάδοσης τρέχει σε ένα σταθμό εργασίας Ultra One διασυνδεδεμένο στο τοπικό ATM δίκτυο, ενώ ο πελάτης στον οποίο καταγράφονται οι πειραματικές μετρήσεις υποστηρίζεται από ένα Sparc-4 υπολογιστή. Τα αντικείμενα ροής έχουν αποθηκευτεί σε τμήματα μεγέθους 192 Kbytes, ενώ για την αποστολή τους διασπώνται σε 4 πακέτα μεγέθους 48K. Στην πλευρά του πελάτη καταγράφουμε τους χρόνους μεταξύ διαδοχικών αφίξεων πακέτων και τον ρυθμό άφιξης ο οποίος υπολογίζεται ανά 8 πακέτα.

Με δεδομένο ότι τα πειραματικά αντικείμενα ροής αποτελούν MPEG1 video με συμπίεση σταθερού ρυθμού (Constant Bit Rate – CBR), ο επιθυμητός ρυθμός άφιξης δεδομένων στον πελάτη είναι 1.5Mbps ή 192KBps, ενώ ο ιδανική καθυστέρηση άφιξης

διαδοχικών πακέτων είναι 250msec. Στην συνέχεια ακολουθεί με λεπτομέρεια ο αλγόριθμος που εφαρμόστηκε για αυτό το πείραμα σε μορφή ψευδοκώδικα. Εξετάζονται τα νήματα εξυπηρετητών δίσκου, του ελεγκτή πραγματικού χρόνου και τέλος η δρομολόγηση του εξυπηρετητή μετάδοσης.

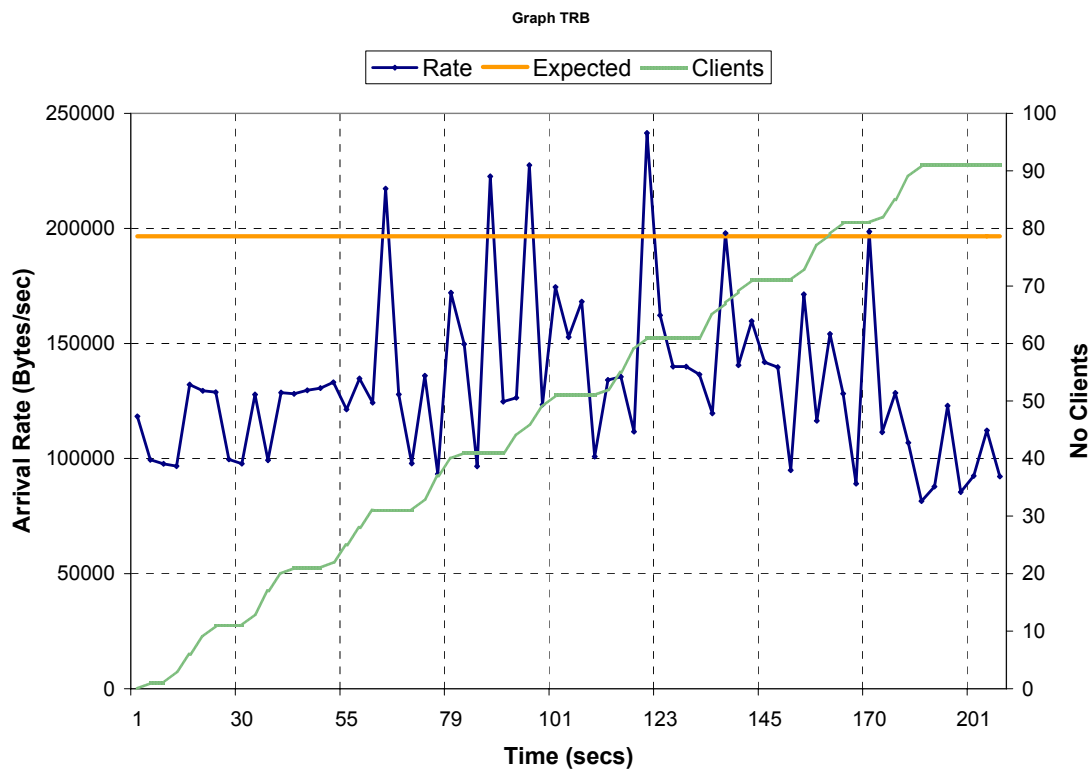
Αλγόριθμος:



Οι γραφικές παραστάσεις που θα ακολουθήσουν σε αυτό και στα επόμενα πειράματα (ενότητες 6.4.3, 6.4.4, 6.4.5) είναι δύο ειδών. Η πρώτη καταγράφει τον ρυθμό άφιξης δεδομένων στον πελάτη (πρωτεύων κάθετος άξονας) σε σχέση με τον χρόνο (οριζόντιος άξονας) και σε αντιπαράθεση με τον αριθμό των ενεργών πελατών που υπάρχουν στον εξυπηρετητή (δευτερεύων κάθετος άξονας). Σε κάθε γραφική παράσταση υπάρχει μια ευθεία γραμμή, η οποία τέμνει τον πρωτεύοντα κάθετο άξονα στην τιμή 192KBps (ή 1.5 Mbps) και αντιπροσωπεύει τον επιθυμητό ρυθμό άφιξης δεδομένων στον πελάτη. Το δεύτερο είδος διαγραμμάτων προκύπτει από την καταγραφή των χρονικών διαστημάτων μεταξύ δύο διαδοχικών αφίξεων πακέτων από τον εξυπηρετητή. Η πειραματική αυτή μέτρηση έχει στόχο να καταγράψει τις διαφοροποιήσεις στους χρόνους αφίξεων των πακέτων και κατά συνέπεια να απεικονίσει το jitter της μετάδοσης. Η διαφοροποίηση αυτή μπορεί να είναι θετική ή αρνητική ή και μηδέν, ανάλογα αν το χρονικό διάστημα μεταξύ των αφίξεων διαδοχικών πακέτων είναι μεγαλύτερο μικρότερο ή ίσο με την αναμενόμενη καθυστέρηση (250 msec σύμφωνα με τα δεδομένα των πειραμάτων). Όπως και στο προηγούμενο είδος γραφικών παραστάσεων έτσι και σε αυτή την περίπτωση υπάρχει ο δευτερεύων κάθετος άξονας που αντιπροσωπεύει τον αριθμό των ενεργών πελατών στον εξυπηρετητή κατά την διάρκεια των μετρήσεων.

Όπως φαίνεται από το Σχήμα 61 οι μετρήσεις δείχνουν ότι η εξυπηρέτηση του πελάτη είναι προβληματική καθώς ο ρυθμός άφιξης δεδομένων απέχει πολύ από τον αναμενόμενο,

λόγω της απώλειας πολλών πακέτων που προκαλείται από την εκρηκτική μετάδοση των τμημάτων του αντικειμένου ροής. Η μείωση αυτή στον ρυθμό άφιξης τείνει να γίνει μεγαλύτερη όσο ο αριθμός των πελατών προσεγγίζει υψηλά επίπεδα.

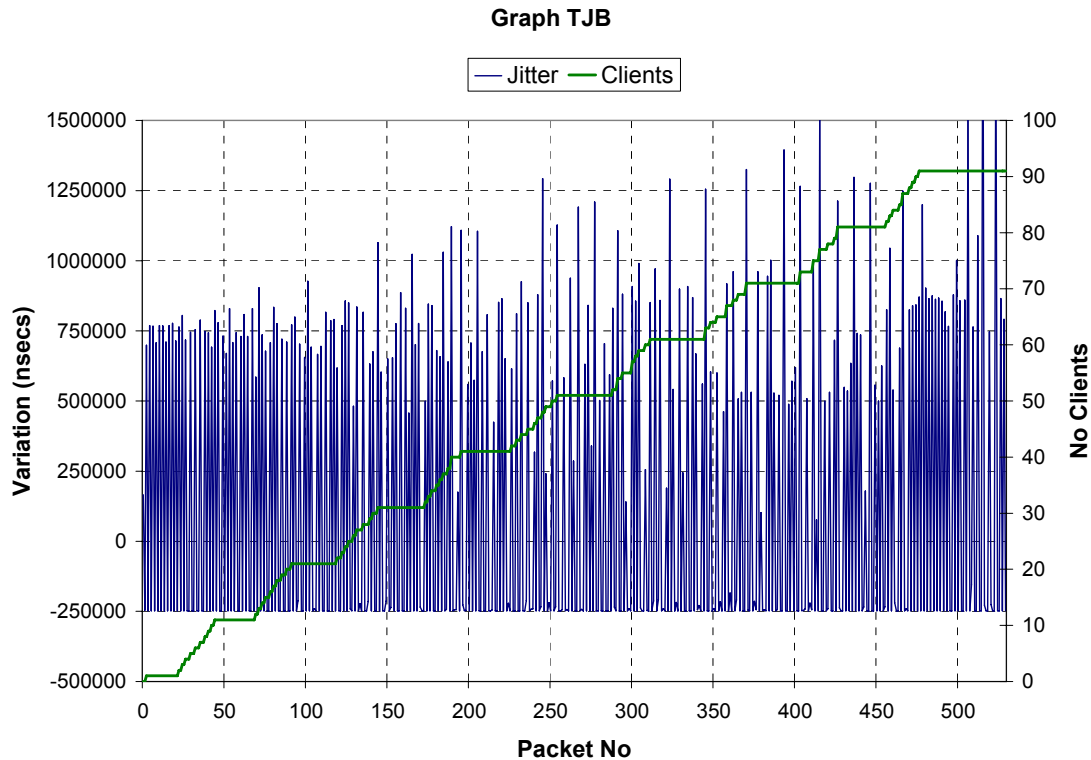


Σχήμα 61: Ρυθμός άφιξης δεδομένων στον πελάτη (Μετρήσεις Πειράματος)

Στο σχήμα εικονίζεται ο ρυθμός άφιξης δεδομένων στην πλευρά του πελάτη που αιτήθηκε την αναπαραγωγή ενός αντικειμένου ροής, όταν η δρομολόγηση στέλνει κάθε τμήμα δεδομένων που είναι διαθέσιμο συνεχόμενα με κυκλική εξυπηρέτηση των πελατών. Όπως φαίνεται στο σχήμα ο ρυθμός απέχει κατά πολύ από τον επιθυμητό (~192 Kbytes/sec), λόγω απώλειας πολλών πακέτων που κατά πάσα πιθανότητα οφείλεται στην εκρηκτική μετάδοση των τμημάτων δεδομένων.

Το επόμενο σχήμα (Σχήμα 62) περιγράφει τις διαφοροποιήσεις στους χρόνους άφιξης διαδοχικών πακέτων που προκαλούνται από τον εφαρμοζόμενο αλγόριθμο αποστολής πακέτων στον εξυπηρετητή μετάδοσης. Παρατηρώντας κανείς το σχήμα γίνεται αμέσως αντιληπτό ότι υπάρχουν μεγάλες διαφοροποιήσεις στα χρονικά διαστήματα μεταξύ διαδοχικών αφίξεων πακέτων που κυμαίνονται από -250msec μέχρι πάνω από 1 sec.

Ευνόητο είναι ότι η αρνητική αυτή συμπεριφορά οφείλεται στον μηχανισμό δρομολόγησης που εφαρμόζεται. Οι πολύ μικροί χρόνοι οφείλονται στην συνεχόμενη μετάδοση σε πακέτα ενός τμήματος του αντικειμένου ροής, ενώ τα μεγάλα χρονικά διαστήματα οφείλονται στην διακοπή της μετάδοσης μέχρι την ανάκτηση του επόμενου τμήματος του αντικειμένου σε συνδυασμό με την απώλεια αρκετών πακέτων λόγω της εκρηκτικής μετάδοσης.



Σχήμα 62: Καθυστέρηση μεταξύ διαδοχικών πακέτων στον πελάτη (Μετρήσεις Πειράματος)
 Παραπάνω απεικονίζεται η καθυστέρηση μεταξύ διαδοχικών αφίξεων πακέτων στη πλευρά του πελάτη, όταν η δρομολόγηση στέλνει κάθε τμήμα του αντικειμένου ροής συνεχόμενα με κυκλική εξυπηρέτηση των πελατών. Ταυτόχρονα καταγράφεται ο αριθμός των ενεργών πελατών στον εξυπηρετητή. Το jitter μετάδοσης στην προκείμενη περίπτωση είναι μεγάλο λόγω της εκρηκτικής μετάδοσης των τμημάτων δεδομένων και την διακοπή της μετάδοσης μέχρι την ανάκτηση του επόμενου τμήματος.

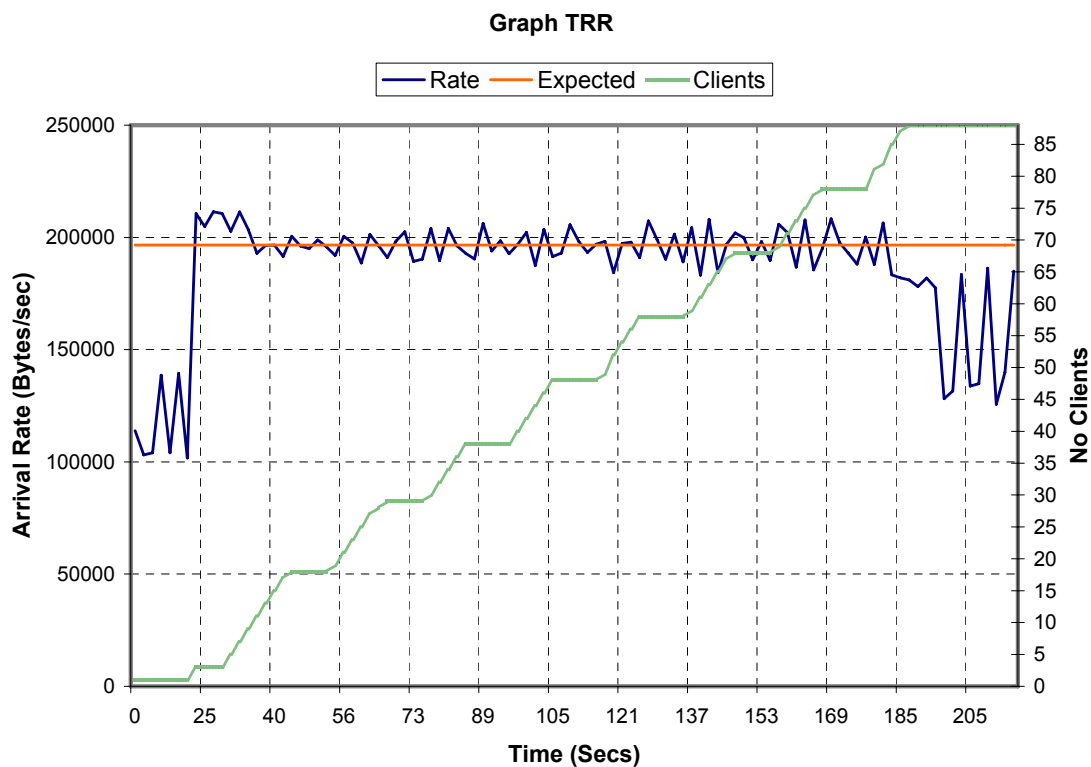
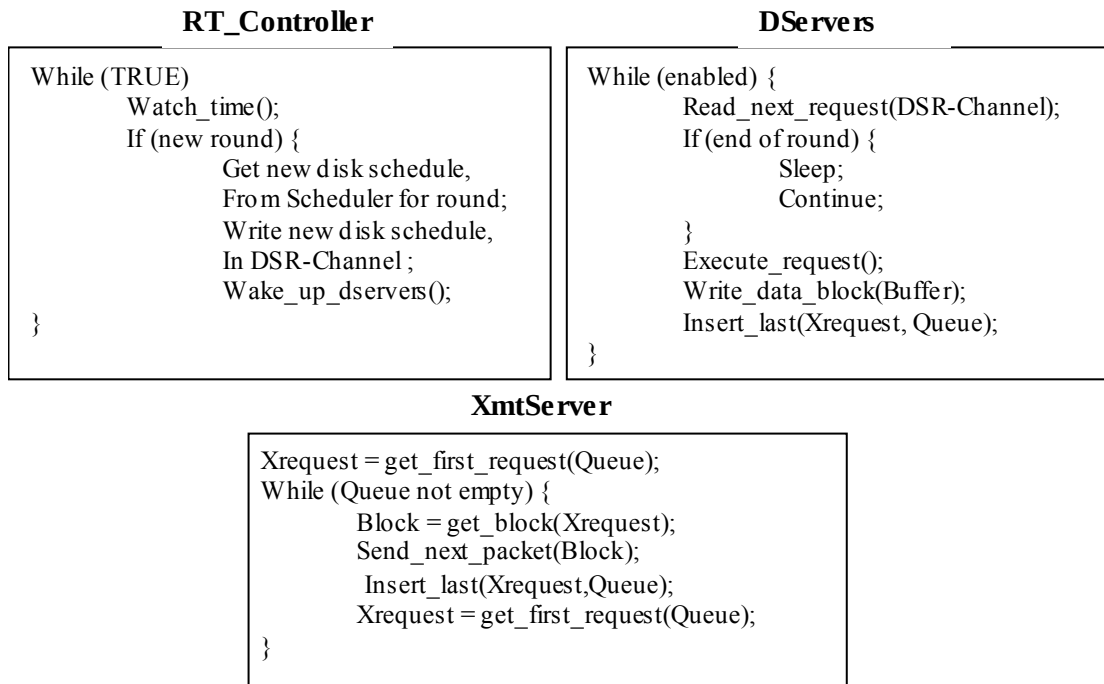
Από τα αποτελέσματα του προηγούμενου πειράματος γίνεται αντιληπτό ότι χρειάζεται να αναπτυχθεί ένας διαφορετικός πιο αποδοτικός μηχανισμός δρομολόγησης.

6.4.3 Πείραμα πολύπλεξης πακέτων πελατών

Σε αυτή την υποενότητα τα δεδομένα είναι τα ίδια με την προηγούμενη. Εδώ διαφοροποιούμε την δρομολόγηση στον εξυπηρετητή μετάδοσης, επιτρέποντας την πολύπλεξη των πακέτων διαφορετικών αντικειμένων. Δηλαδή στέλνουμε διαδοχικά πακέτα που ανήκουν σε δεδομένα διαφορετικών χρηστών ακολουθώντας κυκλική σειρά.

Η νέα πολιτική δρομολόγησης της μετάδοσης δεδομένων που εφαρμόζεται στην υπηρεσία διαχείρισης δεδομένων ροής για τις ανάγκες του τρέχοντος πειράματος περιγράφεται στην συνέχεια με μορφή ψευδοκώδικα για κάθε νήμα εργασίας.

Αλγόριθμος:

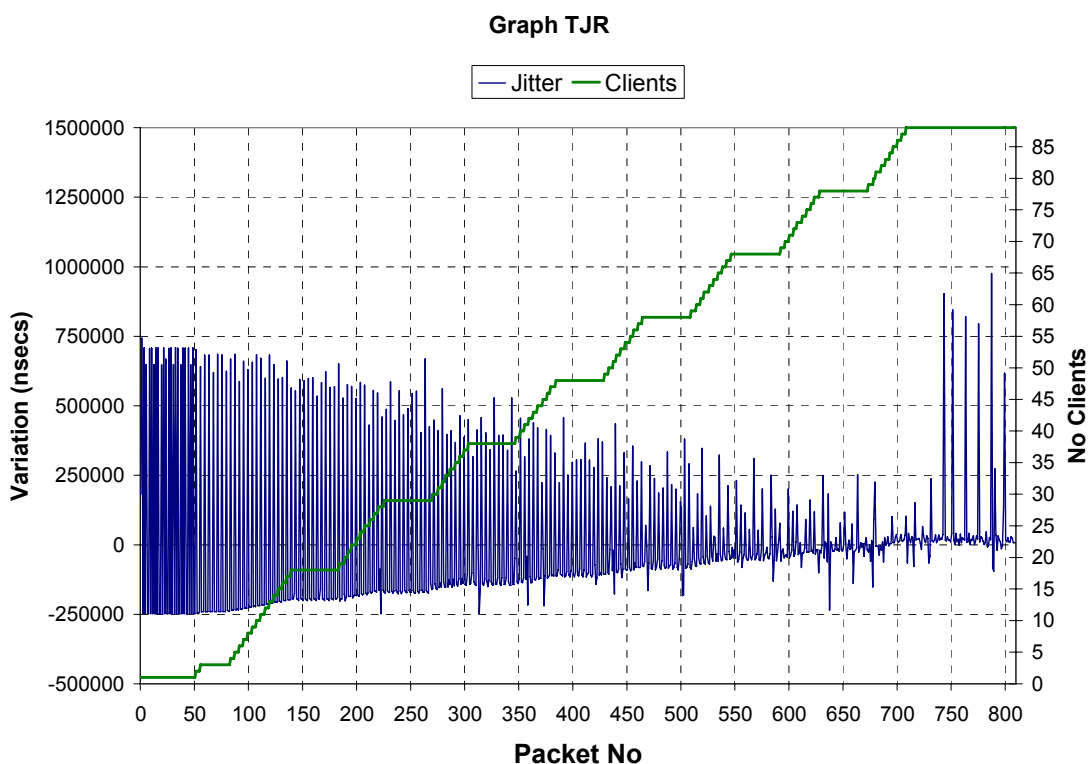


Σχήμα 63: Ρυθμός άφιξης δεδομένων στον πελάτη (Μετρήσεις Πειράματος)
Στο σχήμα εικονίζεται ο ρυθμός άφιξης δεδομένων στην πλευρά του πελάτη, όταν η δρομολόγηση πολυπλέκει τα πακέτα διαφορετικών πελατών κυκλικά. Τα αποτελέσματα του πειράματος βελτιώνονται από αυτά του πρώτου πειράματος όταν ο αριθμός των χρηστών αυξάνει. Αυτό έχει σαν αποτέλεσμα οι αρνητικές επιδράσεις της εκρηκτικής μετάδοσης να αμβλύνονται λόγω της πολυπλεξης των πακέτων.

Από τις μετρήσεις συμπεραίνουμε ότι αμέσως μετά την εισαγωγή νέων χρηστών το πρόβλημα της απώλειας πακέτων βελτιώνεται καθώς η πολύπλεξη των πακέτων διαφορετικών αντικειμένων ροής δημιουργεί καθυστερήσεις αποφεύγοντας τη εκρηκτική μετάδοση. Για τον πρώτο όμως πελάτη το πρόβλημα εξακολουθεί να είναι έντονο, όπως φαίνεται στο Σχήμα 63.

Έτσι ενώ για μικρό αριθμό χρηστών ο ρυθμός άφιξης δεδομένων είναι πολύ χαμηλός, όταν αυξάνει ο αριθμός των χρηστών που εξυπηρετούνται και αμβλύνεται το φαινόμενο της απώλειας πακέτων, ο ρυθμός άφιξης κινείται κοντά στην περιοχή του 192Kbytes/sec με αρκετές όμως αισθητές διακυμάνσεις.

Διαπιστώνεται συνεπώς ότι με την αλλαγή της πολιτικής δρομολόγησης της μετάδοσης βελτιώνονται τα αποτελέσματα στην πλευρά του πελάτη. Αυτό μπορεί να γίνει εύκολα κατανοητό αν ανατρέξουμε στα πειράματα λήψης δεδομένων για τον συγκεκριμένο πελάτη που έλαβαν χώρα στο 0ο κεφάλαιο (Σχήμα 36). Οι εκρηκτικές μεταδόσεις άνω των 75 Mbps προκαλούν σημαντικές απώλειες πακέτων στον πελάτη (Sparc20), ενώ οι τεχνικές πολύπλεξης πακέτων, που έχουν σαν συνέπεια την διαίρεση του μέγιστου αριθμού μετάδοσης (122Mbps) με τον αριθμό των ενεργών πελατών, μειώνει τον ρυθμό αποστολής κάτω από τα επίπεδα του 75Mbps, με άμεση απόρροια την αποφυγή του φαινομένου απώλειας δεδομένων.



Σχήμα 64: Καθυστέρηση διαδοχικών πακέτων στον πελάτη (Μετρήσεις Πειράματος)
Καταγραφή της καθυστέρησης διαδοχικών αφίξεων πακέτων, όταν η δρομολόγηση πολύπλεκει τα πακέτα διαφορετικών πελατών κυκλικά. Για μικρό αριθμό πελατών τα αποτελέσματα είναι τα ίδια με το προηγούμενο πείραμα. Με την αύξηση των ενεργών χρηστών και την δρομολόγηση με δυνατότητα πολύπλεξης διαφορετικών πακέτων περιορίζεται η εκρηκτική μετάδοση για κάθε πελάτη και βελτιώνεται η διαφοροποίηση στους χρόνους αύξησης.

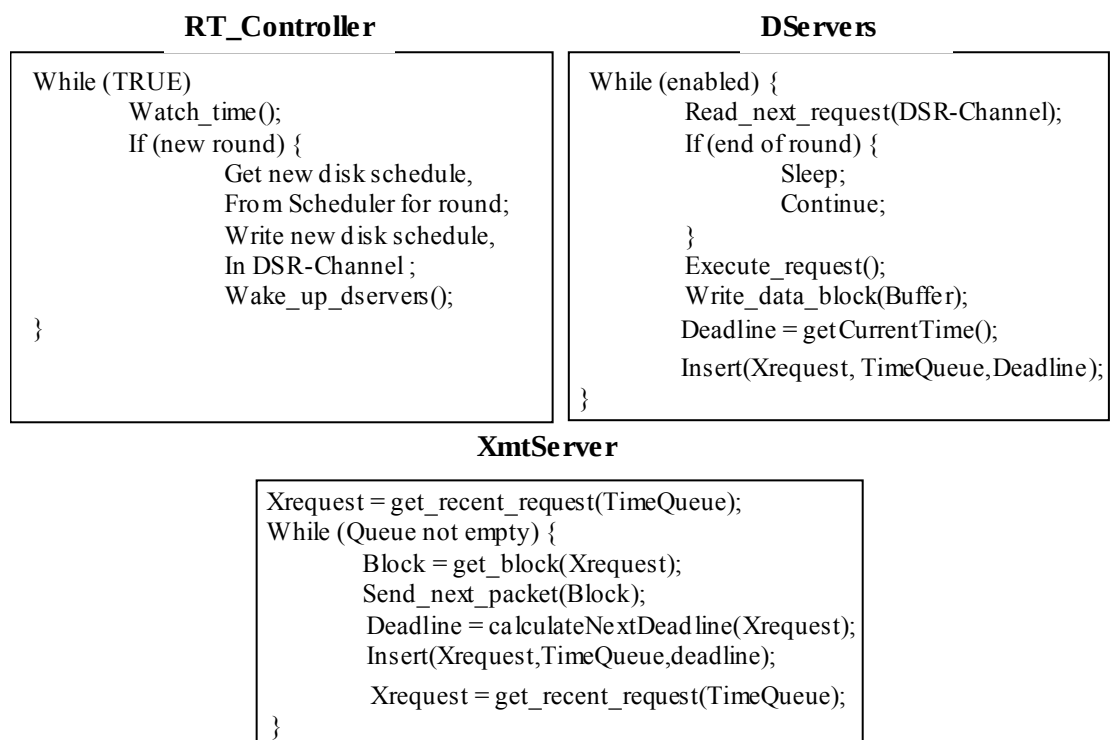
Παρόμοια αποτελέσματα έχουμε κατά την καταγραφή των καθυστερήσεων διαδοχικών αφίξεων πακέτων στον πελάτη. Όταν ο αριθμός των χρηστών είναι μικρός και συνεπώς η πολύπλεξη περιορισμένη παρατηρούμε μεγάλη διαφοροποίηση στους χρόνους διαδοχικών πακέτων. Με την σταδιακή όμως αύξηση χρηστών πραγματοποιείται ακούσια ένας έλεγχος στην μετάδοση δεδομένων, ο οποίος μειώνει το παρατηρούμενο jitter. Τα αποτελέσματα αυτά γίνονται αντιληπτά από την γραφική απεικόνιση των μετρήσεων στο Σχήμα 64.

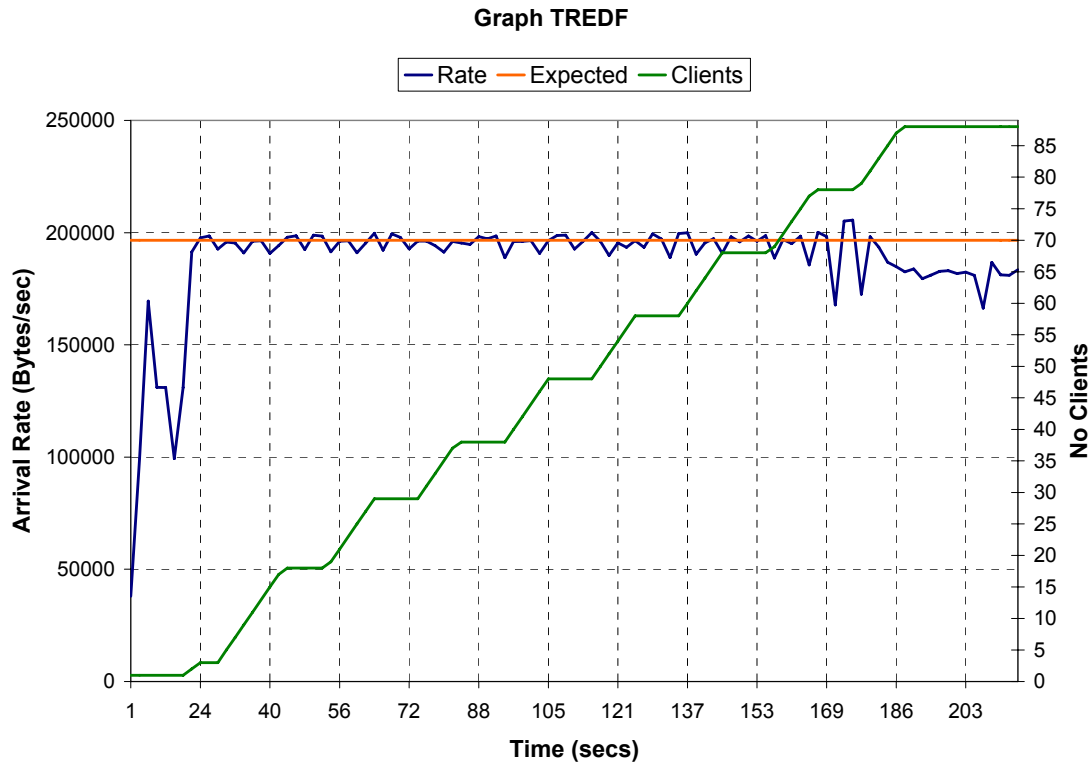
6.4.4 Πείραμα εφαρμογής δρομολόγησης EDF

Σε αυτό το πείραμα αντικαθιστούμε την πολιτική κυκλικής εξυπηρέτησης και επιβάλλουμε στον εξυπηρετητή μετάδοσης δρομολόγηση βάσει προθεσμιών (EDF), ενώ ταυτόχρονα υποθέτουμε ότι είναι αναγκαία η αποστολή των δεδομένων στους πελάτες με ρυθμό ίσο με τον ρυθμό κατανάλωσης των αντικειμένων.

Σύμφωνα με την πολιτική αυτή ανατίθεται σε κάθε πακέτο μια προθεσμία μετάδοσης. Η προθεσμία αυτή προκύπτει από τα χαρακτηριστικά του αντικειμένου ροής, που ορίζουν τον απαιτούμενο ρυθμό κατανάλωσης δεδομένων και το μέγεθος των πακέτων. Έτσι ο εξυπηρετητής μετάδοσης διατηρεί μια ουρά προθεσμιών για τις υποαιτήσεις μετάδοσης. Εφαρμόζοντας σε αυτό το πείραμα την πολιτική EDF ο εξυπηρετητής μετάδοσης αρχίζει να αποστέλλει πακέτα συμβουλευοντας την ουρά προθεσμιών, όπως μπορεί να παρατηρήσει κανείς από την παρακάτω περιγραφή του αλγορίθμου.

Αλγόριθμος:



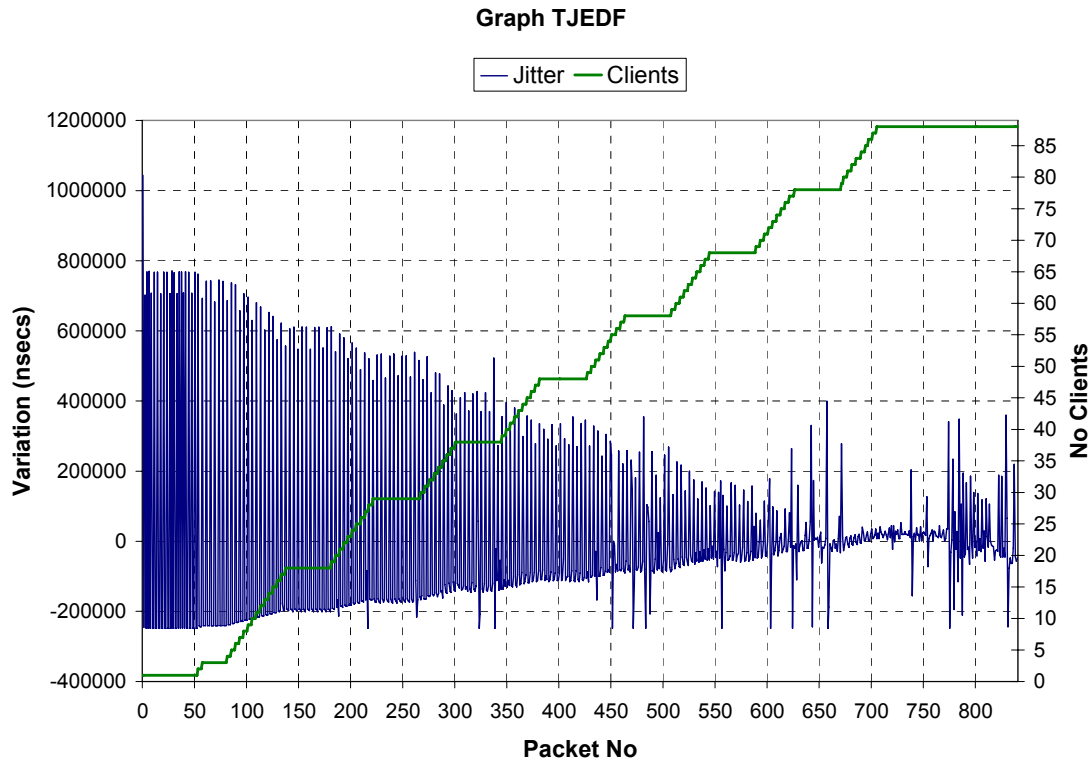


Σχήμα 65: Ρυθμός άφιξης δεδομένων στον πελάτη (Μετρήσεις Πειράματος)

Στο σχήμα εμφανίζεται ο ρυθμός άφιξης δεδομένων στην πλευρά του πελάτη, όταν η δρομολόγηση εφαρμόζει τον EDF αλγόριθμο. Για μικρό αριθμό πελατών παρατηρούνται απώλειες πακέτων και συνεπώς ο ρυθμός άφιξης δεδομένων είναι αισθητά μικρότερος του επιθυμητού. Αυξάνοντας τον αριθμό των χρηστών έχουμε καλύτερα αποτελέσματα.

Και σε αυτή την περίπτωση δεν έχουμε τα θεμιτά αποτελέσματα όπως φαίνεται από τις παραπάνω μετρήσεις. Για μικρό αριθμό πελατών ο ρυθμός αύξησης δεδομένων είναι μειωμένος για λόγους όμοιους με αυτούς των δύο προηγούμενων πειραμάτων. Όταν το σύστημα εξυπηρετεί αρκετούς πελάτες, η εφαρμογή της δρομολόγησης προθεσμιών επιφέρει σχετικά καλά αποτελέσματα. Αξιοσημείωτο είναι ότι ο εφαρμοζόμενος στο πείραμα μηχανισμός δρομολόγησης, επιφέρει ελαφρώς καλύτερα αποτελέσματα, καθώς καταγράφονται μικρότερες διακυμάνσεις στον ρυθμό άφιξης των δεδομένων.

Επόμενο είναι όμοια αποτέλεσμα με το προηγούμενο σχήμα να παίρνουμε και κατά την καταγραφή των καθυστερήσεων διαδοχικών αφίξεων πακέτων στον πελάτη. Όπως προκύπτει από το Σχήμα 66 η διαφοροποίηση στις καθυστερήσεις άφιξης των πακέτων είναι μεγάλη όταν ο αριθμός των χρηστών είναι μικρός. Με την σταδιακή αύξηση των χρηστών που εξυπηρετούνται το jitter κατά την μετάδοση περιορίζεται σε χαμηλά επίπεδα.



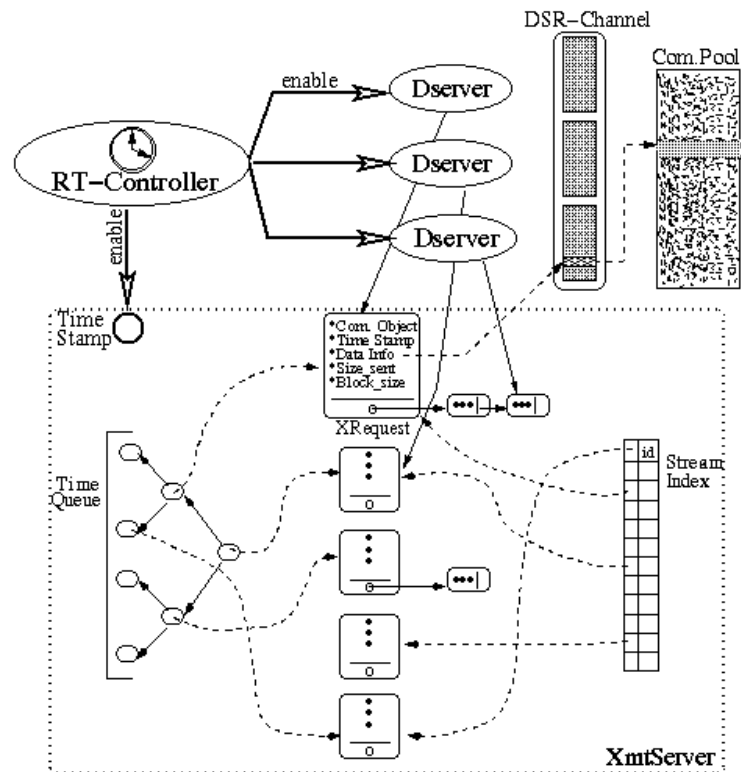
Σχήμα 66: Καθυστέρηση διαδοχικών πακέτων στον πελάτη (Πείραμα 4.2)
Καταγραφή των καθυστερήσεων διαδοχικών αφίξεων πακέτων, όταν η δρομολόγηση εφαρμόζει την πολιτική EDF. Οι μεγάλες διακυμάνσεις ομαλοποιούνται όταν αυξάνει ο αριθμός των χρηστών, αλλά για υψηλό φόρτο εργασίας η διαφοροποίηση των καθυστερήσεων γίνεται πάλι αισθητή.

Αν συγκρίνουμε τα αποτελέσματα των δύο τελευταίων πειραμάτων διαπιστώνουμε ότι η εφαρμογή της EDF πολιτικής σε σύγκριση με την απλή πολύπλεξη πακέτων διαφορετικών πελατών κατά την μετάδοση, έχει καλύτερα αποτελέσματα μειώνοντας το jitter χαμηλότερα σε περιπτώσεις μεσαίου φόρτου εργασίας.

6.4.5 Σχεδιασμός του μηχανισμού δρομολόγησης στον εξυπηρετητή μετάδοσης

Προσπαθώντας να αποφύγουμε τα αρνητικά σημεία των προηγούμενων πολιτικών δρομολόγησης της μετάδοσης, εφαρμόζουμε ένα μηχανισμό δρομολόγησης στον εξυπηρετητή μετάδοσης (XmtServer), με στόχο τον έλεγχο του ρυθμού και του jitter κατά την αποστολή των δεδομένων.

Ο μηχανισμός αποτελεί non work-conserving προσέγγιση (κεφάλαιο 2), και περιγράφεται γραφικά στο επόμενο σχήμα. Οι εξυπηρετητές δίσκου, κάθε φορά που ένα τμήμα αντικειμένου ροής έχει ανακτηθεί στον αποταμιευτή επικοινωνίας, ενημερώνουν τον XmtServer ότι νέα δεδομένα είναι έτοιμα για μετάδοση. Ο XmtServer σχηματίζει τα πακέτα μετάδοσης και αναθέτει τις κατάλληλες προθεσμίες σύμφωνα με τις απαιτήσεις του αντικειμένου ροής. Ο RT-Controller όπως φαίνεται στο σχήμα ενεργοποιεί περιοδικά τους εξυπηρετητές με περίοδο τον κύκλο εξυπηρέτησης. Σε υποδιαιρέσεις αυτού του κύκλου παίρνει συνεχώς χρονικά σημεία από το ρολόι του συστήματος, ενώ ταυτόχρονα ενεργοποιεί τον εξυπηρετητή μετάδοσης και τον ενημερώνει με το τρέχων χρονικό σημείο.



Σχήμα 67: Ο μηχανισμός δρομολόγησης της μετάδοσης δεδομένων

Ο εξυπηρετητής μετάδοσης συνεργάζεται με τον ελεγκτή πραγματικού χρόνου για την δρομολόγηση της αποστολής πακέτων. Η εφαρμοζόμενη δρομολόγηση επιτυγχάνει έλεγχο του ρυθμού μετάδοσης καθώς επίσης περιορίζει το jitter με θετικά αποτελέσματα για τους πελάτες. Ένα δέντρο min heap υλοποιεί την ουρά χρόνου με δείκτες στις δομές μετάδοσης. Εύκολα διακρίνεται ότι ο εξυπηρετητής μετάδοσης μέσω του δείκτη στην αντίστοιχη υποαίτηση στο κανάλι επικοινωνίας, έχει πρόσβαση στην απαραίτητη πληροφορία (όπως την θέση των δεδομένων στον αποταμιευτή).

Ο XmtServer με δεδομένο αυτήν την χρονική πληροφορία αρχίζει να μεταδίδει από την χρονική του ουρά τα πακέτα εκείνα με προθεσμίες πολύ κοντά στο τρέχων χρονικό σημείο ή ήδη παραβιασμένες. Η διαδικασία αυτή συνεχίζεται μέχρι ο εξυπηρετητής μετάδοσης να διαπιστώσει ότι η προθεσμία ενός πακέτου για μετάδοση είναι «αρκετά» μακριά (εξαρτάται από την ακρίβεια του υλικού). Τότε η εργασία του διακόπτεται και ο XmtServer παραμένει αδρανής περιμένοντας την επόμενη ενεργοποίηση από τον RT-Controller.

Με τον μηχανισμό αυτόν επιτυγχάνουμε την πολύπλεξη πακέτων διαφορετικών πελατών. Όταν όμως η πολύπλεξη αυτή δεν δημιουργεί αρκετές καθυστερήσεις στην αποστολή διαδοχικών πακέτων ενός πελάτη, τότε εμποδίζεται η μετάδοση πακέτων δεδομένων στο δίκτυο, παρά το γεγονός ότι τα δεδομένα είναι διαθέσιμα από τους δίσκους, μέχρι να πλησιάσει αρκετά η προθεσμία αποστολής. Αυτή είναι και η ριζική διαφορά του προτεινόμενου μηχανισμού μετάδοσης από την πολιτική προθεσμιών EDF. Έτσι επιχειρούμε να ελέγξουμε τον ρυθμό μετάδοσης των πακέτων και να αποφύγουμε φαινόμενα εκρηκτικής μετάδοσης και αυξημένου jitter.

Για την επίτευξη του μηχανισμού δρομολόγησης που περιγράφηκε παραπάνω, είναι απαραίτητη μια δομή, η οποία θα υλοποιεί μια ουρά προτεραιότητας με βάση τα χρονικά σημεία που αναπαριστούν την προθεσμία έναρξης της μετάδοσης των πακέτων δεδομένων.

Ο αριθμός των προσπελάσεων στην ουρά αυτή, είναι μεγάλος ιδιαίτερα όταν ο αριθμός των πελατών που εξυπηρετούνται από το σύστημα είναι αυξημένος. Για τον λόγο αυτό πρέπει επιλεχθεί μια αποδοτική δομή αναπαράστασης ουράς προτεραιότητας, όπως είναι ένα δέντρο min heap. Όπως φαίνεται στο Σχήμα 67 οι κόμβοι του δέντρου αποτελούν απλώς δείκτες σε δομές Xrequest, οι οποίες περιέχουν τα χρονικά σημεία των προθεσμιών μετάδοσης. Αναφορά στην απόδοση, σύμφωνα με την βιβλιογραφία, του δέντρου heap γίνεται στον πίνακα που ακολουθεί και εκτιμάται ως ικανοποιητική για την εφαρμογή του στον εξυπηρετητή μετάδοσης.

Περιπτώσεις	Καλύτερη	Μέση	Χειρότερη
Απόδοση εισαγωγής	$O(1)$	$O(\log N)$	$O(\log N)$
Απόδοση διαγραφής	$O(1)$	$O(\log N)$	$O(\log N)$

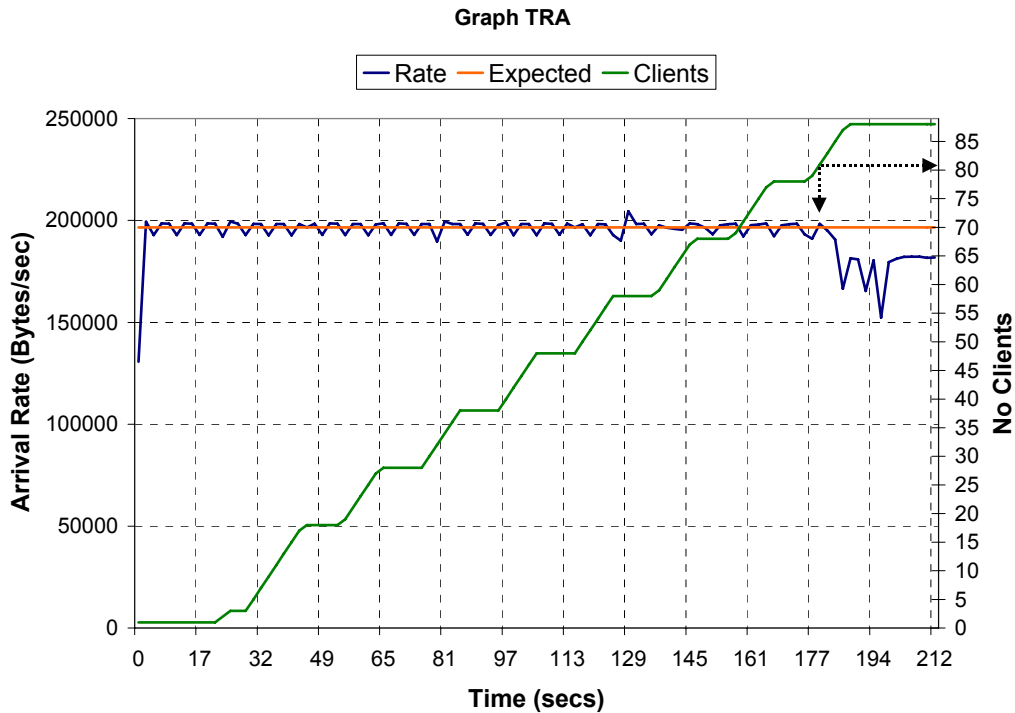
6.4.6 Πειραματική εφαρμογή του μηχανισμού δρομολόγησης στον εξυπηρετητή μετάδοσης

Σε αυτό το πείραμα εφαρμόζεται ο προτεινόμενος μηχανισμός δρομολόγησης στον εξυπηρετητή μετάδοσης, όπως περιγράφηκε στην προηγούμενη ενότητα, για να πάρουμε μετρήσεις στην πλευρά του πελάτη. Στο Σχήμα 68 εμφανίζεται ο ρυθμός άφιξης των δεδομένων στον πελάτη. Σε σύγκριση με τις προηγούμενους τρόπους δρομολόγησης που εφαρμόστηκαν στις υποενότητες 6.4.2, 6.4.3 και 6.4.4, ο προτεινόμενος μηχανισμός δρομολόγησης εμφανίζει ιδιαίτερα θετικά αποτελέσματα.

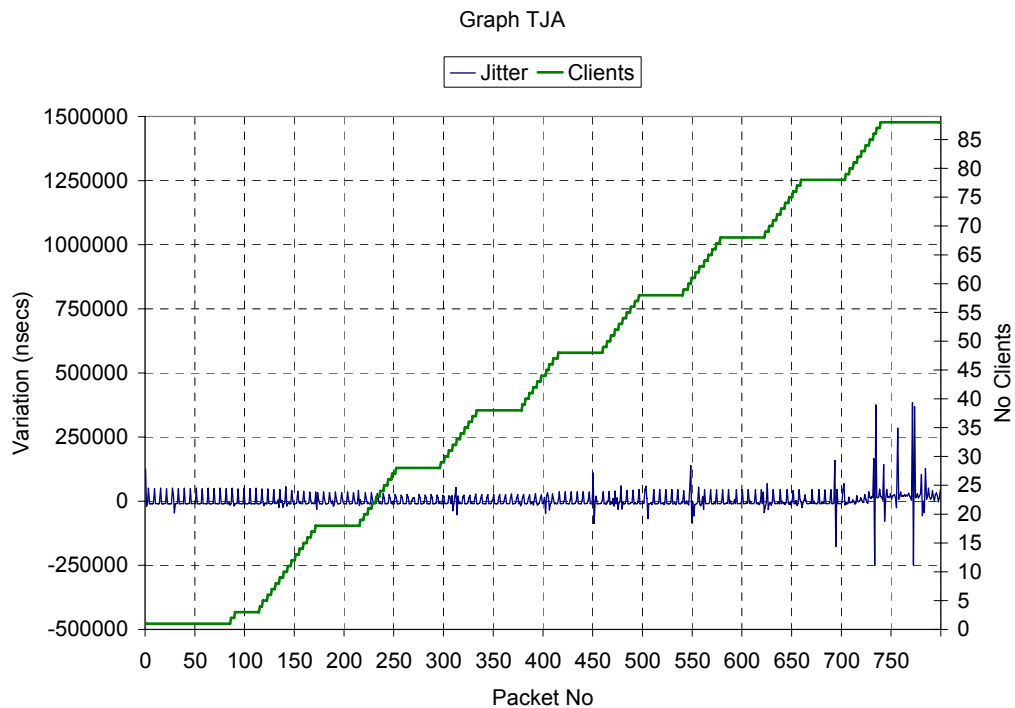
Σε αυτό το πείραμα έχουμε ελαχιστοποίηση της απώλειας πακέτων με άμεσο αποτέλεσμα την διόρθωση του ρυθμού άφιξης δεδομένων στον πελάτη, όταν ο αριθμός των εξυπηρετούμενων πελατών είναι μικρός. Μια περίπτωση που ήταν προβληματική στα προηγούμενα πειράματα. Αλλά και στις περιπτώσεις αύξησης του αριθμού των χρηστών ο ρυθμός άφιξης δεδομένων παραμένει σε ικανοποιητικά επίπεδα.

Χρησιμοποιώντας το διάγραμμα του ρυθμού αποστολής στο Σχήμα 68, προκύπτει ότι ο εξυπηρετητής μετάδοσης μπορεί να ικανοποιήσει την εξυπηρέτηση μέχρι 81 πελατών, καθώς μεγαλύτερη αύξηση στον αριθμό πελατών που εξυπηρετούνται επιφέρει μείωση του ρυθμού άφιξης δεδομένων στον πελάτη, προφανώς λόγω εμφάνισης φαινομένων συμφόρησης στο υποσύστημα μετάδοσης.

Το αποτέλεσμα είναι θετικό, όσον αφορά την απόδοση του μηχανισμού δρομολόγησης, καθώς αποδεικνύεται ότι καταφέρνει να οδηγεί τις ροές μετάδοσης σύμφωνα με τις απαιτήσεις του μέσου και τα χαρακτηριστικά των πελατών. Το σημαντικότερο όμως είναι ότι αυτό συμβαίνει χωρίς ανεπιθύμητες επιπτώσεις στην μέγιστη απόδοση του συστήματος ($81 \times 1.5 \text{Mbps} = 121.5 \text{Mbps}$, μια απόδοση που προσεγγίζει σε μεγάλο βαθμό την μέγιστη 122Mbps, βλ. Σχήμα 35).



Σχήμα 68: Ρυθμός άφιξης δεδομένων στον πελάτη (Μετρήσεις Πειράματος)
 Στο σχήμα φαίνεται ο ρυθμός άφιξης δεδομένων στην πλευρά του πελάτη, χρησιμοποιώντας τον μηχανισμό δρομολόγησης της διατριβής. Είναι εμφανές ότι τα αποτελέσματα είναι αρκετά καλύτερα από τα προηγούμενα πειράματα, τόσο για μικρό όσο και για μεγάλο αριθμό ενεργών χρηστών.



Σχήμα 69 Καθυστέρηση διαδοχικών πακέτων στον πελάτη (Μετρήσεις Πειράματος)
 Διάγραμμα καθυστέρησης διαδοχικών αφίξεων πακέτων στην πλευρά του πελάτη, χρησιμοποιώντας τον μηχανισμό δρομολόγησης της διατριβής. Όπως προκύπτει από τα πειραματικά αποτελέσματα έχουμε ικανοποιητική μείωση του jitter μετάδοσης.

Στο σχήμα Σχήμα 69 καταγράφηκαν οι καθυστερήσεις διαδοχικών πακέτων στον πελάτη. Από την παρατήρηση του σχήματος είναι εμφανής η ριζική βελτίωση της νευρικότητας της αποστολής δεδομένων εφαρμόζοντας στον εξυπηρετητή μετάδοσης τον νέο μηχανισμό δρομολόγησης της μετάδοσης. Αξιοσημείωτο είναι το γεγονός ότι το jitter που παρατηρείται στην πλευρά του πελάτη παραμένει σε πολύ χαμηλά επίπεδα, ακόμα και για μεγάλο αριθμό πελατών, μειώνοντας σημαντικά την πιθανότητα απώλειας πακέτων, λόγω εκρηκτικών μεταδόσεων, σε πελάτες με ευαίσθητες ρυθμίσεις.

Συμπερασματικά μπορούμε να πούμε ότι ο προτεινόμενος μηχανισμός δρομολόγησης επιτυγχάνει σταθερή ροής μετάδοσης και μειώνει σε θεμιτά επίπεδα το jitter κατά την μετάδοση πακέτων για ένα συγκεκριμένο πελάτη.

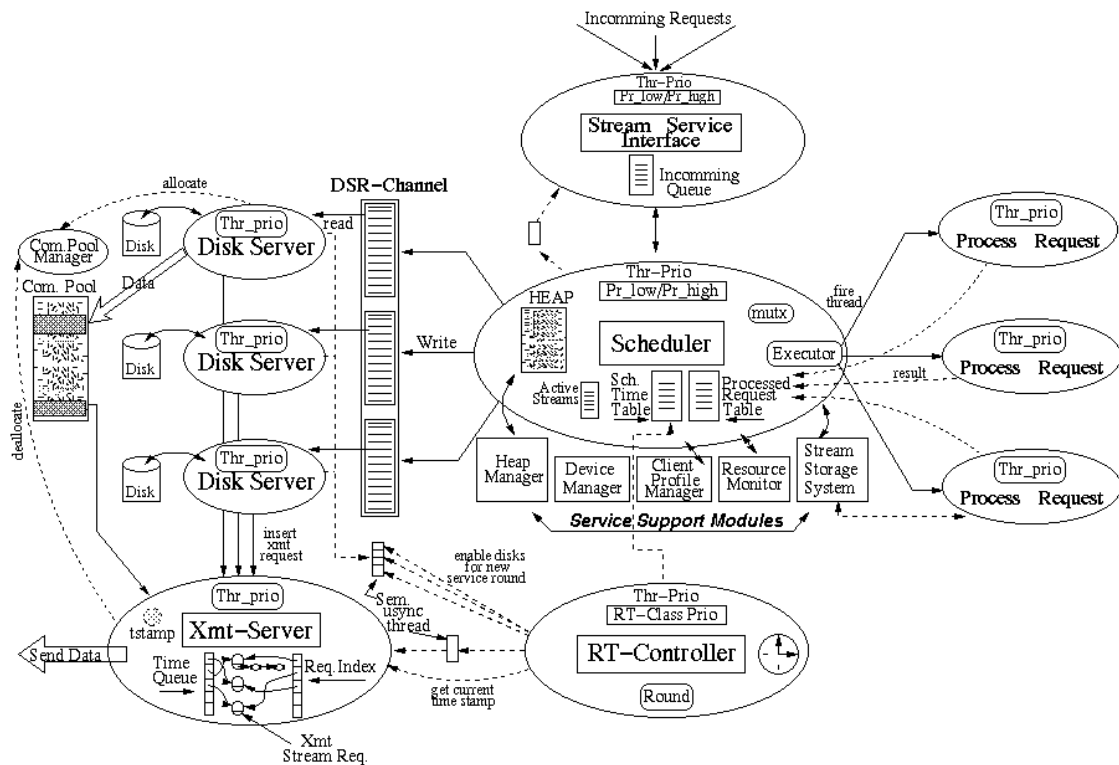
6.5 Συνολική επισκόπηση της υπηρεσίας δεδομένων συνεχόμενης ροής

Στην προηγούμενη ενότητα έγινε μια περιγραφή των επιμέρους τμημάτων που αποτελούν την υπηρεσία αποθήκευσης δεδομένων ροής και της λειτουργικότητά τους. Ο σχεδιασμός και η ανάπτυξη της υπηρεσίας ροής είχαν σαν στόχο την αποδοτική αντιμετώπιση των απαιτητικών (σε πόρους του συστήματος) μέσων συνεχόμενης ροής. Έγινε μια προσπάθεια για την κατάλληλη χρήση τεχνικών ανάπτυξης λογισμικού και για προσεκτική υλοποίηση.

Ο σχεδιασμός αξιοποίησε τα πλεονεκτήματα της πολυνηματικής τεχνολογίας, που ανοίγει δρόμους προς την παράλληλη επεξεργασία. Μέσω του πολυνηματικού σχεδιασμού επιτεύχθηκαν βασικοί στόχοι, όπως η δραστική μείωση του επικοινωνιακού κόστους και του κόστους συγχρονισμού διαφορετικών εργασιών και η δυνατότητα επέκτασης του συστήματος. Η λειτουργία της υπηρεσίας ροής οδηγείται με χρονικούς κύκλους εξυπηρέτησης. Μια πολιτική που συμβαδίζει με την περιοδική φύση των αντικειμένων ροής. Με τον κατάλληλο συγχρονισμό των νημάτων έχουμε σαν αποτέλεσμα την πλήρη αξιοποίηση του επεξεργαστή, αποφεύγοντας περιττές εργασίες να απασχολούν ή και να μονοπωλούν τον επεξεργαστή (π.χ. busy waiting τεχνικές).

Στο Σχήμα 70 παρουσιάζεται η συνολική αρχιτεκτονική της υπηρεσίας ροής. Εμφανίζονται όλα τα επιμέρους νήματα εργασιών και τα τμήματα λογισμικού υποστήριξης. Στο ίδιο επίσης σχήμα αναπαριστάται και ο μηχανισμός δρομολόγησης που επιβάλλεται στο σύστημα.

Στην συνέχεια του κεφαλαίου ακολουθούν διάφορα ενδεικτικά πειράματα που έλαβαν χώρα. Τα πειράματα αυτά έχουν σαν στόχο την επιβεβαίωση της ορθότητας του σχεδιασμού αλλά και την καταγραφή των επιδόσεων της υπηρεσίας ροής. Πολλά βέβαια από αυτά οδήγησαν σε αποφάσεις και αλλαγές στον σχεδιασμό για να έχουμε τα καλύτερα δυνατά αποτελέσματα, ώστε να οδηγηθούμε στο τέλος του κεφαλαίου στον τελικό προτεινόμενο σχεδιασμό για την ανάπτυξη της υπηρεσίας αποθήκευσης δεδομένων ροής



Σχήμα 70: Η αρχιτεκτονική της υπηρεσίας αποθήκευσης δεδομένων ροής. Ολόκληρος ο μηχανισμός δρομολόγησης που αναπτύχθηκε και εφαρμόζεται στην υπηρεσία αποθήκευσης δεδομένων ροής εικονίζεται σε αυτό το σχήμα. Διακρίνονται τα βασικά νήματα ελέγχου όπως το νήμα ζεύξης και επαφής, ο κεντρικός δρομολογητής, ο ελεγκτής πραγματικού χρόνου, οι εξυπηρετητές δίσκου και τέλος ο εξυπηρετητής μετάδοσης. Τα νήματα στα δεξιά του σχήματος είναι προσωρινά νήματα ελέγχου, τα οποία δημιουργούνται για την επεξεργασία κάθε νέας αίτησης. Στο σχήμα εμφανίζονται επίσης ο συγχρονισμός των νημάτων (με χρήση σηματοφόρων και mutex locks), τα υποσυστήματα υποστήριξης, οι προσωρινοί αποταμιευτές και η ροή των δεδομένων. Τέλος στην παραπάνω αρχιτεκτονική μπορεί κανείς να κατανοήσει τον τρόπο αλληλεπίδρασης των νημάτων και την μεταφορά πληροφορίας ανάμεσά τους..

6.6 Απόδοση της υπηρεσίας διαχείρισης δεδομένων ροής

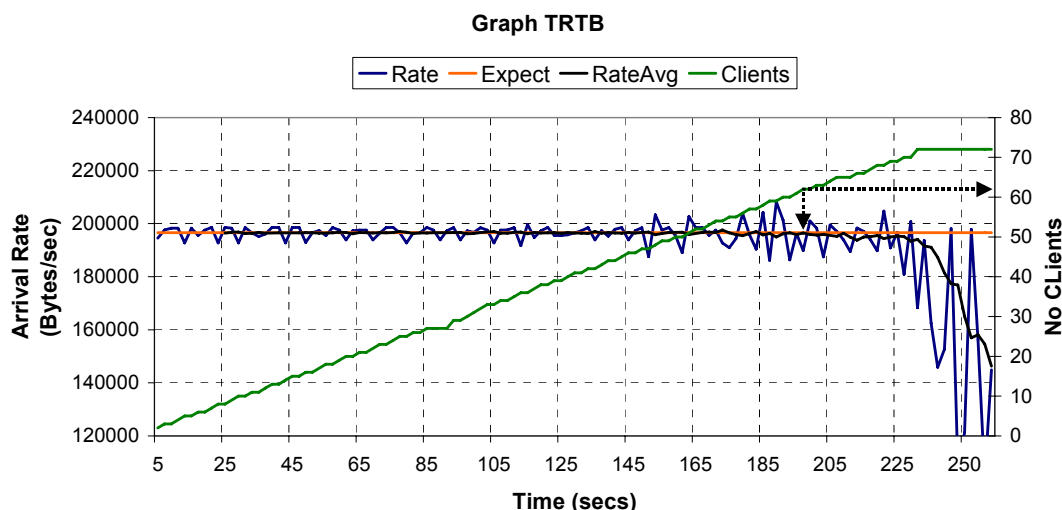
6.6.1 Συνολική συμπεριφορά της υπηρεσίας ροής (Πείραμα)

Σε προηγούμενες ενότητες έλαβαν χώρα διάφορα πειράματα με σκοπό να μελετήσουμε την συμπεριφορά και την αποτελεσματικότητα του μηχανισμού δρομολόγησης του υποσυστήματος μετάδοσης δεδομένων. Τα πειράματα αυτά είχαν εστιαστεί στον εξυπηρετητή μετάδοσης, απομονώνοντας τα υπόλοιπα τμήματα λογισμικού.

Ευνόητο είναι το γεγονός ότι υπάρχει ανάγκη προσεκτικότερης μελέτης του συστήματος σε συνολικό πια επίπεδο, από την οποία θα προκύψουν σημαντικά συμπεράσματα για την υπηρεσία ροής και την αποδοτική συνλειτουργία όλων των επιμέρους μηχανισμών από την ανάκτηση δεδομένων, την αποταμίευσή τους μέχρι την μετάδοσή τους τελικά στους πελάτες.

Έτσι στην συνέχεια αυτής της υποενότητας ακολουθούν πειράματα και λαμβάνονται μετρήσεις στην πλευρά του πελάτη, τρέχοντας το σύνολο της υπηρεσίας διαχείρισης δεδομένων ροής, το οποίο παρουσιάστηκε γραφικά στο Σχήμα 70, απομακρύνοντας

τμήματα λογισμικού εσωτερικά της υπηρεσίας, τα οποία αποσκοπούσαν στην λήψη μετρήσεων σε προηγούμενα πειράματα και πιθανώς να έχουν επιδράσεις στην απόδοση του συστήματος. Στα επόμενα πειράματα η υπηρεσία ροής τρέχει σε ένα υπολογιστή Ultra One εξοπλισμένο με τέσσερις μαγνητικούς δίσκους που είναι προσαρμοσμένοι σε δύο διαύλους SCSI. Στο υλικό του εξυπηρετητή ροής συμπεριλαμβάνεται η κάρτα δικτύου με την οποία διασυνδέεται το σύστημα στο τοπικό ATM δίκτυο. Στο σύστημα εισάγονται σταδιακά (ανά 3sec) νέοι πελάτες μέχρι τους 80, ενώ ταυτόχρονα παρακολουθούμε την ποιότητα των υπηρεσιών που προσφέρει ο εξυπηρετητής στον πρώτο πελάτη. Ο πελάτης τρέχει σε μια Sparc20 μηχανή.



Σχήμα 71: Ρυθμός άφιξης δεδομένων στον πελάτη (Μετρήσεις Πειράματος)

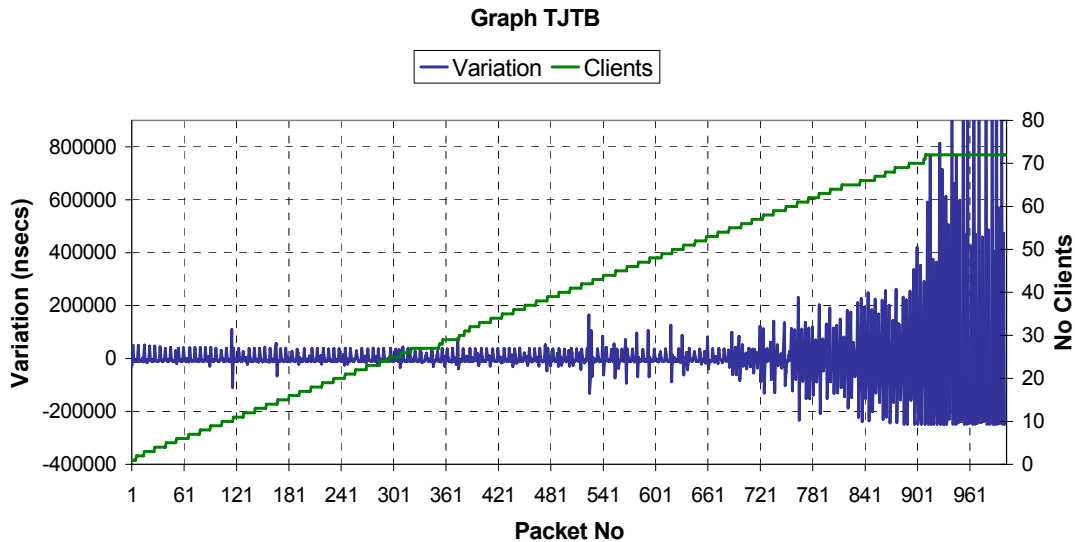
Στο σχήμα καταγράφονται ο ρυθμός άφιξης δεδομένων στον πρώτο πελάτη, όταν η υπηρεσία διαχείρισης δεδομένων ροής τρέχει στο σύνολό της. Στο διάγραμμα διακρίνεται η γραμμή του επιθυμητού ρυθμού άφιξης, ο αριθμός των ενεργών πελατών σε κάθε χρονική στιγμή και η μέση τιμή των μετρήσεων που λαμβάνονται. Τα αποτελέσματα δείχνουν ότι η υπηρεσία ροής στην τρέχουσα της μορφή μπορεί να ικανοποιήσει ικανοποιητικά μέχρι 63 πελάτες.

Τα αποτελέσματα του πειράματος είναι ικανοποιητικά και δείχνουν ότι ο σχεδιασμός του συστήματος είναι επιτυχής. Στο Σχήμα 71 εικονίζεται ο ρυθμός άφιξης δεδομένων στον πρώτο πελάτη σε συνδυασμό με τον αριθμό των ενεργών πελατών. Από το γράφημα το ρυθμού άφιξης παρατηρούμε ότι η υπηρεσία ροής παρέχει δεδομένα στον πελάτη με τον κατάλληλο ρυθμό, χωρίς να επηρεάζεται από την συνεχή εισαγωγή νέων πελατών. Μετά όμως από τους 50 περίπου πελάτες, αρχίζουν να παρουσιάζονται διακυμάνσεις στον καταγεγραμμένο ρυθμό, ενώ μετά τους 65 πελάτες παρατηρείται αισθητή μείωση του ρυθμού με τον οποίο αποστέλλει δεδομένα ο εξυπηρετητής ροής.

Οι μετρήσεις του πειράματος δείχνουν επίσης ότι παρά την συνλειτουργία όλων των υποσυστημάτων το jitter μετάδοσης διατηρείται σε ικανοποιητικά χαμηλά επίπεδα.

Παρά όμως το γεγονός ότι οι μετρήσεις αναδεικνύουν την καλή συμπεριφορά του συστήματος κατά την εξυπηρέτηση των πελατών, το σύστημα εμφανίζει μειωμένη απόδοση. Τα πειράματα που έλαβαν χώρα στα υποσυστήματα της υπηρεσίας ροής έδειξαν μεγαλύτερη απόδοση. Ο εξυπηρετητής μετάδοσης απέδειξε (Σχήμα 68) ότι μπορεί να υποστηρίξει 81 πελάτες στο τρέχων σύστημα, ενώ το υποσύστημα μετάδοσης με τις

τρέχουσες ρυθμίσεις ήταν ικανό να αντιμετωπίσει φόρτο εργασίας μεγαλύτερο από αυτό των 63 πελατών (κεφάλαιο 5).



Σχήμα 72 Καθυστέρηση άφιξης διαδοχικών πακέτων στον πελάτη (Μετρήσεις Πειράματος)
Στο σχήμα εικονίζονται οι αποκλίσεις της χρονικής καθυστέρησης άφιξης διαδοχικών πακέτων από την αναμενόμενη τιμή που καταγράφονται στον πρώτο πελάτη, όταν η υπηρεσία διαχείρισης δεδομένων ροής τρέχει στο σύνολό της με τις τρέχουσες ρυθμίσεις. Το θετικό συμπέρασμα των μετρήσεων είναι ότι οι διαφοροποιήσεις στους χρόνους άφιξης είναι περιορισμένες σε χαμηλά επίπεδα, όταν ο αριθμός των πελατών που εξυπηρετούνται δεν είναι μεγάλος.

Συνεπώς, μπορούμε να συμπεράνουμε ότι η ταυτόχρονη λειτουργία των μηχανισμών δρομολόγησης της ανάκτησης και μετάδοσης δεδομένων περιορίζει την απόδοση του συστήματος.

Γίνεται επομένως εύκολα κατανοητό ότι χρειάζεται ένας αποδοτικός μηχανισμός της πρόσβασης των εργασιών στον επεξεργαστή, καθώς ανεξάρτητα οι μηχανισμοί των υποσυστημάτων αποδίδουν καλά, ενώ συνδυασμένοι εμφανίζουν καθυστερήσεις και μικρότερες επιδόσεις. Το πρόβλημα οφείλεται πιθανόν στο ότι επιμέρους εργασίες απασχολούν τον επεξεργαστή σε χρονικά σημεία στα οποία η εκτέλεση άλλων εργασιών είναι κρίσιμη, δημιουργώντας αθέμιτες καθυστερήσεις.

Στις επόμενες ενότητες γίνεται μια προσπάθεια για καλύτερο συντονισμό του συνολικού μηχανισμού δρομολόγησης της υπηρεσίας ροής. Εισάγουμε την χρήση προτεραιοτήτων στα νήματα που αναδεικνύουν τις κρίσιμες εργασίες, ενώ για ακόμα καλύτερα αποτελέσματα υιοθετούμε προσεγγίσεις πραγματικού χρόνου, ώστε να αναπτύξουμε ένα σύστημα με σταθερή συμπεριφορά και ικανοποιητική ακρίβεια λειτουργίας, ικανό να παρέχει εγγυήσεις εξυπηρέτησης πραγματικού χρόνου.

6.7 Δρομολόγηση νημάτων με χρήση προτεραιοτήτων

Στο σύστημα τα διαφορετικά νήματα δρομολογούνται με βάση προτεραιότητες που δυναμικά αναθέτονται στα νήματα κατά τη διάρκεια της εκτέλεσης. Χρησιμοποιούνται δύο

βασικά επίπεδα τιμών προτεραιοτήτων για κάθε νήμα. Μια τιμή αντιπροσωπεύει την χαμηλή προτεραιότητα και υποδεικνύει στο λειτουργικό σύστημα ότι η εργασία του νήματος δεν είναι τόσο κρίσιμη, ενώ μια υψηλή τιμή προτεραιότητας υποδηλώνει ότι η σημασία εκτέλεσης του νήματος την τρέχουσα χρονική στιγμή είναι αυξημένη. Η εφαρμογή προτεραιοτήτων στα νήματα απλά σημαίνει ότι ένα νήμα με υψηλή προτεραιότητα θα προηγηθεί κατά την πολύπλεξη των νημάτων πάνω σε μία LWP, ώστε να τύχει επεξεργασίας πιο σύντομα.

Βάσει του σχεδιασμού το νήμα RT-Controller πρέπει να έχει την μεγαλύτερη προτεραιότητα εκτέλεσης (RTCrrio), εφόσον οδηγεί χρονικά τους κύκλους εξυπηρέτησης. Ο RT-Controller όπως έχει αναφερθεί, παίρνει ένα χρονικό σημείο από το σύστημα, ελέγχει αν πρέπει να αρχίσει η επεξεργασία του επόμενου κύκλου και στην συνέχεια καθίσταται αδρανής μετά την κατάλληλη κλήση υπηρεσίας για μικρό χρονικό διάστημα. Αυτό το διάστημα εκτελούνται διεργασίες άλλων νημάτων. Μόλις το διάστημα της αδράνειας παρέλθει, ο RT-Controller αποκτά κατάσταση που υποδηλώνει ότι είναι έτοιμο για εκτέλεση (runnable). Η υψηλότερη προτεραιότητα που του έχει ανατεθεί επιτρέπει στο νήμα να ευνοηθεί στην δέσμευση διαθέσιμης LWP έναντι όλων των άλλων νημάτων κερδίζοντας με αυτόν τον τρόπο ακρίβεια στον χρονικό συντονισμό των εργασιών της υπηρεσίας.

Σταθερές και υψηλές προτεραιότητες επίσης φέρουν τα νήματα Dservers και XmtServer (DSprio, XSprio) που οδηγούν την ανάκτηση και την μετάδοση των δεδομένων αντίστοιχα. Κι' αυτό γιατί τόσο τα νήματα εξυπηρέτησης δίσκων όσο και το νήμα μετάδοσης πρέπει να δρομολογούνται για εκτέλεση γρήγορα ώστε να ικανοποιούνται οι απαιτήσεις πραγματικού χρόνου στην ανάκτηση και την μετάδοση δεδομένων όπως επιβάλλουν τα αντικείμενα ροής.

Το νήμα εισόδου του συστήματος (SSI) τον περισσότερο χρόνο παραμένει φραγμένο (blocked) περιμένοντας νέες αιτήσεις. Όταν η ουρά αναμονής είναι άδεια ή περιέχει λίγες αιτήσεις φέρει την υψηλή προτεραιότητα (SSImax-prio), ώστε να δεχθεί γρήγορα την σύνδεση της νέας αίτησης και να βελτιωθεί ο χρόνος απόκρισης του συστήματος. Αν όμως στην ουρά αναμονής συσσωρευτεί μεγάλος αριθμός αιτήσεων, γεγονός που υποδηλώνει ότι το σύστημα είναι πολύ φορτωμένο, η προτεραιότητα του SSI νήματος χαμηλώνει (SSImin-prio), ευνοώντας με αυτόν τον τρόπο την επεξεργασία των υπαρχόντων αντικειμένων ροής που ήδη εξυπηρετούνται.

Το νήμα του κεντρικού δρομολογητή φέρει επίσης υψηλή προτεραιότητα (SCHmax-prio) όταν υπάρχουν αιτήσεις στην ουρά αναμονής για να δρομολογήσει την επεξεργασία τους ή όταν υπάρχουν αποτελέσματα της επεξεργασίας των αιτήσεων. Διαφορετικά όταν βρίσκεται στην κατάσταση όπου περιοδικά ελέγχει τις ουρές και τους πίνακες για νέα στοιχεία αποκτά την χαμηλή προτεραιότητα (SCHmin-prio), ώστε να αποδεσμεύει πόρους για πιο σημαντικές εργασίες. Ο κεντρικός δρομολογητής επίσης καθορίζει και τις προτεραιότητες (PRprio) των νημάτων επεξεργασίας της κάθε αίτησης ανάλογα με στοιχεία που φέρει η αίτηση. Στο υπάρχων σύστημα εφαρμόζεται ο περιορισμός RTprio>DSprio>XSprio>SSImax-prio>SCHmax-prio, PRprio.

6.8 Προγραμματισμός πραγματικού χρόνου

Για να μπορέσουμε να βελτιώσουμε την απόδοση και την ποιότητα των υπηρεσιών του εξυπηρετητή αποθήκευσης, είναι αναγκαίο να εφαρμόσουμε τις βασικές αρχές ανάπτυξης εφαρμογών πραγματικού χρόνου. Η απόκριση πραγματικού χρόνου μπορεί να εγγυηθεί όταν εξασφαλίζονται ορισμένες καταστάσεις. Ο σχεδιασμός όμως πρέπει να είναι πολύ προσεκτικός γιατί συχνά βασικά λάθη οδηγούν σε δραματική μείωση της απόδοσης ή και κατάρρευση ολόκληρου του υπολογιστικού συστήματος.

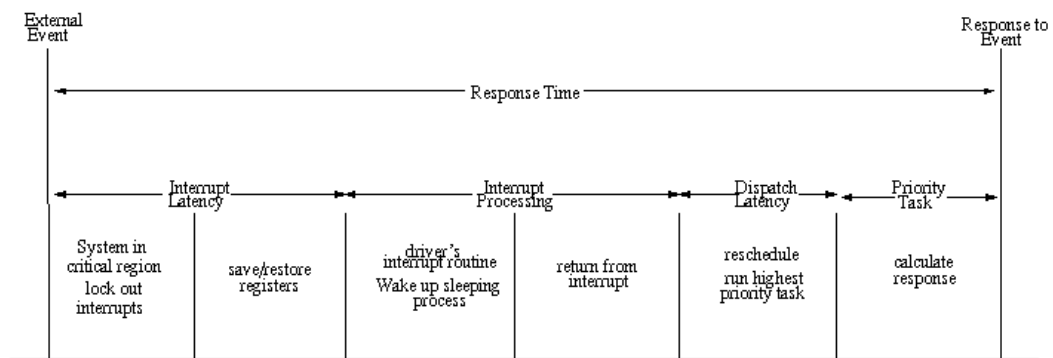
Στην προσπάθεια προσέγγισης της απαραίτητης συμπεριφοράς πραγματικού χρόνου μιας εφαρμογής, θα πρέπει:

1. Οι διεργασίες πρέπει να τύχουν ειδικής δρομολόγησης από το λειτουργικό σύστημα.
2. Κρίσιμοι αποταμιευτές πρέπει να κλειδώνονται στην κύρια μνήμη, ώστε να είναι εγγυημένη η ταχεία προσπέλασή τους.
3. Η εφαρμογή πρέπει να προέρχεται από στατική σύνδεση είτε οι δυναμικές βιβλιοθήκες πρέπει να έχουν φορτωθεί πριν την έναρξη της εκτέλεσης.

Η δρομολόγηση πραγματικού χρόνου επιβάλλει στην διεργασία να αντιδρά σε διάφορα γεγονότα μέσα σε φραγμένο χρονικό διάστημα. Τα λειτουργικά συστήματα που επιτρέπουν την ανάπτυξη εφαρμογών πραγματικού χρόνου παρέχουν μια ειδική κλάση δρομολόγησης που διαφοροποιεί την παραδοσιακή αντιμετώπιση των διεργασιών. Θεωρητικά όταν μια διεργασία τρέχει στην κλάση πραγματικού χρόνου με την μεγαλύτερη προτεραιότητα, δέχεται την εξυπηρέτηση του επεξεργαστή με εγγυήσεις μέγιστης καθυστέρησης από την στιγμή που θα ζητήσει επεξεργασία και δεσμεύει τον επεξεργαστή για όσο διάστημα εξακολουθεί να παραμένει η διεργασία με την υψηλότερη προτεραιότητα εκτέλεσης. Συνεπώς μια διεργασία πραγματικού χρόνου μπορεί να χάσει ή να μην μπορεί να αποκτήσει τον έλεγχο του επεξεργαστή μόνο σε περιπτώσεις ειδικών συμβάντων. Σε αυτά τα συμβάντα συμπεριλαμβάνονται εξωτερικές ενέργειες (π.χ. διακοπές), έλλειψη διαθέσιμων πόρων, περιπτώσεις συγχρονισμού και τέλος διακοπή εκτέλεσης από άλλη διεργασία υψηλότερης προτεραιότητας.

6.8.1 Δρομολόγηση πραγματικού χρόνου

Η δημιουργία ενός περιβάλλοντος πραγματικού χρόνου απαιτεί την δυνατότητα μερικές διεργασίες να μπορούν να αντιδρούν σε εξωτερικές αιτίες σε φραγμένο χρόνο. Το γεγονός αυτό συχνά έρχεται σε αντίθεση με τον πυρήνα των λειτουργικών συστημάτων, ο οποίος έχει σχεδιαστεί με κύρια μέριμνα την δίκαιη κατανομή των πόρων επεξεργασίας σε όλες τις διεργασίες. Στην συνέχεια του κειμένου θα γίνει αναφορά στα ιδιαίτερα χαρακτηριστικά του λειτουργικού συστήματος Solaris και τις δυνατότητες που παρέχει, όσον αφορά τις τεχνικές προγραμματισμού πραγματικού χρόνου.



Σχήμα 73:Χρόνος απόκρισης εφαρμογής σε εξωτερικό γεγονός

Το σχήμα δείχνει τον χρόνο απόκρισης μιας διεργασίας ή οποία είναι ανενεργή (sleeping) και περιμένει κάποιο εξωτερικό γεγονός. Ο χρόνος απόκρισης καθορίζεται από τον χρόνο αντίδρασης του συστήματος (Interrupt latency), τον χρόνο επεξεργασίας της διακοπής (interrupt processing), τον χρόνο επαναδρομολόγησης και τον χρόνο επεξεργασίας της απόκρισης από την διεργασία.

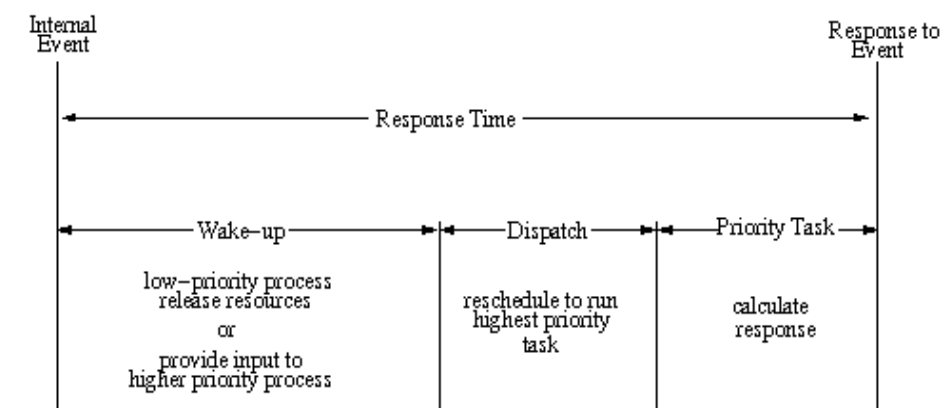
Το σημαντικότερο στοιχείο για την δρομολόγηση εφαρμογών πραγματικού χρόνου, είναι η παροχή διαφόρων κλάσεων δρομολόγησης, οι οποίες υποστηρίζονται από τον πυρήνα. Οι κλάσεις ορίζονται σε: πραγματικού χρόνου (RT), συστήματος (Sys) και διαμοιρασμού χρόνου (TS). Η συνηθισμένη και επικρατέστερη πολιτική δρομολόγησης -η κλάση διαμοιρασμού χρόνου- δεν είναι κατάλληλη για εφαρμογές που χρειάζονται αποκρίσεις πραγματικού χρόνου, καθώς διέπεται από την γενική αρχή της ισότιμης μεταχείρισης όλων των διεργασιών και υιοθετεί ένα περιορισμένο σχήμα προτεραιοτήτων. Ιδιαίτερα οι διεργασίες πραγματικού χρόνου έχουν ανάγκη από την ύπαρξη απόλυτων προτεραιοτήτων, οι οποίες δε αλλάζουν παρά μόνο μετά από ενέργεια της ίδιας της εφαρμογής. Ο τεχνικός όρος καθυστέρηση αποστολής εκτέλεσης (dispatch latency) περιγράφει το χρονικό διάστημα που χρειάζεται το σύστημα, προκειμένου να ανταποκριθεί στην αίτηση της διεργασίας για έναρξη επεξεργασίας. Με την προϋπόθεση ότι έχουμε ένα δρομολογητή πυρήνα ο οποίος σχεδιάστηκε για να αντιμετωπίζει τις προτεραιότητες των εφαρμογών, είναι δυνατή η ανάπτυξη εφαρμογών με φραγμένη καθυστέρηση αποστολής εκτέλεσης (bounded dispatch latency).

Στο Σχήμα 73 διακρίνεται ο χρόνος που μεσολαβεί για να απαντήσει η εφαρμογή σε ένα εξωτερικό γεγονός. Ο συνολικός χρόνος απόκρισης συνθέτεται από τον χρόνο απόκρισης στην διακοπή, την καθυστέρηση αποστολής εκτέλεσης και τον χρόνο που χρειάζεται η ίδια η εφαρμογή να καθορίσει την απόκρισή της.

Ο χρόνος απόκρισης στην διακοπή περιλαμβάνει την καθυστέρηση διακοπής, όπως προκύπτει από το σύστημα και τον χρόνο επεξεργασίας της διακοπής (από το λογισμικό οδηγού - driver). Κατά την διάρκεια επεξεργασίας μιας διακοπής, η ρουτίνα χειρισμού διακοπής του οδηγού συσκευής ενεργοποιεί την διεργασία με την υψηλή προτεραιότητα και επιστρέφει όταν ολοκληρώσει τις απαραίτητες εργασίες. Το σύστημα ανακαλύπτει ότι μια διεργασία με υψηλότερη προτεραιότητα (από αυτήν της διεργασίας που διακόπηκε) είναι έτοιμη για επεξεργασία και αναλαμβάνει να αποστείλει για εκτέλεση την διεργασία αυτή. Ο απαραίτητος χρόνος για να ολοκληρωθεί η εναλλαγή εκτέλεσης (context-switch) υπολογίζεται στην καθυστέρηση αποστολής εκτέλεσης.

Από την άλλη πλευρά στο σχήμα φαίνεται ο χρόνος απόκρισης μιας εφαρμογής σε εσωτερικό γεγονός. Η καθυστέρηση απόκρισης σε εσωτερικό συμβάν, αναπαριστά το χρονικό διάστημα που απαιτείται για μια διεργασία να ενεργοποιήσει μια άλλη υψηλότερης προτεραιότητας και για το σύστημα να αποστείλει για εκτέλεση την κρίσιμη διεργασία.

Τα δύο αυτά παραδείγματα καθυστέρησης απόκρισης έχουν ιδιαίτερη σημασία στον σχεδιασμό της υπηρεσίας ροής. Η πρώτη περίπτωση που περιγράφηκε, αναφέρεται στην καθυστέρηση απόκρισης σε εξωτερικό γεγονός και σχετίζεται άμεσα με τους εξυπηρετητές δίσκου και μετάδοσης δεδομένων. Όπως στο παράδειγμα έτσι και ένας εξυπηρετητής δίσκου μετά την κλήση της υπηρεσίας του συστήματος για ανάκτηση δεδομένων από μια συσκευή, μεταβαίνει σε μια κατάσταση αναμονής (sleep). Όταν το τελευταίο σήμα διακοπής σταλεί στο σύστημα (δηλαδή όλα τα δεδομένα έχουν ανακτηθεί), επιτρέπεται στην ρουτίνα επεξεργασίας της διακοπής του οδηγού να ενεργοποιήσει την αδρανή διεργασία (μετά από την καθυστέρηση διακοπής βλ. Σχήμα 73) και στο σύστημα να δρομολογήσει την εκτέλεση της διεργασίας του εξυπηρετητή δίσκου.



Σχήμα 74: Χρόνος απόκρισης εφαρμογής σε εσωτερικό γεγονός

Όταν συμβεί ένα εσωτερικό γεγονός που υποδηλώνει ότι μια διεργασία με υψηλή προτεραιότητα πρέπει να συνεχίσει την εκτέλεση, ο χρόνος απόκρισης της διεργασίας περιλαμβάνει τον χρόνο ενεργοποίησης της διεργασίας, τον χρόνο επαναδρομολόγησης και τον χρόνο επεξεργασίας της απόκρισης από την διεργασία.

Αναδεικνύεται συνεπώς πόσο σημαντικό είναι το γεγονός παροχής εγγυήσεων φραγμένης καθυστέρησης απόκρισης, καθώς αυξάνει την απόδοση των εξυπηρετητών δίσκου και μετάδοσης και εξασφαλίζει την ροή δεδομένων χωρίς παραβιάσεις των προθεσμιών πραγματικού χρόνου.

Εξίσου σημαντική είναι προφανώς και η δυνατότητα εγγύησης πάνω ορίου στους χρόνους απόκρισης σε εσωτερικά γεγονότα, εφόσον αυτό μεταφράζεται σε φραγμένο χρόνο αποστολής εκτέλεσης εργασιών που αποδεικνύονται χρονικά κρίσιμες. Έτσι δίνεται η δυνατότητα εφαρμογής ενός μοντέλου δρομολόγησης με προτεραιότητες που θα διαμορφώνονται δυναμικά, ώστε να επιτυγχάνεται η πολύπλεξη πολλών εργασιών πάνω στον επεξεργαστή, χωρίς να παραβιάζονται οι προθεσμίες επεξεργασίας των χρονικά ευαίσθητων εργασιών.

6.8.2 Κλάσεις δρομολόγησης

Όπως αναφέρθηκε στην προηγούμενη ενότητα το λειτουργικό σύστημα Solaris υποστηρίζει την έννοια των κλάσεων δρομολόγησης. Οι βασικές κλάσεις που ορίζονται είναι: η κλάση πραγματικού χρόνου (RT), συστήματος (SYS) και διαμοιρασμού χρόνου (TS) (σχήμα). Κάθε κλάση καθορίζει μια μοναδική πολιτική δρομολόγησης για τις διεργασίες που ανήκουν σε αυτές. Ο δρομολογητής του πυρήνα αποστέλλει πρώτη για εκτέλεση την διεργασία με την υψηλότερη προτεραιότητα. Εξ' ορισμού οι διεργασίες που ανήκουν στην κλάση πραγματικού χρόνου προηγούνται αυτών που ανήκουν στις άλλες δύο.

Την μεγαλύτερη προτεραιότητα έχουν τα σήματα διακοπής του υλικού (hardware interrupts) και δεν μπορούν να ελεγχθούν από το λογισμικό. Οι ρουτίνες επεξεργασίας των διακοπών υλικού ενεργοποιούνται και εκτελούνται αυτόματα, ανεξάρτητα από τις προτεραιότητες των τρέχουσων διεργασιών. Οι διεργασίες της κλάσης πραγματικού χρόνου έχουν την υψηλότερη προτεραιότητα λογισμικού και φέρουν δύο τιμές, την τιμή της προτεραιότητας και μια τιμή χρονικού κβάντα.

Scheduling Order	Global Priority	Process Priority	Scheduling Class
First	159	59	Real Time
.	.	.	
.	.	.	
.	.	.	
.	.	.	
.	.	0	System
.	99	59	
.	.	.	
.	.	.	
.	.	.	
.	.	0	Time Sharing
.	59	59	
.	.	.	
.	.	.	
.	.	.	
Last	0	0	

Σχήμα 75: Οι κλάσεις δρομολόγησης διεργασιών

Οι βασικές κλάσεις δρομολόγησης διεργασιών εμφανίζονται στο παραπάνω σχήμα. Κάθε κλάση έχει στην διάθεσή της ένα εύρος τιμών προτεραιοτήτων, οι οποίες μεταφράζονται από τον πυρήνα σε απόλυτες προτεραιότητες, ώστε να καθορισθεί η σειρά εκτέλεσης που θα επιβάλλει.

Εφόσον μια διεργασία πραγματικού χρόνου είναι έτοιμη για εκτέλεση καμία άλλη είτε στην κλάση συστήματος είτε στην κλάση διαμοιρασμού χρόνου δεν μπορεί να εκτελεστεί. Η δρομολόγηση σταθερών προτεραιοτήτων επιτρέπει σε κρίσιμες διεργασίες να εκτελούνται μέχρι την ολοκλήρωσή τους. Αυτές οι προτεραιότητες δεν αλλάζουν παρά μόνο μετά από εντολή της ίδιας της εφαρμογής. Η τιμή του χρονικού κβάντα είτε είναι άπειρη είτε περιορισμένη κληρονομείται από τη διεργασία-πατέρα. Μια διεργασία με περιορισμένη τιμή χρονικού κβάντα επεξεργάζεται μέχρι να εκπνεύσει το χρονικό κβάντα ή η διεργασία τερματιστεί, ή ανασταλεί η εκτέλεση για λόγους συγχρονισμού, ή μια άλλη διεργασία αποκτήσει μεγαλύτερη προτεραιότητα και είναι έτοιμη για εκτέλεση.

Η κλάση του συστήματος υπάρχει για την επεξεργασία ειδικών εργασιών. Δεν επιτρέπεται η αλλαγή κλάσης σε μια διεργασία που ήδη τρέχει στην κλάση συστήματος, αλλά και ούτε επιτρέπεται σε μια διεργασία χρήστη να τρέξει στην κλάση αυτή.

Η κλάση διαμοιρασμού χρόνου φέρει τις μικρότερες προτεραιότητες. Οι διεργασίες που ανήκουν στην κλάση αυτή δρομολογούνται δυναμικά και η εκτέλεσή τους εναλλάσσεται (ακολουθώντας ένα κυκλικό σχήμα) μετά από χρονικά διαστήματα μερικών εκατοντάδων χιλιοστών δευτερολέπτου. Η πολιτική διαμοιρασμού προσπαθεί να προσφέρει ίσες ευκαιρίες επεξεργασίας σε όλες τις διεργασίες, λαμβάνοντας υπόψη τα χρονικά διαστήματα εκτέλεσης, την ιστορία των διεργασιών και την καλύτερη αξιοποίηση του επεξεργαστή.

Συμπερασματικά μπορούμε να πούμε ότι διαφορετικοί αλγόριθμοι εφαρμόζονται σε κάθε κλάση δρομολόγησης. Ρουτίνες εξαρτώμενες από τις κλάσεις καλούνται συχνά από τον πυρήνα για την λήψη αποφάσεων σχετικά με την δρομολόγηση των διεργασιών. Ο ίδιος όμως ο πυρήνας είναι ανεξάρτητος κλάσεων και εξυπηρετεί την διεργασία με την υψηλότερη προτεραιότητα από την ουρά που διατηρεί.

Κάθε κλάση έχει στην διάθεσή της ένα σύνολο από επίπεδα προτεραιοτήτων, που εφαρμόζονται στις διεργασίες της κλάσης. Μια αντιστοιχία - εξαρτώμενη από την κλάση - απεικονίζει τις προτεραιότητες της κλάσης σε απόλυτες (global) προτεραιότητες (Σχήμα 75. Αξίζει να σημειωθεί ότι δεν είναι απαραίτητο οι αντιστοιχίες με τις απόλυτες προτεραιότητες να αρχίζουν από το μηδέν, ούτε και να είναι συνεχόμενες.

6.8.3 Αποστολή και εναλλαγή εκτέλεσης (Dispathing & Preemption)

Η ουρά αποστολής εκτέλεσης (dispath queue) είναι μια γραμμική διπλά διασυνδεδεμένη λίστα από διεργασίες με την ίδια απόλυτη προτεραιότητα. Κάθε διεργασία φέρει πληροφορία σχετική με την κλάση στην οποία ανήκει. Όταν μια διεργασία αποστέλλεται για εκτέλεση, το περιβάλλον της διεργασίας (context) ανακτάται στην μνήμη, μαζί με την πληροφορία διαχείρισης της μνήμης, τους καταχωρητές της και την στοίβα της. Από εκείνη την στιγμή και μετά αρχίζει η εκτέλεση της διεργασίας.

Όταν μια διεργασία με υψηλότερη προτεραιότητα καταστεί έτοιμη για αποστολή εκτέλεσης, ο πυρήνας διακόπτει την τρέχουσα επεξεργασία και επιβάλλει την διαδικασία εναλλαγής εκτέλεσης. Μια διεργασία μπορεί να διακοπεί σε οποιαδήποτε στιγμή διαπιστώσει ο πυρήνας ότι άλλη διεργασία προηγείται. Για παράδειγμα αν υποθέσουμε ότι μια διεργασία Α εκτελεί μια εντολή ανάκτησης δεδομένων από μια συσκευή αποθήκευσης, τότε η διεργασία αυτή μεταβαίνει στην ανενεργή κατάσταση (sleep) από τον πυρήνα. Στην συνέχεια ο πυρήνα βρίσκει μια άλλη διεργασία Β χαμηλότερης προτεραιότητας και την αποστέλλει για εκτέλεση. Κάποτε η συσκευή αποθήκευσης ολοκληρώνει την εργασία και στέλνει σήμα διακοπής στον οδηγό, ο οποίος δέχεται άμεσα επεξεργασία διακόπτοντας την διεργασία Β. Ο οδηγός της συσκευής καθιστά την διεργασία Α ενεργή και επιστρέφει. Ο πυρήνας σε αυτό το σημείο, αντί να συνεχίσει την επεξεργασία της διεργασίας Β, την σταματά και επιλέγει για εκτέλεση την ενεργή πλέον διεργασία Α. Ανάλογα φαινόμενα συμβαίνουν όταν πολλές διεργασίες ανταγωνίζονται πόρους του πυρήνα και μια διεργασία χαμηλής προτεραιότητας ελευθερώνει ένα πόρο, που είναι απαραίτητος για μια άλλη -

υψηλότερης προτεραιότητας- διεργασία. Όταν αυτά τα φαινόμενα διαρκούν για αρκετά μεγάλο διάστημα χαρακτηρίζονται ως περιπτώσεις αναστροφής προτεραιότητας στον πυρήνα (kernel priority inversion) και προκαλούνται κυρίως από μηχανισμούς συγχρονισμού των διεργασιών.

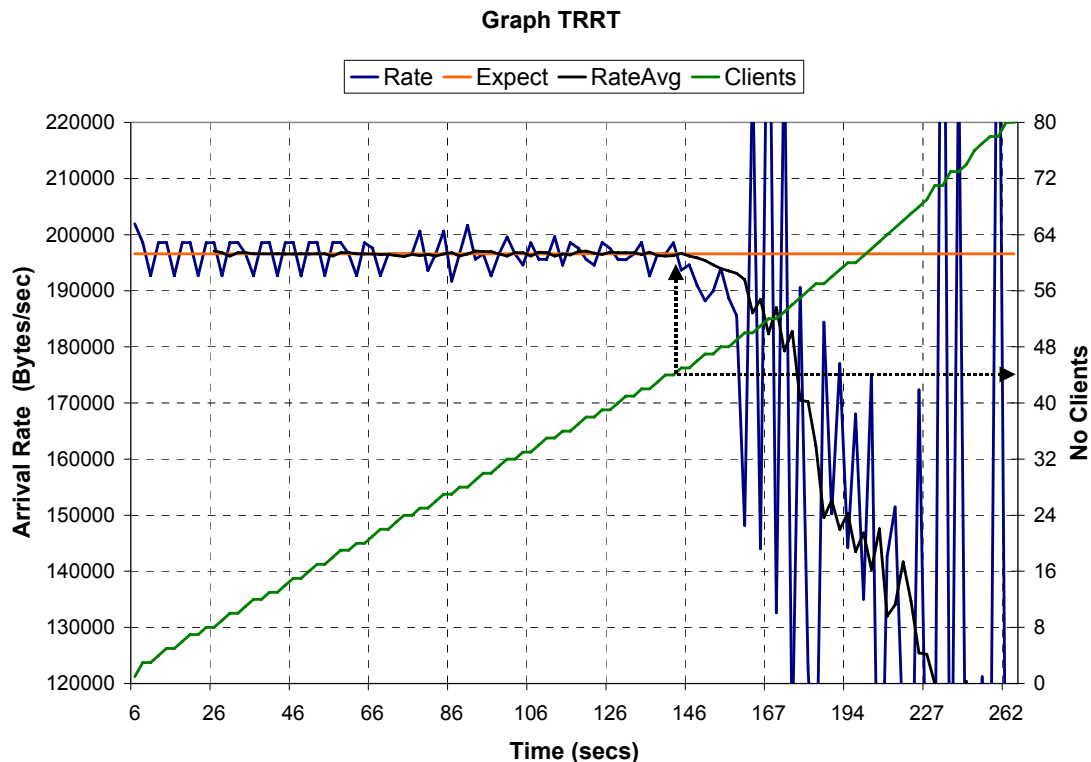
6.8.4 Κλείδωμα μνήμης

Το κλείδωμα μνήμης είναι ένα από τα πιο σημαντικά ζητήματα για τις εφαρμογές πραγματικού χρόνου. Σε ένα περιβάλλον πραγματικού χρόνου μια διεργασία πρέπει να είναι ικανή να εγγυηθεί την αποφυγή καθυστερήσεων που οφείλονται σε μηχανισμούς αλλαγής σελίδων (paging-swapping) στην εικονική μνήμη (virtual memory). Ειδικότερα στο λειτουργικό σύστημα Solaris, η διατήρηση μιας διεργασίας στην κύρια μνήμη, εξαρτάται από την τρέχουσα κατάστασή της, από το συνολικό ποσό διαθέσιμης φυσικής μνήμης, τον αριθμό των ενεργών διεργασιών και τις απαιτήσεις αυτών των διεργασιών σε ποσά μνήμης. Η πολιτική αυτή είναι κατάλληλη για περιβάλλοντα που εφαρμόζεται η πολιτική διαμοιρασμού χρόνου εκτέλεσης, αλλά δεν μπορεί να γίνει αποδεκτή σε περιβάλλοντα πραγματικού χρόνου τις περισσότερες φορές. Κι' αυτό γιατί μια διεργασία η οποία επιβάλλεται να έχει απόκριση πραγματικού χρόνου, δεν πρέπει να γνωρίζει καθυστερήσεις προσπέλασης της μνήμης ή καθυστερήσεις στην αποστολή εκτέλεσης (λόγω swapping).

6.9 Αλλαγή της κλάσης δρομολόγησης (Πείραμα)

Σε αυτό το πείραμα επιβάλλουμε να τρέξει η διεργασία της υπηρεσίας διαχείρισης δεδομένων ροής στην κλάση πραγματικού χρόνου του λειτουργικού συστήματος και να τύχει της ανάλογης δρομολόγησης. Η ανάθεση της υπηρεσίας ροής στην κλάση δρομολόγησης πραγματικού χρόνου, σημαίνει ότι αυτή η διεργασία έχει εξ' ορισμού την μεγαλύτερη προτεραιότητα εκτέλεσης από οποιαδήποτε άλλη διεργασία στο σύστημα. Η εκτέλεσή της μπορεί να ανασταλεί μόνο για λόγους έλλειψης απαραίτητων πόρων ή λόγω της ανάγκης προσπέλασης μέσω δευτερεύουσας μνήμης. Η ενέργεια αυτή γίνεται στα πλαίσια της προσπάθειας να παράσχουμε εγγυήσεις πραγματικού χρόνου στην εξυπηρέτηση των αιτήσεων ανάκτησης αντικειμένων συνεχόμενης ροής.

Όπως φαίνεται από τα αποτελέσματα, η αλλαγή αυτή έχει καταστροφικές συνέπειες, ιδιαίτερα όταν ο φόρτος του συστήματος αρχίζει να αυξάνει. Γίνεται αντιληπτό ότι με αυτόν τον τρόπο, εργασίες μη σημαντικές μονοπωλούν τον επεξεργαστή και πιθανόν να καθυστερούν την εκτέλεση ζωτικών λειτουργιών όπως είναι η ανάκτηση δεδομένων από τα μέσα αποθήκευσης. Αναδεικνύεται επομένως η ανάγκη για μια καλύτερη προσέγγιση στο ζήτημα της δρομολόγησης με χρήση των υπηρεσιών πραγματικού χρόνου.



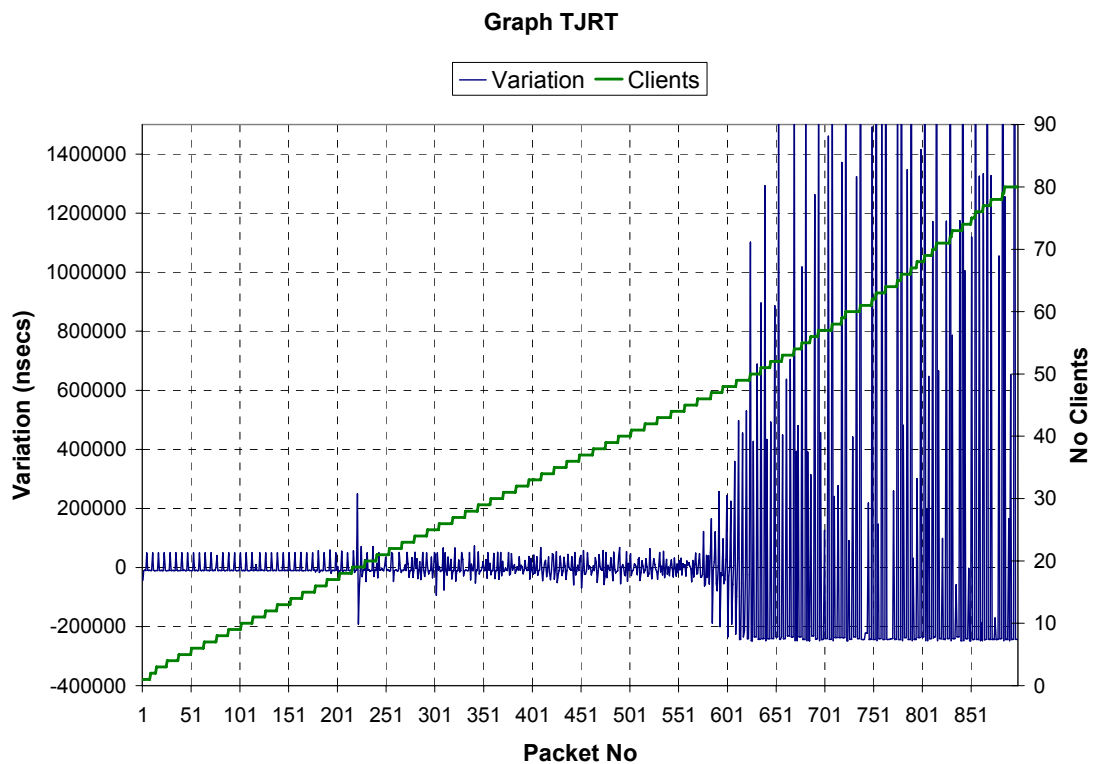
Σχήμα 76: Ρυθμός άφιξης δεδομένων στον πελάτη (Μετρήσεις Πειράματος)
 Ο ρυθμός άφιξης δεδομένων στην πλευρά του πελάτη, επιβάλλοντας την κλάση δρομολόγησης πραγματικού χρόνου σε ολόκληρη την διεργασία της υπηρεσίας δεδομένων συνεχόμενης ροής. Τα αποτελέσματα δείχνουν ότι για μεγάλο αριθμό πελατών το σύστημα εμφανίζει έντονη αστάθεια.

Στο Σχήμα 76 καταγράφονται οι μετρήσεις από την επανάληψη του τελευταίου πειράματος της ενότητας 6.6, με αλλαγή της κλάσης δρομολόγησης της διεργασίας της υπηρεσίας ροής. Είναι εμφανές ότι όταν ο αριθμός των χρηστών είναι μικρός και συνεπώς οι ανάγκες επεξεργαστικής ισχύος μειωμένες, η ποιότητα εξυπηρέτησης στην πλευρά του πελάτη είναι ικανοποιητική. Με την αισθητή όμως αύξηση του αριθμού των χρηστών που ζητούν εξυπηρέτηση, και την εντατικοποίηση χρήσης του επεξεργαστικού πόρου, το σύστημα εμφανίζει έντονα σημεία αστάθειας προκαλώντας μεγάλες διακυμάνσεις στον ρυθμό μετάδοσης δεδομένων στον πελάτη.

Συμπερασματικά, η ενέργεια τις αλλαγής κλάσης δρομολόγησης εμφανίζεται άστοχη, καθώς οδηγεί σε μείωση της ικανότητας του εξυπηρετητή που ικανοποιεί τις ανάγκες μόλις 44 πελατών.

Όμοια και στο Σχήμα 77 που περιέχει τα αποτελέσματα των μετρήσεων για τις καθυστερήσεις άφιξης διαδοχικών πακέτων στον πελάτη, προκύπτουν αρνητικά συμπεράσματα. Όταν ο φόρτος εργασίας του συστήματος είναι μικρός, η διαφοροποίηση στους χρόνους άφιξης των πακέτων είναι αρκετά περιορισμένη. Με την συνεχή άφιξη των εισερχόμενων αιτήσεων εξυπηρέτησης, αυξάνεται και η νευρικότητα που παρατηρείται στην μετάδοση του αντικειμένου ροής στον πελάτη. Το γεγονός αυτό έχει σαν αποτέλεσμα την αύξηση του jitter που καταγράφεται σε υψηλά επίπεδα και την γενική αποδιοργάνωση του όλου μηχανισμού δρομολόγησης. Οι μεγάλες διακυμάνσεις που παρατηρούνται στους

χρόνους άφιξης διαδοχικών πακέτων και κυμαίνονται από σχεδόν μηδενικά επίπεδα σε χρόνους τετραπλάσιους από τους θεμιτούς ($>1\text{sec}$), φωτογραφίζουν την δυσλειτουργία του μηχανισμού δρομολόγησης και την έλλειψη αποδοτικού συντονισμού των εργασιών μετάδοσης και ανάκτησης δεδομένων. Έτσι έχουμε σαν αποτέλεσμα την εμφάνιση σημαντικών καθυστερήσεων μετάδοσης δεδομένων, με άμεση απόρροια την παραβίαση πολλών προθεσμιών. Όταν δοθεί η ευκαιρία μεταδίδονται όλα τα δεδομένα με παραβιασμένες προθεσμίες, λόγω του εφαρμοζόμενου αλγορίθμου μετάδοσης και έτσι εμφανίζονται φαινόμενα εκρηκτικής μετάδοσης και κατά συνέπεια αυξημένου ρυθμού μετάδοσης στην πλευρά του πελάτη.



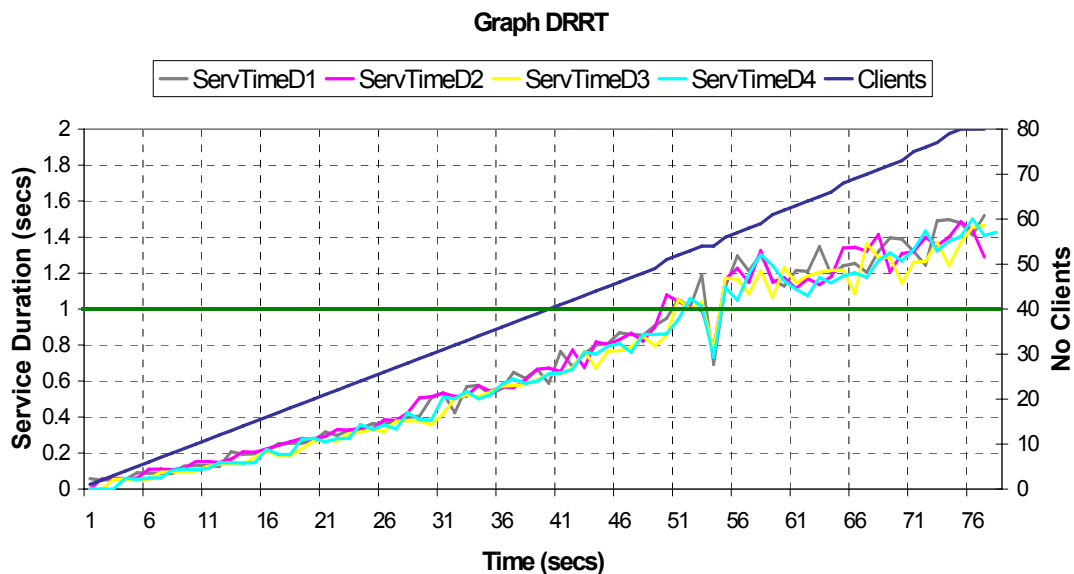
Σχήμα 77: Καθυστερήση διαδοχικών πακέτων στον πελάτη (Πείραμα 9.2)

Στο σχήμα εικονίζονται οι καθυστερήσεις άφιξης διαδοχικών πακέτων, επιβάλλοντας στην διεργασία της υπηρεσίας δεδομένων συνεχόμενης ροής την κλάση δρομολόγησης πραγματικού χρόνου. Με την αύξηση του φόρτου εργασίας εμφανίζονται μη αναστρέψιμα φαινόμενα αυξημένου jitter μετάδοσης.

Ανάλογη μη ικανοποιητική εικόνα εμφανίζεται επίσης στην συμπεριφορά του υποσυστήματος ανάκτησης δεδομένων της υπηρεσίας, τα αποτελέσματα του οποίου παρουσιάζονται στο Σχήμα 78.

Το τρέχων πείραμα βοηθάει στην κατανόηση της δυσκολίας ανάπτυξης εφαρμογών με συμπεριφορά πραγματικού χρόνου. Αποδεικνύεται ότι δεν είναι αρκετό να αυξάνουμε την προτεραιότητα σε διεργασίες που περιέχουν χρονικά ευαίσθητες εργασίες, ώστε να έχουμε εγγυήσεις εμπρόθεσμης επεξεργασίας. Επίσης οι επεξεργασίες πραγματικού χρόνου μπορεί να έχει σημαντικές επιδράσεις στην λειτουργία άλλων εφαρμογών που τρέχουν στην κλάση διαμοιρασμού χρόνου, καθώς μπορεί να αποτρέψει τις τελευταίες από την εκτέλεση για μεγάλα χρονικά διαστήματα. Ιδιαίτερη προσοχή επίσης χρειάζεται στην διάρκεια και το

ποσοστό της υπολογιστικής ισχύος που καταναλώνεται από διεργασίες πραγματικού χρόνου, καθώς μπορεί να εμποδιστεί και η ομαλή λειτουργία ακόμα και των απαραίτητων διεργασιών πυρήνα.



Σχήμα 78: Καθυστέρηση ανάκτησης διαδοχικών τμημάτων δεδομένων (Πείραμα 9)
Το παραπάνω διάγραμμα παρουσιάζει τα αποτελέσματα των απόδοσης στο υποσύστημα ανάκτησης, ενώ ταυτόχρονα προβάλλεται ο αριθμός των ενεργών πελατών που εξυπηρετούνται σε κάθε χρονική στιγμή από το σύστημα. Από το σχήμα προκύπτει ότι η υπηρεσία ροής που τρέχει στην κλάση πραγματικού χρόνου γρήγορα εμφανίζει φαινόμενα μειωμένης αξιοποίησης των μέσων αποθήκευσης.

6.10 Σχεδιασμός δρομολόγησης πραγματικού χρόνου

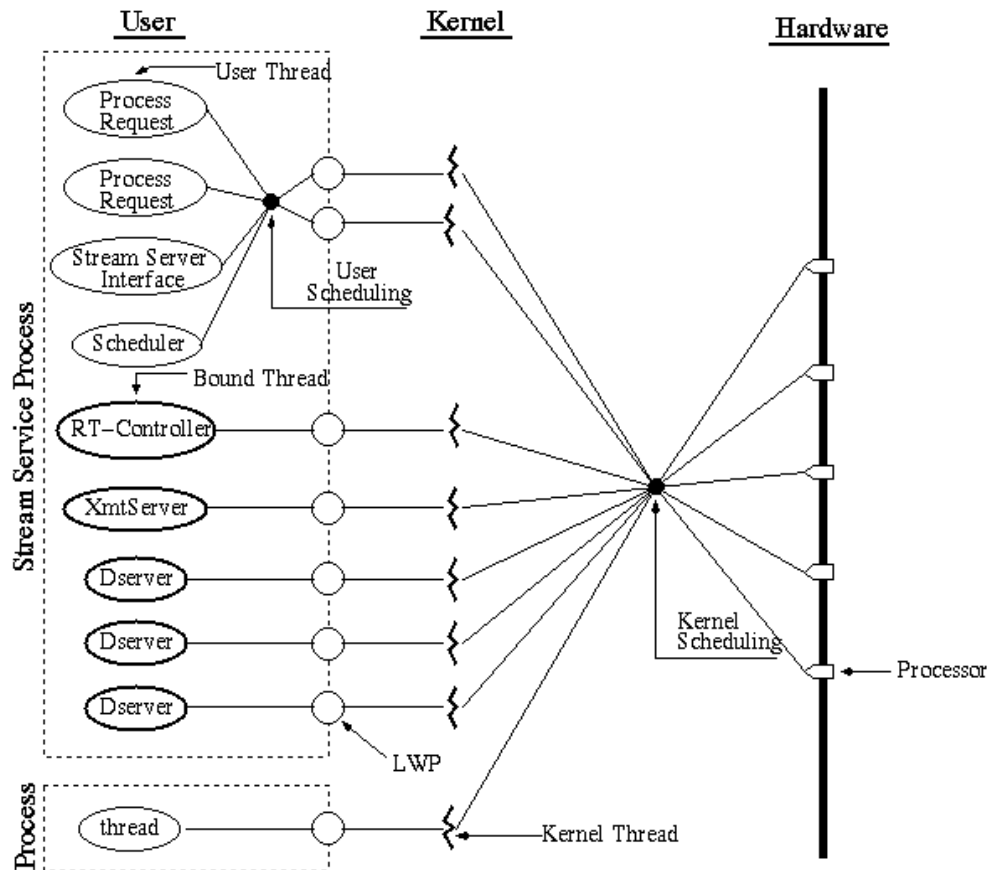
Από το προηγούμενο πείραμα γίνεται εμφανές ότι χρειάζεται ένας καλύτερος τρόπος δρομολόγησης για να ανταποκριθούμε στις απαιτήσεις πραγματικού χρόνου των δεδομένων ροής. Η βασική ιδέα, η οποία οδήγησε τον σχεδιασμό της υπηρεσίας ροής, είναι τα κρίσιμα κομμάτια λογισμικού να δρομολογούνται διαφορετικά από τον ίδιο τον πυρήνα του λειτουργικού συστήματος και όχι σε επίπεδο εφαρμογής, γεγονός που θα περιόριζε την ευελιξία δρομολόγησης στην πολιτική διαμοιρασμού του χρόνου εκτέλεσης

Τα κρίσιμα νήματα στην υπηρεσία ροής είναι ο ελεγκτής πραγματικού χρόνου, που διατηρεί και καθορίζει τους κύκλους εξυπηρέτησης, οι εξυπηρετητές δίσκου, που αναλαμβάνουν την ανάκτηση των δεδομένων από τα αποθηκευτικά μέσα και ο εξυπηρετητής μετάδοσης, ο οποίος είναι υπεύθυνος για τη αποστολή των δεδομένων. Με βάση τα παραπάνω, για να τύχουν τα νήματα αυτά δρομολόγησης πραγματικού χρόνου, σχεδιάστηκαν ως δεμένα νήματα (bounded threads), δηλαδή όπως έχει ήδη περιγραφεί, δεσμεύουν μία LWP πυρήνα το καθένα για αποκλειστική εκτέλεση. Κάθε μια από αυτές τις LWP αναθέτονται στην κλάση δρομολόγησης πραγματικού χρόνου και φέρουν σταθερές προθεσμίες οριζόμενες από την ίδια την εφαρμογή. Με αυτόν τον τρόπο επιτυγχάνουμε ορισμένα νήματα της υπηρεσίας ροής να δεχθούν διαφορετική πολιτική δρομολόγησης

αποστολής εκτέλεσης, εφόσον για τα νήματα επιπέδου χρήστη δεν επιτρέπεται άλλη πολιτική δρομολόγησης εκτός από αυτή του διαμοιρασμού του χρόνου εκτέλεσης.

Συνεπώς τα τρία αυτά είδη νημάτων (RT-Controller, Xmt-Server, Dservers) πρέπει να τύχουν δρομολόγησης σε επίπεδο πυρήνα στην κλάση πραγματικού χρόνου. Με τον κατάλληλο μηχανισμό συγχρονισμού με την χρήση σημαφόρων και μηχανισμών αποκλειστικής εκτέλεσης, αποφεύγεται η μονοπώληση του επεξεργαστή από τα νήματα αυτά, αντίθετα ο μηχανισμός αναγκάζει σε εναλλαγή εκτέλεσης (preemption) άλλες διεργασίες όταν η επεξεργασία κάποιου από τα νήματα αυτά είναι κρίσιμη. Το σημαντικότερο σε αυτή την περίπτωση είναι ότι η αποστολής εκτέλεσης του κρίσιμου νήματος γίνεται με εγγυήσεις φραγμένης καθυστέρησης, κάτι που οδηγεί στην ικανοποίηση των προθεσμιών πραγματικού χρόνου.

Στο παρακάτω σχήμα (Σχήμα 79) φαίνεται γραφικά η νέα προσέγγιση του σχεδιασμού λαμβάνοντας υπόψη όσα αναφέρθηκαν προηγουμένως σε αυτή την υποενότητα. Στο αριστερό μέρος του σχήματος εικονίζονται τα βασικά νήματα ελέγχου που συντελούν την υπηρεσία αποθήκευσης δεδομένων ροής. Το παράδειγμα περιλαμβάνει δύο διεργασίες, οι οποίες αναπαριστώνται με τα διακεκομμένα παραλληλόγραμμα, την υπηρεσία ροής και μια άλλη τυχαία συντρέχουσα διεργασία του συστήματος που περιλαμβάνει ένα μόνο νήμα εκτέλεσης. Τα νήματα που εικονίζονται με έντονη γραμμή αποτελούν «δεμένα» νήματα, ενώ τα υπόλοιπα είναι νήματα επιπέδου χρήστη. Γίνονται εύκολα κατανοητά τα δύο σημεία δρομολόγησης όπως προκύπτουν από το σχήμα. Τα μη-κρίσιμα νήματα εργασιών αντιστοιχίζονται σε νήματα επιπέδου χρήστη και η δρομολόγηση τους λαμβάνει χώρα σε επίπεδο εφαρμογής βάσει των υπάρχουσων προτεραιοτήτων. Η δρομολόγηση σε αυτό το σημείο οδηγεί την πολύπλεξη των νημάτων αυτών στις διαθέσιμες LWP του πυρήνα. Αντίθετα κρίσιμες εργασίες, ως προς την χρονική τους εκτέλεση, δεσμεύουν αποκλειστικά μια LWP για εκτέλεση και αποκτούν το προνόμιο να τρέχουν στην κλάση πραγματικού χρόνου. Με αυτό τον τρόπο η δρομολόγηση της εκτέλεσης, ή με άλλα λόγια η δρομολόγηση της πρόσβασης στους επεξεργαστές, μεταξύ αυτών των νημάτων αλλά και εμμέσως μεταξύ των υπολοίπων της ίδιας ή άλλων διεργασιών αναλαμβάνεται από τον δρομολογητή του πυρήνα, με την αξιοποίηση της πληροφορίας των κλάσεων δρομολόγησης



Σχήμα 79: Η Δρομολόγηση των νημάτων της υπηρεσίας δεδομένων ροής. Οι οντότητες που αναπαριστώνται με ελλείψεις στο σχήμα αποτελούν τα νήματα της υπηρεσίας ροής. Οι κύκλοι αναπαριστούν τις διεργασίες μικρού βάρους οι οποίες αντιστοιχούν σε ένα νήμα του πυρήνα. Στα νήματα εφαρμόζονται τεχνικές συγχρονισμού για τον έλεγχο της λειτουργίας τους. Όσα νήματα (RT-Controller, Dservers, Xmt-Server) χρειάζονται εγγυήσεις απόκρισης πραγματικού χρόνου, «δένονται» αποκλειστικά με μία LWP και αναθέτονται στην κλάση δρομολόγησης πραγματικού χρόνου με την κατάλληλη προτεραιότητα. Στο σχήμα εμφανίζεται η διαφοροποίηση στον τρόπο δρομολόγησης των νημάτων. Αρκετά νήματα δρομολογούνται σε επίπεδο χρήστη (TS), ενώ τα κρίσιμα νήματα έχουν την εύνοια να τύχουν ειδικής δρομολόγησης από τον πυρήνα.

6.11 Εφαρμογή του μηχανισμού δρομολόγησης πραγματικού χρόνου της υπηρεσίας ροής

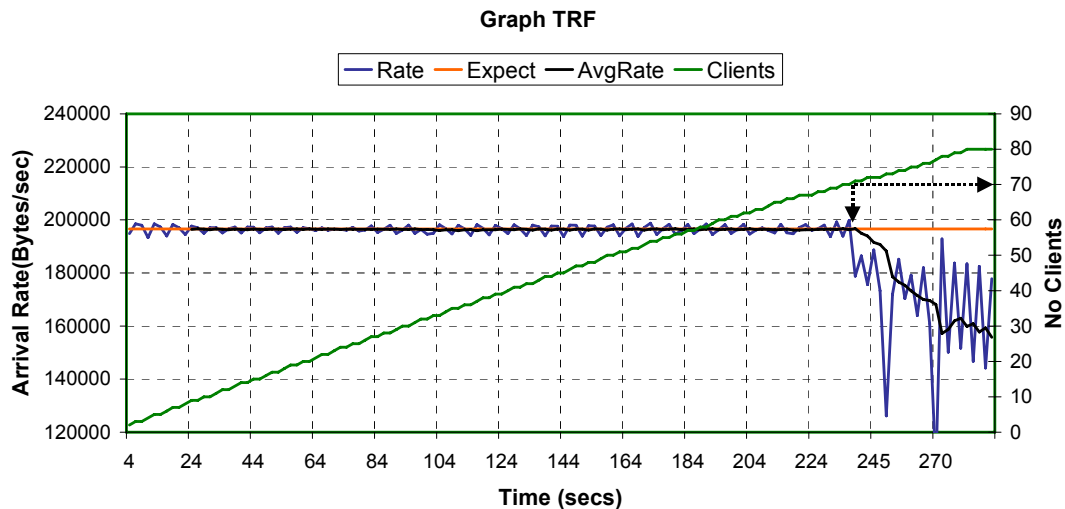
6.11.1 Πειραματική εφαρμογή

Επαναλαμβάνουμε το πείραμα της ενότητας 6.6 με τον προτεινόμενο μηχανισμό δρομολόγησης στην υπηρεσία διαχείρισης δεδομένων ροής. Από τα αποτελέσματα των μετρήσεων αναδεικνύονται εύστοχες οι προηγούμενες αλλαγές στην μηχανισμό δρομολόγησης. Στο Σχήμα 80 καταγράφονται τα αποτελέσματα των μετρήσεων για τον ρυθμό άφιξης των δεδομένων στην πλευρά του πελάτη.

Παρατηρούμε ότι τα αποτελέσματα παράγουν μια ομαλή γραφική παράσταση, που σημαίνει ότι δεν υπάρχουν ενοχλητικές διακυμάνσεις στον ρυθμό άφιξης των δεδομένων στον πελάτη, ένα γεγονός που προσφέρει ποιότητα εξυπηρέτησης και ελαχιστοποίηση των περιπτώσεων διακοπών κατά την αναπαραγωγή των αντικειμένων ροής. Το σημαντικότερο

σε αυτήν την περίπτωση είναι ότι παρατηρούμε αύξηση του αριθμού των πελατών που εξυπηρετούνται ικανοποιητικά (~70).

Ανάλογα θετικά αποτελέσματα όπως είναι φυσικό, έχουμε από την καταγραφή των καθυστερήσεων άφιξης διαδοχικών πακέτων στον πελάτη. Στο Σχήμα 81 παρατηρούμε ότι η εφαρμογή του προτεινόμενου μηχανισμού δρομολόγησης, επιτυγχάνει τον περιορισμό των φαινομένων νευρικότητας κατά την μετάδοση των δεδομένων. Έτσι οι διαφοροποιήσεις κυμαίνονται σε χαμηλά μη σημαντικά επίπεδα, ενώ εκμηδενίζονται εντελώς φαινόμενα εκρηκτικής μετάδοσης.

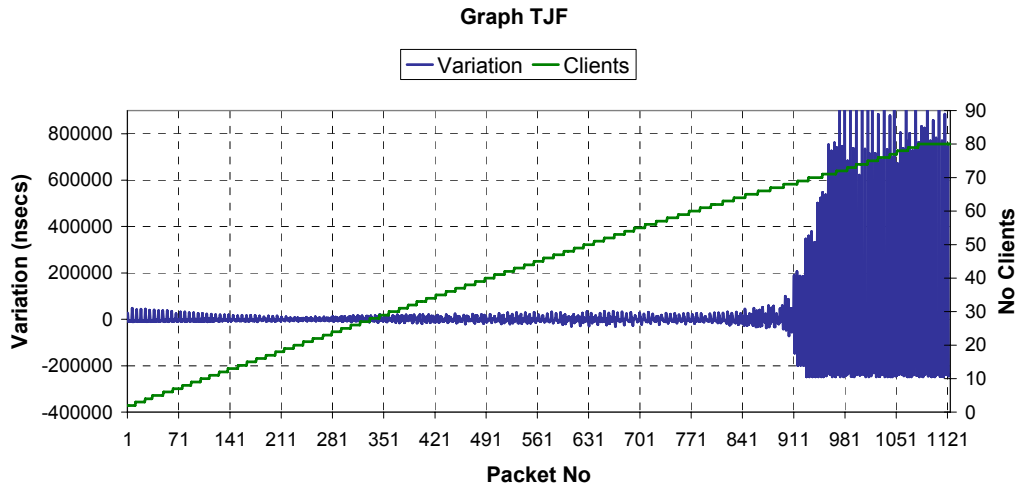


Σχήμα 80: Ρυθμός άφιξης δεδομένων στον πελάτη (Μετρήσεις Πειράματος)

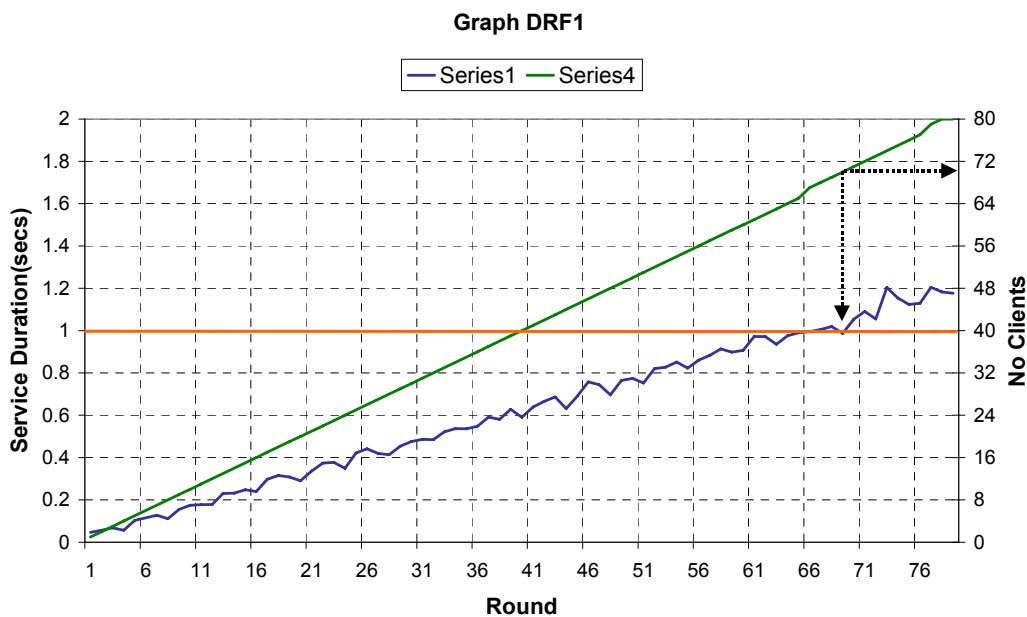
Στο σχήμα εικονίζεται ο ρυθμός άφιξης δεδομένων στην πλευρά του πελάτη, χρησιμοποιώντας τον προτεινόμενο μηχανισμό δρομολόγησης της διατριβής. Τα αποτελέσματα είναι πολύ θετικά καθώς έχουμε σταθερό ρυθμό μετάδοσης δεδομένων ανεξάρτητα από τον φόρτο εργασίας του εξυπηρετητή. Αποτέλεσμα της σωστής εφαρμογής της πολυνηματικής τεχνολογίας και των τεχνικών ανάπτυξης εφαρμογών πραγματικού χρόνου.

Οι μετρήσεις είναι ιδιαίτερα ενθαρρυντικές καθώς αποδεικνύουν ότι αναπτύχθηκε ένα σύστημα ικανό να παρέχει δεδομένα πολυμέσων με σταθερή ποιότητα σε μεγάλο αριθμό χρηστών, ανεξάρτητα από το φόρτο εργασίας που δέχεται.

Θεμιτό είναι σε αυτό το σημείο να συγκρίνουμε τα αποτελέσματα του πρώτου πειράματος στην ενότητα 6.6. Στο πρώτο πείραμα προσπαθούμε να καταγράψουμε την συμπεριφορά του πολυνηματικού μηχανισμού δρομολόγησης, όταν όλα τα νήματα εφαρμόζονται στο επίπεδο χρήστη, χωρίς προτεραιότητες. Είναι εμφανές ότι ο ολοκληρωμένος μηχανισμός δρομολόγησης (προτεραιότητες και δρομολόγηση πραγματικού χρόνου) έχει να παρουσιάσει μεγαλύτερη σταθερότητα και καλύτερη απόδοση.



Σχήμα 81: Καθυστέρηση διαδοχικών πακέτων στον πελάτη (Μετρήσεις Πειράματος)
Καταγραφή των καθυστερήσεων διαδοχικών πακέτων στην πλευρά του πελάτη, χρησιμοποιώντας τον προτεινόμενο μηχανισμό δρομολόγησης της διατριβής, με χρήση πολυνηματικής τεχνολογίας και υπηρεσιών πραγματικού χρόνου. Τα αποτελέσματα αυτά επιβεβαιώνουν την ορθότητα του σχεδιασμού του δρομολογητή μετάδοσης που οδηγεί σε ικανοποιητικό έλεγχο του jitter μετάδοσης.



Σχήμα 82: Καθυστέρηση διαδοχικών τμημάτων δεδομένων (Μετρήσεις Πειράματος)
Το παραπάνω διάγραμμα παρουσιάζει τα αποτελέσματα των μετρήσεων για ένα τυχαίο δίσκο, σχετικά με τον χρόνο εξυπηρέτησης των υποαιτήσεων ανάκτησης ενός κύκλου, ενώ ταυτόχρονα προβάλλεται ο αριθμός των ενεργών πελατών που εξυπηρετούνται σε κάθε χρονική στιγμή από το σύστημα. Από το σχήμα προκύπτει ότι ο δίσκος, με τις τρέχουσες ρυθμίσεις του συστήματος, μπορεί να εξυπηρετήσει ικανοποιητικά την αναπαραγωγή 70 περίπου αντικειμένων ροής.

Ιδιαίτερη σημασία έχει η παρακολούθηση της απόδοσης του υποσυστήματος ανάκτησης με την εφαρμογή του νέου μηχανισμού δρομολόγησης πραγματικού χρόνου. Έτσι επαναλαμβάνουμε το πείραμα, καταγράφοντας αυτήν την φορά μετρήσεις στους εξυπηρετητές δίσκους. Τα αποτελέσματα παρουσιάζονται στο Σχήμα 82 και φέρουν το σύστημα να υποστηρίξει οριακά 70 πελάτες. Μεγαλύτερη αύξηση των πελατών προκαλεί

παραβίαση της απαραίτητης συνθήκης εξυπηρέτησης (χρόνος κύκλου $\geq$ χρόνος επεξεργασίας όλων των πελατών από τους δίσκους), γεγονός που εμφανίζεται και στο Σχήμα 80 με την μείωση του ρυθμού άφιξης δεδομένων μετά περίπου τους 70 πελάτες.

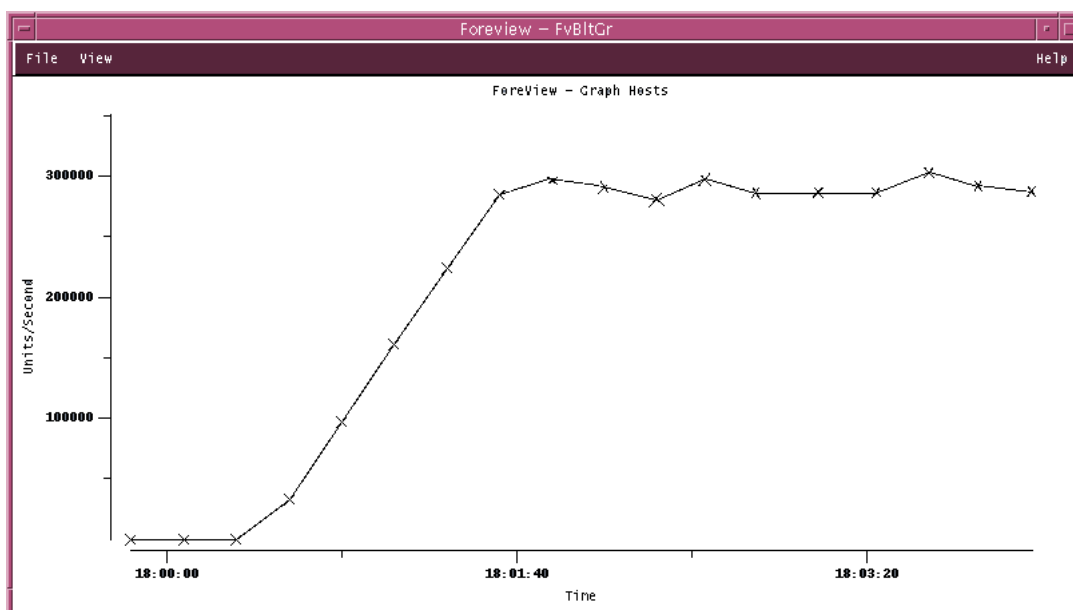
Σε αυτό το σημείο αναδεικνύεται η αποτελεσματικότητα του εφαρμοζόμενου μηχανισμού δρομολόγησης, εφόσον η απόδοση των υποσυστημάτων μετάδοσης και ανάκτησης συμβαδίζουν (70 πελάτες), αλλά και αξιοποιούμε πλήρως τις δυνατότητες του υλικού αποθήκευσης (βλ. Σχήμα 48).

Στο επόμενο κεφάλαιο εξετάζονται οι ενέργειες που πρέπει να λάβουν χώρα για την αύξηση της απόδοσης στο υποσύστημα ανάκτησης δεδομένων του εξυπηρετητή ροής.

6.11.2 Συνολική ρυθμοαπόδοση του συστήματος (Πείραμα)

Σε αυτή την υποενότητα καταγράφουμε με κατάλληλο εργαλείο (foreview) τις ATM κυψέλες που δέχεται ο διακόπτης από το υπολογιστικό σύστημα του εξυπηρετητή. Με αυτόν τον τρόπο επιχειρούμε να έχουμε μια εικόνα για την συνολική απόδοση του συστήματος (throughput).

Τρέχουμε την υπηρεσία διαχείρισης δεδομένων ροής με τις ρυθμίσεις υλικού των προηγούμενων πειραμάτων (Ultra One, 4 δίσκοι, 2 δίαυλοι SCSI, ATM δίκτυο) και ενεργοποιούμε 70 πελάτες να ζητήσουν την αναπαραγωγή αντικειμένων ροής.

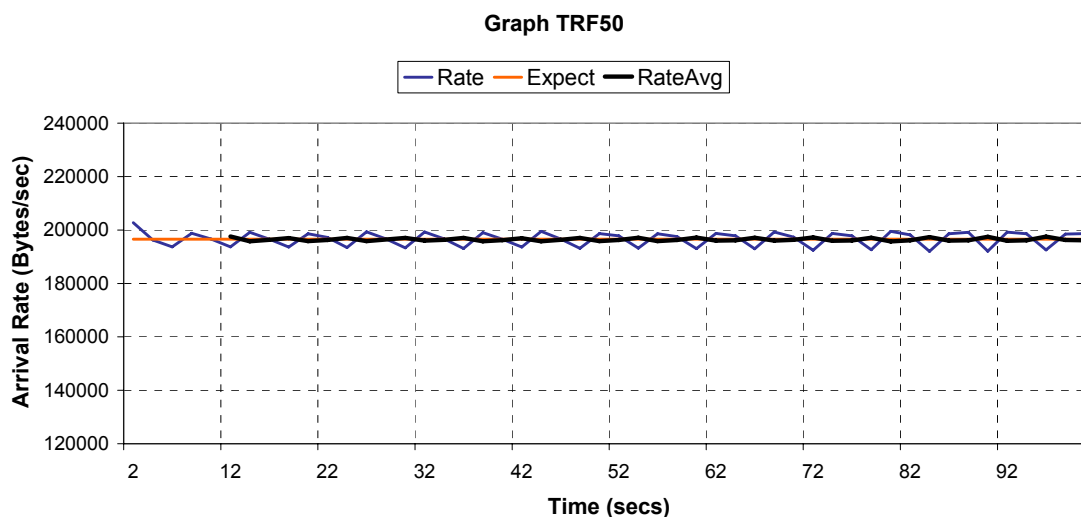


Σχήμα 83: Η συνολική ρυθμοαπόδοση του συστήματος (Μετρήσεις Πειράματος)

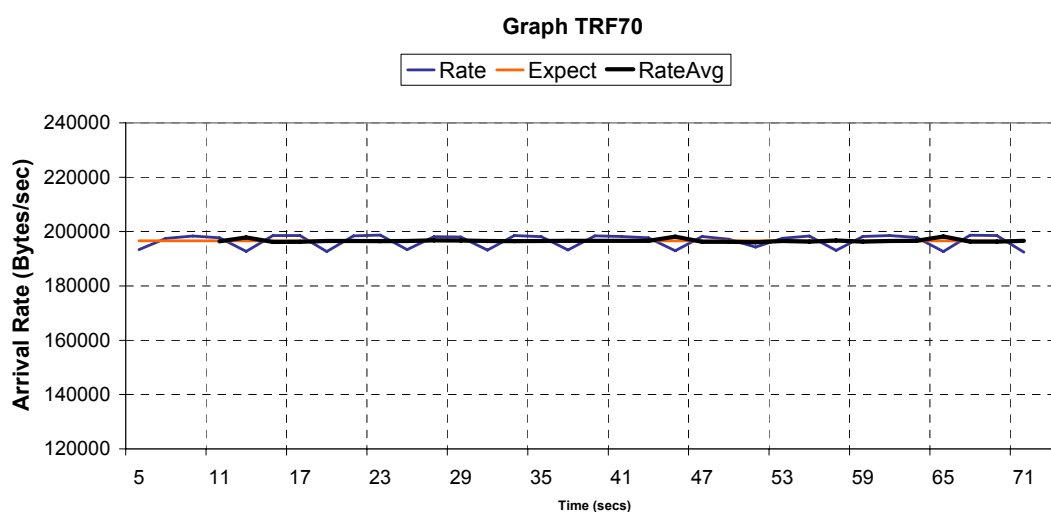
Στο Σχήμα 83 καταγράφεται την συνολική απόδοση του συστήματος, υπολογίζοντας τον ρυθμό μετάδοσης κυψελών στο τοπικό ATM δίκτυο. Ο ρυθμός μετάδοσης παρατηρούμε ότι κυμαίνεται λίγο χαμηλότερα από το επίπεδο των 300000cells/sec ($300000 \times 48 \text{ bytes} = 109 \text{ Mbps}$), που είναι αναμενόμενο αν αναλογιστούμε ότι η εξυπηρέτηση 70 πελατών επιβάλλει συνολικό ρυθμό μετάδοσης δεδομένων $70 \times 1.5 \text{ Mbps} = 105 \text{ Mbps}$.

6.11.3 Αποτελέσματα στην πλευρά των πελατών

Στις ίδιες συνθήκες με την τελευταία υποενοότητα, γίνεται ταυτόχρονη καταγραφή του ρυθμού άφιξης δεδομένων σε τυχαίους πελάτες για να εξετάσουμε αν ο αυξημένος φόρτος εργασίας στον εξυπηρετητή έχει επιπτώσεις στους υπόλοιπους πελάτες εκτός από τον πρώτο. Εξετάζουμε τυχαία τον 50ο και τον 70ο πελάτη



Σχήμα 84: Ρυθμός άφιξης δεδομένων στον 50ο πελάτη



Σχήμα 85: Ρυθμός άφιξης δεδομένων στον 70ο πελάτη

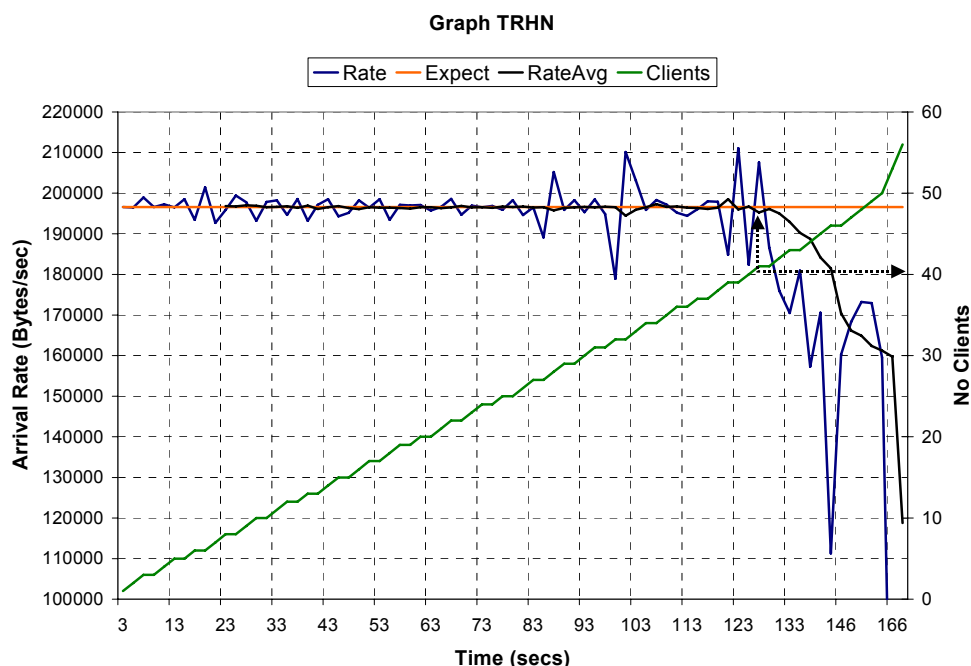
Στο παραπάνω σχήμα εμφανίζονται οι μετρήσεις του ρυθμού άφιξης δεδομένων στους δύο πελάτες. Είναι προφανές ότι το σύστημα συμπεριφέρεται ικανοποιητικά και προσφέρει ποιότητα υπηρεσιών σε όλους τους πελάτες.

6.11.4 Συμπεριφορά της υπηρεσίας ροής σε επεξεργαστικά φορτωμένο περιβάλλον (Πείραμα)

Επαναλαμβάνουμε το πείραμα στο οποίο χρησιμοποιήθηκε ο εξυπηρετητής δεδομένων ροής χωρίς τις υπηρεσίες πραγματικού χρόνου με την διαφορά ότι τώρα το σύστημα είναι επεξεργαστικά φορτωμένο και με άλλες διεργασίες.

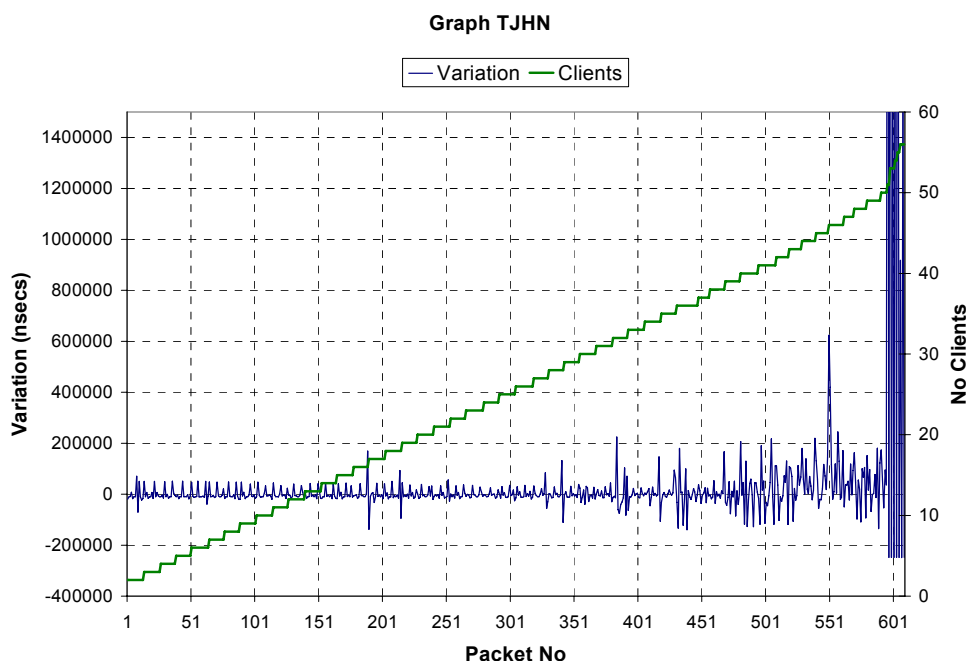
Είναι απολύτως φυσικό μαζί με την διεργασία της υπηρεσίας δεδομένων ροής να τρέχουν στο ίδιο σύστημα πολλές άλλες διεργασίες. Αν ανατρέξουμε στην επισκόπηση της αρχιτεκτονικής του συστήματος Kydonia, θα δούμε ότι είναι απαραίτητη η συνύπαρξη αρκετών άλλων διεργασιών (όπως είναι οι διεργασίες υπηρεσιών του φλοιού DAVIC, ο διαχειριστής αντικειμένων και συνόδων, η υπηρεσία αποθήκευσης δεδομένων και άλλες). Επίσης πιθανή είναι και η συνύπαρξη εξυπηρετητή ιστού (web server) για την ολοκλήρωση του εξυπηρετητή πολυμέσων με εφαρμογές του διαδικτύου.

Γίνεται επομένως εύκολα κατανοητό, ότι πρέπει να μελετηθεί η συμπεριφορά της υπηρεσίας αποθήκευσης δεδομένων ροής σε ένα φορτωμένο επεξεργαστικά περιβάλλον. Κι' αυτό γιατί η υπηρεσία ροής είναι υπεύνη για την δρομολόγηση των εργασιών ανάκτησης και μετάδοσης δεδομένων, που και οι δύο είναι ευαίσθητες στις χρονικές καθυστερήσεις. Είναι απαραίτητο να διασφαλίζεται η ροή των δεδομένων ανεξάρτητα των συνθηκών, ώστε να υπάρχει η δυνατότητα για παροχή εγγυήσεων ποιότητας υπηρεσιών για τις αιτήσεις που έχουν γίνει δεκτές από τον αλγόριθμο ελέγχου υποβολής αιτήσεων του δρομολογητή.



Σχήμα 86 Ρυθμός άφιξης δεδομένων στον πελάτη (Μετρήσεις Πειράματος)
Ρυθμός άφιξης δεδομένων στην πλευρά του πελάτη, χρησιμοποιώντας τον μηχανισμό δρομολόγησης της διατριβής (Χωρίς χρήση υπηρεσιών πραγματικού χρόνου) όταν ο επεξεργαστής είναι φορτωμένος εξυπηρετώντας και άλλες διεργασίες. Όπως προκύπτει από το σχήμα η απόδοση του συστήματος πέφτει δραστικά μετά τους 40 πελάτες, όσο αυξάνεται ο αριθμός των χρηστών που εξυπηρετούνται. Η πολιτική διαμοιρασμού του χρόνου εκτέλεσης αναδεικνύεται συνεπώς μη ικανή για την παροχή εγγυήσεων ποιότητας εξυπηρέτησης.

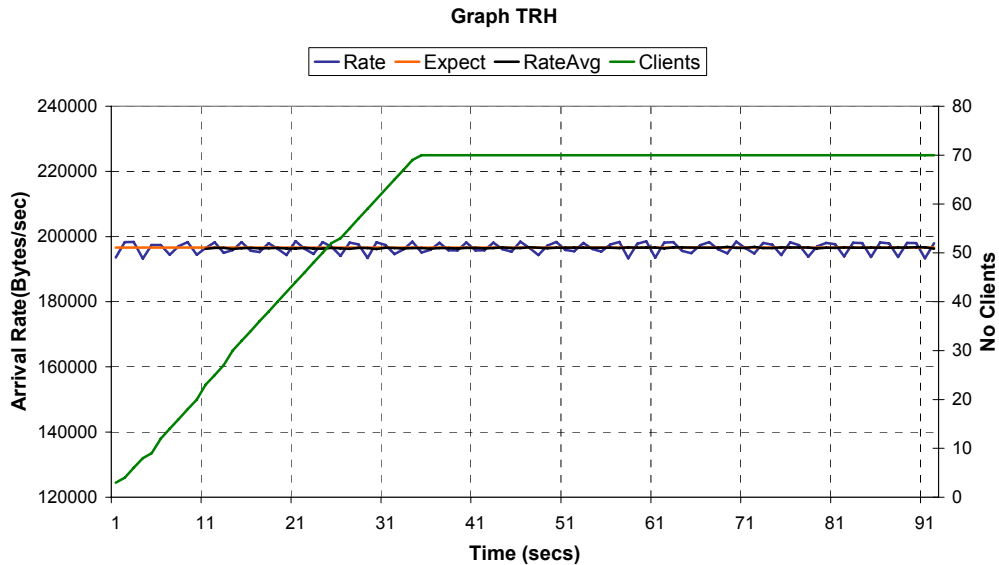
Στο Σχήμα 86 εμφανίζεται γραφικά ο ρυθμός άφιξης των δεδομένων στον πελάτη. Καθώς ο αριθμός των χρηστών αυξάνει το σύστημα αρχίζει να εμφανίζει γρήγορα προβλήματα και τελικά καταρρέει (για περισσότερους από 40 πελάτες). Είναι εμφανές ότι η ίση αντιμετώπιση που προσφέρεται από την κλάση διαμοιρασμού χρόνου που εφαρμόζεται παραδοσιακά δεν επιτρέπει στην διεργασία ροής δεδομένων να παρέχει εγγυήσεις πραγματικού χρόνου, κι' αυτό γιατί χρονικά κρίσιμα νήματα δεν μπορούν να διεκδικήσουν την πρόσβαση στον επεξεργαστή μέσα σε φραγμένο χρονικό διάστημα, με άμεση απόρροια την παραβίαση των προθεσμιών, τόσο στην ανάκτηση όσο και στην μετάδοση των δεδομένων.



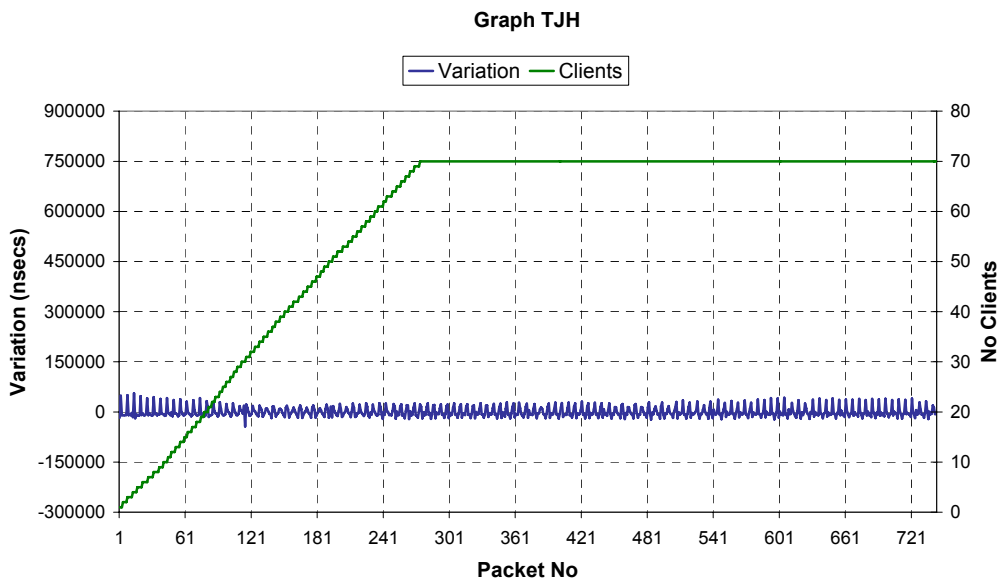
Σχήμα 87: Καθυστέρηση διαδοχικών πακέτων στον πελάτη (Μετρήσεις Πειράματος)
Στο σχήμα εικονίζεται η καθυστέρηση διαδοχικών πακέτων, χρησιμοποιώντας τον μηχανισμό δρομολόγησης της διατριβής (Χωρίς χρήση υπηρεσιών πραγματικού χρόνου), όταν ο επεξεργαστής είναι φορτωμένος εξυπηρετώντας και άλλες διεργασίες. Το σύστημα δεν μπορεί να ανταπεξέλθει στις απαιτήσεις εμφανίζοντας πολύ υψηλό jitter, ως αποτέλεσμα της αδυναμίας της πολιτικής διαμοιρασμού χρόνου εκτέλεσης να δρομολογήσει με αρχές πραγματικού χρόνου.

Όμοια στο Σχήμα 87 φαίνεται πόσο αναποτελεσματική είναι η εφαρμογή της συνηθισμένης δρομολόγησης διεργασιών των λειτουργικών συστημάτων στην παροχή εγγυήσεων στον πελάτη. Η νευρικότητα που παρουσιάζεται στην άφιξη των πακέτων στην πλευρά του χρήστη αντικατροπτίζει την προβληματική λειτουργία του μηχανισμού δρομολόγησης στην πλευρά του εξυπηρετητή. Όπως προκύπτει από τα σχήματα του πειράματος τελικά το σύστημα καταρρέει λόγω έλλειψης πόρων επεξεργαστικής ισχύος και κυρίως διαθεσιμότητας προσωρινής μνήμης.

Επαναλαμβάνουμε το προηγούμενο πείραμα, εφαρμόζοντας τον μηχανισμό δρομολόγησης της διατριβής με χρήση πολυνηματισμού και τεχνικών ανάπτυξης πραγματικού χρόνου. Και σε αυτή την περίπτωση το πείραμα λαμβάνει χώρα σε ένα επεξεργαστικά φορτωμένο περιβάλλον.



Σχήμα 88: Ρυθμός άφιξης δεδομένων στον πελάτη, (Μετρήσεις Πειράματος)
 Ρυθμός άφιξης δεδομένων στην πλευρά του πελάτη, χρησιμοποιώντας τον μηχανισμό δρομολόγησης πραγματικού χρόνου της διατριβής, όταν ο επεξεργαστής είναι φορτωμένος εξυπηρετώντας και άλλες διεργασίες. Η απόδοση του συστήματος εμφανίζεται ανεπηρέαστη από τον υψηλό φόρτο εργασίας του επεξεργαστή και συμπίπτει με την απόδοση του συστήματος σε κανονικές συνθήκες (70 πελάτες).



Σχήμα 89: Καθυστέρηση διαδοχικών πακέτων στον πελάτη(Μετρήσεις Πειράματος)
 Στο σχήμα εικονίζεται η καθυστέρηση διαδοχικών πακέτων, χρησιμοποιώντας τον μηχανισμό δρομολόγησης πραγματικού χρόνου της διατριβής, όταν ο επεξεργαστής είναι φορτωμένος εξυπηρετώντας και άλλες διεργασίες. Το σύστημα εμφανίζει αξιοσημείωτη σταθερότητα, προσφέροντας εγγυήσεις ποιότητας υπηρεσιών στην πλευρά του πελάτη.

Τα αποτελέσματα δείχνουν ότι η ύπαρξη επιπρόσθετου επεξεργαστικού φορτίου από άλλες υπηρεσίες δεν επηρεάζουν καθόλου την λειτουργία της υπηρεσίας ροής. Το αποτέλεσμα αυτό είναι ιδιαίτερα θετικό, καθώς οδηγεί στην ανάπτυξη ενός εξυπηρετητή πολυμέσων, ικανό να παρέχει εγγυήσεις εξυπηρέτησης δεδομένων ροής πραγματικού

χρόνου, ανεξάρτητα τόσο από τον φόρτο εργασίας, που καλείται να αντιμετωπίσει, όσο και από τον φορτίο επεξεργασίας του υπολογιστικού περιβάλλοντος στο οποίο τρέχει.

Στο Σχήμα 89 των καθυστερήσεων άφιξης διαδοχικών πακέτων, αναδεικνύει επίσης την ευστοχία του σχεδιασμού. Η ολοκληρωμένη αντιμετώπιση της διαχείρισης της ροής δεδομένων με τον αποδοτικό συνδυασμό της δρομολόγησης των εργασιών ανάκτησης, προσωρινής αποταμίευσης και μετάδοσης των δεδομένων, έχει επιτύχει τον έλεγχο του ρυθμού και του jitter στην αποστολή πληροφορίας. Δυο χαρακτηριστικά τα οποία έχουν άμεσες επιδράσεις στο επίπεδο εξυπηρέτησης των πελατών.

Σε αυτό το πείραμα επίσης προκύπτει ότι παρά τις αντίξοες συνθήκες επεξεργασίας τα μεσοδιαστήματα διαδοχικών αφίξεων πακέτων εξακολουθούν να είναι φραγμένα σε ικανοποιητικά επίπεδα.

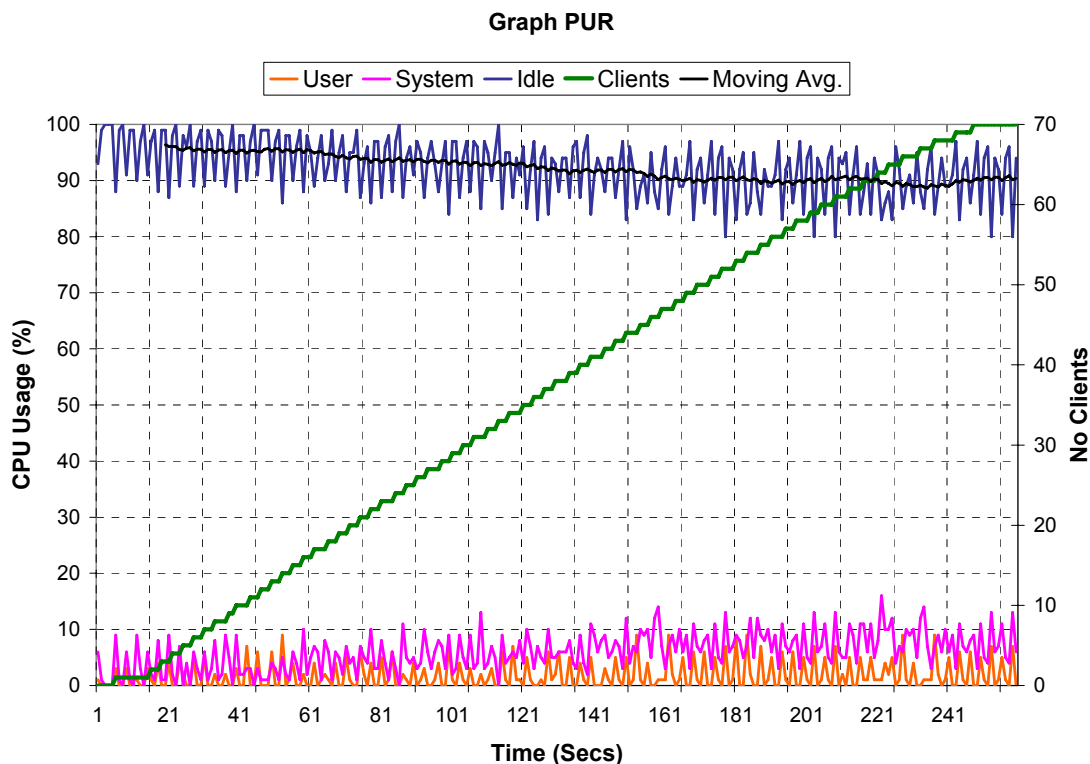
Η λειτουργία όμως της υπηρεσίας ροής σε υψηλά επίπεδα απόδοσης είναι φυσικό να έχει αρνητικές επιπτώσεις, λόγω της ευνοϊκής δρομολόγησης από το λειτουργικό σύστημα, σε άλλες εφαρμογές που πιθανώς χρειάζονται να τρέχουν στο ίδιο υπολογιστικό σύστημα, καθώς μπορούν να εμφανιστούν φαινόμενα παρατεταμένης αναβολής εκτέλεσης (starvation) εξ' αιτίας έλλειψης διαθέσιμης επεξεργαστικής ισχύος.

Σε αυτές τις περιπτώσεις είναι δυνατόν ο κεντρικός δρομολογητής με την βοήθεια του ελεγκτή των πόρων του συστήματος (Σχήμα 70), να αρνείται την εξυπηρέτηση νέων αιτήσεων, όταν η υπηρεσία ροής απασχολεί τον επεξεργαστή πάνω από ένα ποσοστό ασφαλείας για την ομαλή λειτουργία των υπολοίπων εφαρμογών.

6.11.5 Πείραμα χρήσης του επεξεργαστή

Στόχος της ενότητας αυτής είναι η παρακολούθηση της συμπεριφοράς του επεξεργαστή με την εφαρμογή του μηχανισμού δρομολόγησης της διατριβής, σε μια προσπάθεια να αναδείξουμε τις ανάγκες σε επεξεργαστική ισχύ της υπηρεσίας διαχείρισης δεδομένων ροής.

Στο πρώτο πείραμα της ομάδας απομονώνουμε το υποσύστημα μετάδοσης και παρακολουθούμε την χρήση του επεξεργαστή, ενώ το σύστημα εξυπηρετεί 70 πελάτες. Τα ποσοστά χρήσης του επεξεργαστή από τις εργασίες που λαμβάνουν χώρα σε επίπεδο χρήστη και συστήματος εμφανίζονται γραφικά στο Σχήμα 90. Τα αποτελέσματα δείχνουν ότι η εντατική λειτουργία του υποσυστήματος ανάκτησης δεσμεύει τον επεξεργαστή σε ποσοστό μέχρι 10% κατά μέσο όρο. Οι επεξεργαστικές ανάγκες σε επίπεδο χρήστη είναι πολύ μικρές (<5%), ενώ από την άλλη μεριά οι λειτουργίες του συστήματος (κλήσεις υπηρεσιών για ανάκτηση δεδομένων) καταλαμβάνουν λίγο μεγαλύτερο ποσοστό.

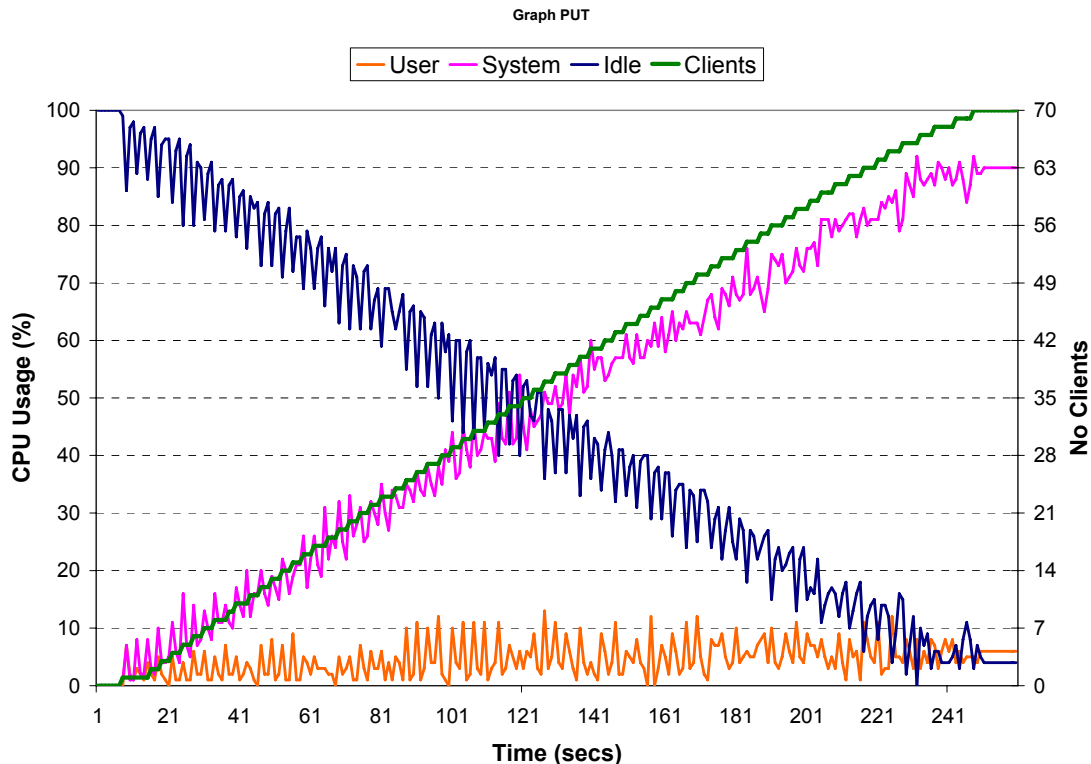


Σχήμα 90: Ποσοστό χρήσης του επεξεργαστή (Μετρήσεις Πειράματος)

Οι μετρήσεις καταγράφονται στο υπολογιστικό σύστημα που τρέχει η υπηρεσία ροής, όταν η λειτουργία του υποσυστήματος μετάδοσης έχει απομονωθεί. Τα παραπάνω διαγράμματα αναδεικνύουν την μικρή επεξεργαστική επιβάρυνση του υποσυστήματος ανάκτησης δεδομένων, ακόμα και σε περιπτώσεις υψηλού φόρτου εργασίας.

Τρέχοντας τώρα το σύστημα συνολικά, είμαστε σε θέση να συγκρίνουμε τις υπολογιστικές ανάγκες των εργασιών ανάκτησης και μετάδοσης. Τα σχεδιαγράμματα στο Σχήμα 91 των ποσοστών χρήσης του επεξεργαστή, αποδεικνύουν ότι οι εργασίες της μετάδοσης δεδομένων έχουν μεγάλες ανάγκες επεξεργαστικής ισχύος. Όπως προκύπτει από την παρατήρηση του σχήματος, το συντριπτικό ποσοστό της χρήσης του επεξεργαστή χρησιμοποιείται για τις υπηρεσίες μετάδοσης δεδομένων του συστήματος. Έτσι είναι γεγονός ότι στην λειτουργία ενός συστήματος εξυπηρέτησης πολυμέσων μεγάλης κλίμακας, εκτός από το υποσύστημα ανάκτησης, σημαντικά προβλήματα φαινομένων συμφόρησης είναι πιθανόν να εμφανιστούν και στην χρήση των επεξεργαστών.

Με βάση την παραπάνω διαπίστωση είναι απαραίτητο το γεγονός να γνωρίζει ο μηχανισμός δρομολόγησης της υπηρεσίας τα ποσοστά δέσμευσης της επεξεργαστικής ισχύος κάτω από διάφορες συνθήκες, με απώτερο στόχο τον κατάλληλο συντονισμό των εργασιών και αποφυγή ανεπιθύμητων φαινομένων συμφόρησης.



Σχήμα 91: Ποσοστό χρήσης του επεξεργαστή (Μετρήσεις Πειράματος)

Στο σχήμα παρουσιάζονται γραφικά οι μετρήσεις που έλαβαν χώρα στο υπολογιστικό σύστημα, όταν η υπηρεσίας ροής εξυπηρετούσε μέχρι τον αριθμό των 70 πελατών. Τα αποτελέσματα υποδεικνύουν το υποσύστημα μετάδοσης να έχει τις μεγαλύτερες επεξεργαστικές ανάγκες

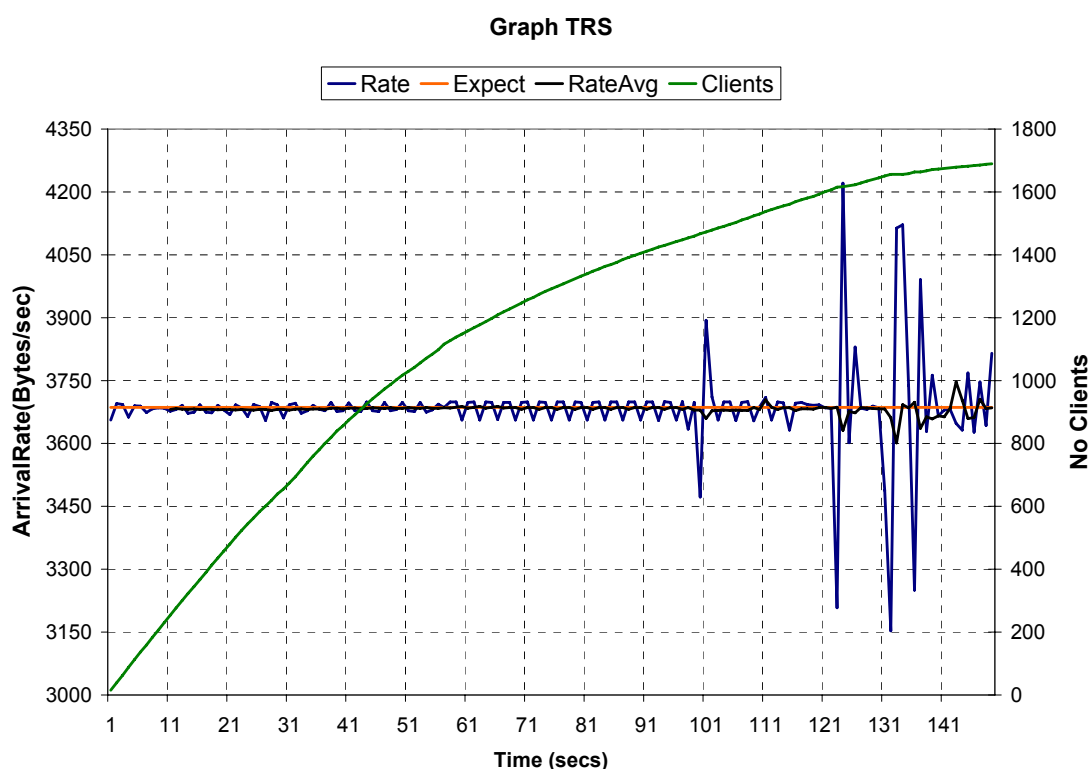
6.12 Δρομολόγηση μεγάλου αριθμού πελατών μικρών απαιτήσεων (Πειράματα)

Στην υποενότητα αυτή εξετάζεται η συμπεριφορά του εφαρμοζόμενου μηχανισμού δρομολόγησης όταν υπάρχει ανάγκη για υποστήριξη πολύ μεγάλου αριθμού πελατών, αλλά με μικρές απαιτήσεις όσον αφορά την ανάκτηση και μετάδοση δεδομένων, όπως συμβαίνει στους εξυπηρετητές ήχου (audio servers).

Η περίπτωση αυτή είναι αρκετά σημαντική και είναι πιθανόν να έχει αντίκτυπο στην απόδοση του συστήματος, καθώς θα αυξήσει τον αριθμό πελατών δραστικά και με αυτόν τον τρόπο θα δημιουργηθούν ανάγκες για μεγαλύτερη προσοχή στην επεξεργασία των αιτήσεων και στον μηχανισμό δρομολόγησης της μετάδοσης δεδομένων.

Στην τρέχουσα ομάδα πειραμάτων αποθηκεύουμε δείγματα αντικειμένων ήχου, οι απαιτήσεις των οποίων είναι 28.8 Kbps. Επίσης υποθέτουμε ότι ένας μεγάλος αριθμός πάνω σε δίκτυο ευρείας κλίμακας είναι διαθέσιμος, με μικρές ικανότητες (π.χ. modem), ώστε να επιβάλλεται η αποστολή δεδομένων με τον απαιτούμενο ρυθμό κατανάλωσης (28.8Kbps). Η αποθήκευση των αντικειμένων διασπάται σε 4 δίσκους και ο κύκλος εξυπηρέτησης καθορίζεται στα 2.5 δεύτερα (τμήμα ανάκτησης 9216 bytes).

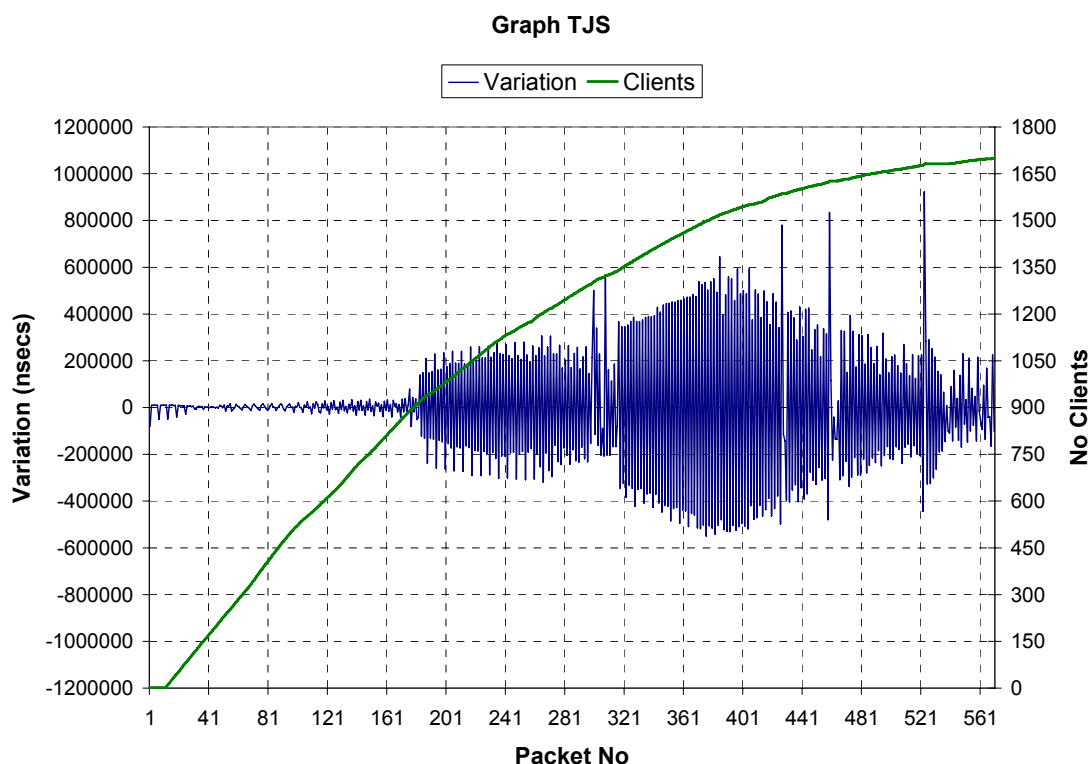
Η επιλογή μικρού σχετικά μεγέθους τμήματος δεδομένων, παρά το γεγονός ότι περιορίζει την απόδοση των δίσκων, έγινε με σκοπό να έχουμε χαμηλό χρόνο απόκρισης αλλά κυρίως για να αντεπεξέλθουμε στις μεγάλες απαιτήσεις μνήμης, όπως θα δούμε στην συνέχεια. Μια διεργασία στέλνει συνεχώς αιτήσεις αναπαραγωγής των αντικειμένων ήχου από εικονικούς πελάτες, ενώ ταυτόχρονα παρατηρούμε πειραματικά τα αποτελέσματα της εξυπηρέτησης στην πλευρά του πρώτου πελάτη. Ο εξυπηρετητής μετάδοσης αποστέλλει πακέτα μεγέθους 4608 bytes.



Σχήμα 92: Ρυθμός άφιξης δεδομένων στον πελάτη (Μετρήσεις Πειράματος)

Στο Σχήμα 92 παρουσιάζεται η γραφική παράσταση του ρυθμού άφιξης δεδομένων στον πρώτο πελάτη, η γραμμή του επιθυμητού ρυθμού και το διάγραμμα του αριθμού των ενεργών πελατών σε σχέση με τον χρόνο. Ο μέσος όρος ρυθμός άφιξης, που εικονίζεται με την έντονη γραμμή, δείχνει ότι ο πελάτης δέχεται τα δεδομένα με τον κανονικό ρυθμό ακόμα και για μεγάλο αριθμό πελατών (1700). Στην περίπτωση όμως αυξημένου φόρτου εμφανίζονται αρκετές διακυμάνσεις.

Καλύτερη εικόνα της εξυπηρέτησης του πελάτη μπορούμε να έχουμε στο Σχήμα 93, στο οποίο καταγράφονται οι διαφοροποιήσεις στους χρόνους άφιξης διαδοχικών πακέτων. Από το σχήμα προκύπτει ότι μέχρι τους 900 πελάτες το jitter της μετάδοσης είναι ελάχιστο, ενώ όσο αυξάνει ο αριθμός πελατών που εισάγονται στην υπηρεσία ροής, αυξάνεται και το jitter, το οποίο όμως εξακολουθεί να διατηρείται σε όχι ιδιαίτερα υψηλά επίπεδα, αν αναλογιστεί κανείς ότι το επιθυμητό χρονικό διάστημα μεταξύ δύο διαδοχικών αφίξεων πακέτων είναι 1250msec.

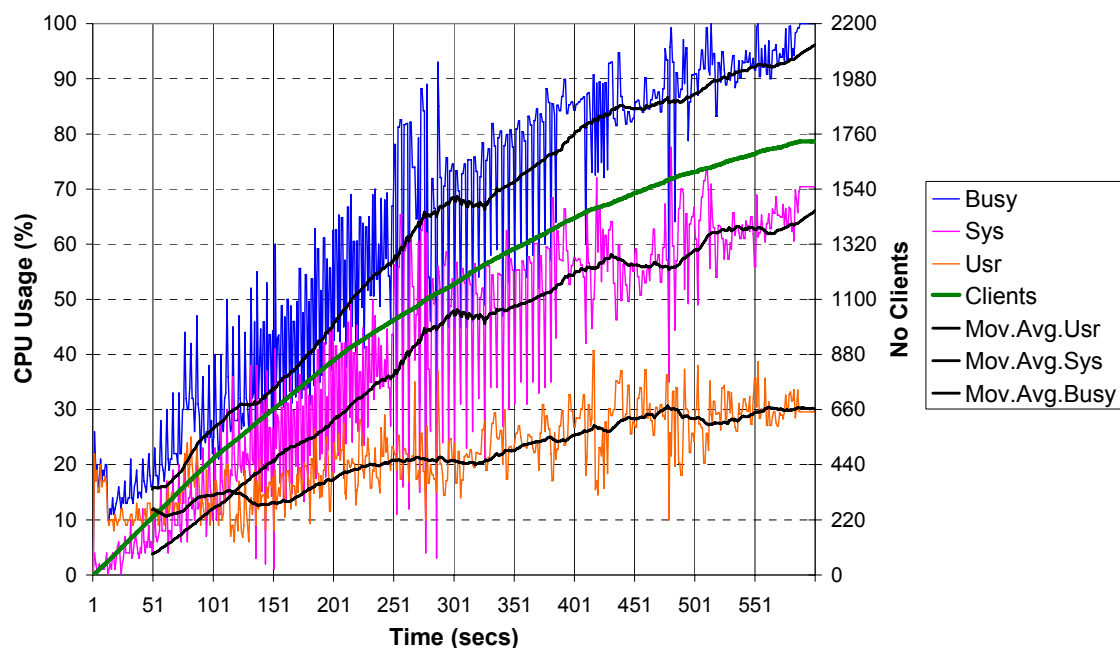


Σχήμα 93: Καθυστερήσεις άφιξης διαδοχικών πακέτων στον πελάτη (Μετρήσεις Πειράματος)

Κατά την διάρκεια του πειράματος παρατηρούμε ότι το σύστημα δεν δέχεται άλλες αιτήσεις πελατών μετά από ένα αριθμό ενεργών χρηστών (~1700), καθώς η σύνδεση των πελατών για την αποστολή των αιτήσεων δεν είναι δυνατή. Το γεγονός αυτό οφείλεται στην φύση του μηχανισμού δρομολόγησης, ο οποίος στην προσπάθεια να διατηρήσει τις εγγυήσεις πραγματικού χρόνου (ικανοποιώντας τις ανάγκες ανάκτησης και μετάδοσης) αδυνατεί να δρομολογήσει εργασίες λιγότερο κρίσιμες, όπως για παράδειγμα την εκτέλεση των νημάτων εισόδου (υπεύθυνο για την λήψη αιτήσεων) και επεξεργασίας των νέων αιτήσεων. Με αυτό τον τρόπο ακούσια επιβάλλεται ένα είδος ελέγχου υποβολής αιτήσεων από τον εξυπηρετητή.

Η συμπεριφορά αυτή μας οδήγησε να συγκεντρώσουμε μετρήσεις από την χρήση του επεξεργαστή κατά την διάρκεια του πειράματος.

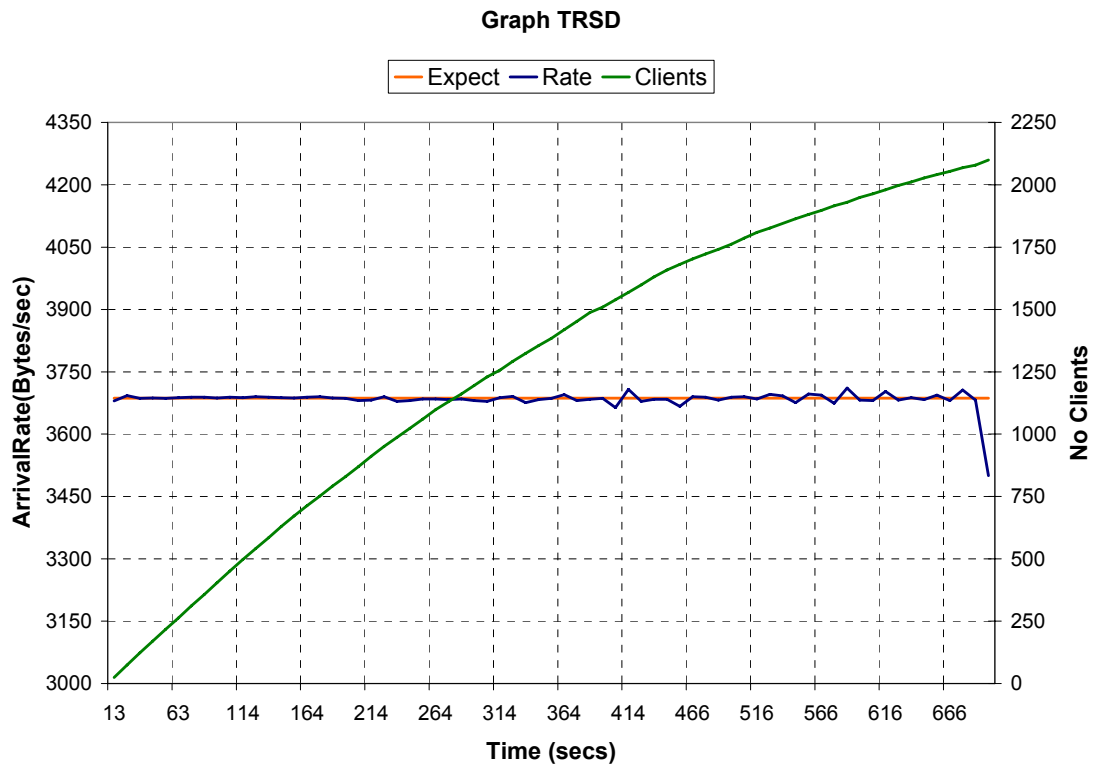
Graph PUS



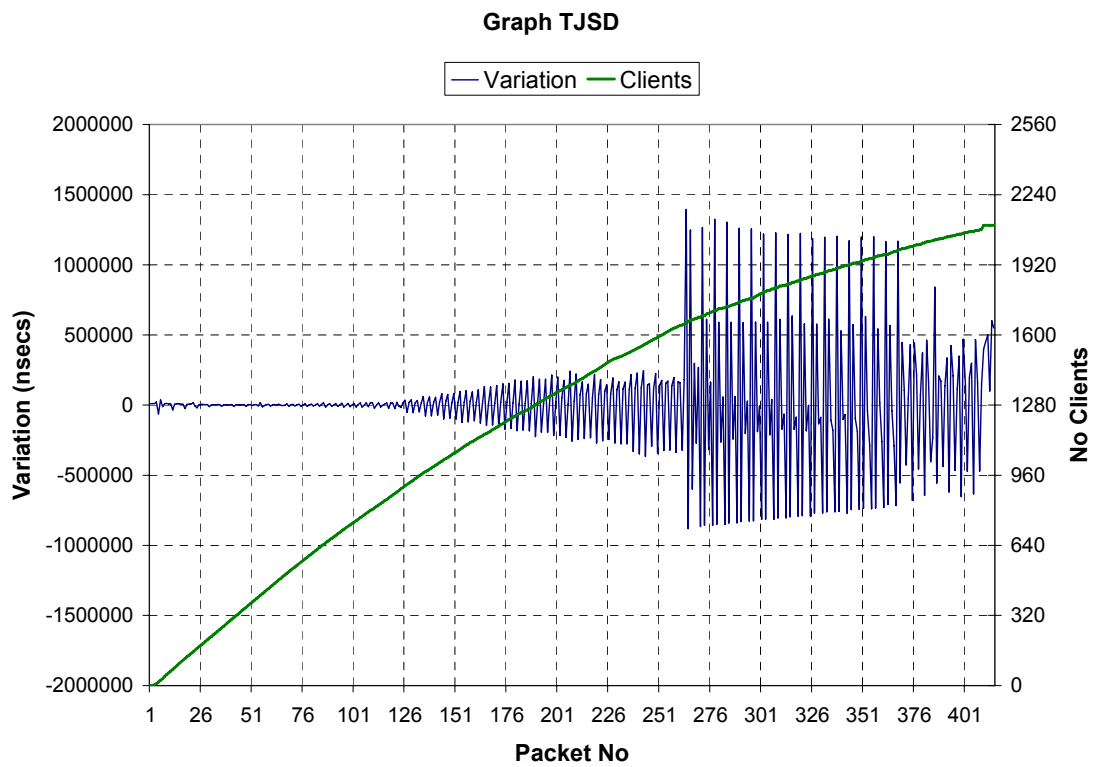
Σχήμα 94: Ποσοστά χρήσης του επεξεργαστή (Μετρήσεις Πειράματος)

Τα αποτελέσματα αυτών των μετρήσεων παραθέτονται στο Σχήμα 94. Έτσι επιβεβαιώνονται τα προηγούμενα, καθώς παρατηρούμε ότι έχουμε πλήρη απασχόληση του επεξεργαστή (100%) για περισσότερους από 1700 πελάτες. Αυξημένο εμφανίζεται το ποσοστό δέσμευσης του επεξεργαστή από εργασίες επιπέδου χρήστη, που κυμαίνεται περίπου στο 30% για τον υψηλότερο φόρτο εργασίας, ενώ οι υπηρεσίες ανάκτησης και μετάδοσης σε επίπεδο συστήματος απασχολούν τον επεξεργαστή σε ποσοστό μεγαλύτερο του 60%. Για την καλύτερη εποπτική εικόνα διαγράφονται οι κινητοί μέσοι όροι των γραφικών παραστάσεων χρήσης του επεξεργαστή.

Μια λύση βελτίωσης της απόδοσης είναι η αύξηση του τμήματος ανάκτησης και του πακέτου μετάδοσης, ώστε να μειωθούν οι επεξεργαστικές απαιτήσεις και να αξιοποιηθούν καλύτερα τα μέσα αποθήκευσης. Έτσι επιχειρούμε να αλλάξουμε τον τρόπο αποθήκευσης και εξυπηρέτησης των αντικειμένων ήχου και παρατηρούμε πάλι τα αποτελέσματα των αλλαγών στην πλευρά του πελάτη. Σε αυτή την περίπτωση διπλασιάζουμε το μέγεθος του τμήματος αποθήκευσης (από 9216 σε 18432 bytes). Ο κύκλος εξυπηρέτησης τώρα διαμορφώνεται στα 5 δεύτερα, ενώ το μέγεθος πακέτων που χρησιμοποιείται στην μετάδοση είναι 9216 bytes.



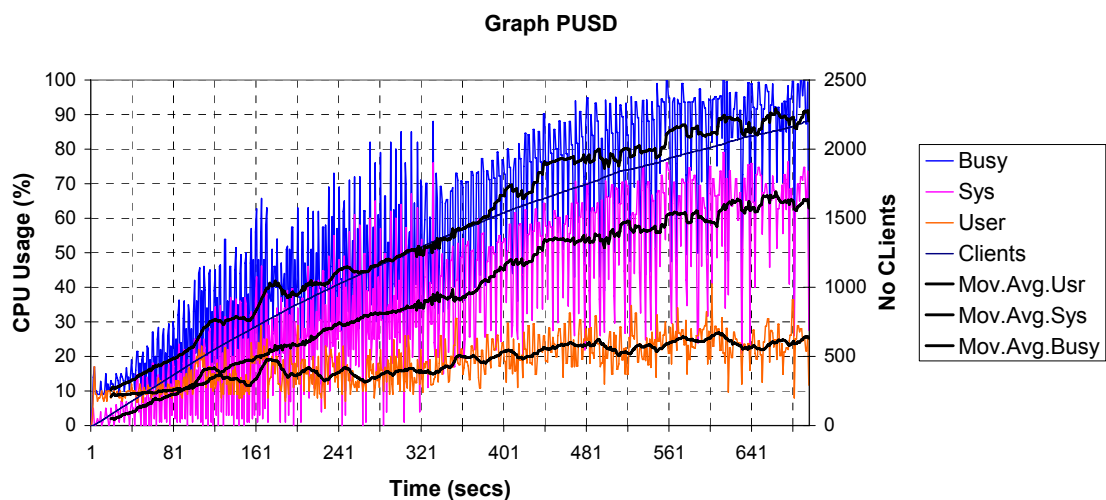
Σχήμα 95: Ρυθμός άφιξης δεδομένων στον πελάτη (Μετρήσεις Πειράματος)



Σχήμα 96: Καθυστερήσεις διαδοχικών πακέτων στον πελάτη (Μετρήσεις Πειράματος)

Ο ρυθμός άφιξης δεδομένων στον πελάτη εμφανίζεται στο Σχήμα 95 και ταυτίζεται κατά μέσο όρο με τον επιθυμητό με πολύ μικρές διακυμάνσεις. Οι αλλαγές που πραγματοποιήθηκαν έχουν, όπως μπορεί να διαπιστωθεί θετικά αποτελέσματα, καθώς το σύστημα δείχνει ότι μπορεί να αντεπεξέλθει στις απαιτήσεις εξυπηρέτησης μεγαλύτερου αριθμού πελατών (~2000). Από την άλλη πλευρά όμως η αύξηση του μεγέθους των τμημάτων ανάκτησης έχει ως τίμημα την αύξηση του αναγκαίου χώρου αποταμίευσης δεδομένων. Έτσι όπως βλέπουμε στο σχήμα αρχίζουν τα προβλήματα απόδοσης στην υπηρεσία ροής, ενώ στην διάρκεια του πειράματος εμφανίστηκαν μηνύματα έλλειψης διαθέσιμου χώρου στον αποταμιευτή επικοινωνίας (το μέγεθος του αποταμιευτή επικοινωνίας ήταν 65 Mbytes).

Το jitter της μετάδοσης διατηρείται από τον εξυπηρετητή σε χαμηλά επίπεδα μέχρι περίπου τον αριθμό των 1700 πελατών (Σχήμα 96) Η μεγαλύτερη αύξηση των ενεργών χρηστών προκαλεί, όπως φαίνεται στο σχήμα, εντονότερες διακυμάνσεις στους χρόνους άφιξης διαδοχικών πακέτων.



Σχήμα 97: Ποσοστά χρήσης του επεξεργαστή (Μετρήσεις Πειράματος)

Στο Σχήμα 97 στο οποίο περιγράφονται τα ποσοστά χρήσης του επεξεργαστή, επιβεβαιώνεται ότι αναφέρθηκε προηγουμένως. Η συνολική δέσμευση του επεξεργαστή ανέρχεται περίπου σε ποσοστό 90%. Έτσι ενώ υπάρχουν ακόμα διαθέσιμοι επεξεργαστικοί πόροι για την εξυπηρέτηση νέων πελατών, αυτό δεν είναι δυνατόν λόγω έλλειψης χώρου αποταμίευσης.

Παρατηρώντας επίσης το τελευταίο σχήμα μπορούμε να βγάλουμε το συμπέρασμα ότι ο διπλασιασμός των τμημάτων ανάκτησης και η αύξηση του μεγέθους των πακέτων μετάδοσης, έχει σαν αποτέλεσμα την μείωση των επεξεργαστικών απαιτήσεων και κατά συνέπεια δίνει την δυνατότητα στο σύστημα να εξυπηρετήσει μεγαλύτερο αριθμό πελατών.

Η αύξηση του τμήματος ανάκτησης και του κύκλου εξυπηρέτησης είναι προφανές ότι οδηγεί σε σημαντική βελτίωση της απόδοσης του υποσυστήματος ανάκτησης. Το σημαντικό όμως τίμημα σε αυτή την περίπτωση είναι η δραστική αύξηση των απαιτήσεων προσωρινής

μνήμης. Μεγαλύτερη ανάλυση του προβλήματος της διαχείρισης της προσωρινής μνήμης και προσπάθειες για την αντιμετώπισή του λαμβάνουν χώρα στο 7ο κεφάλαιο.

6.13 Περίληψη

Στο κεφάλαιο αυτό παρουσιάστηκε ο σχεδιασμός μιας υπηρεσίας λογισμικού, η οποία οργανώνει την αποθήκευση σε μεγάλης κλίμακας συστήματα και παρέχει κατ' απαίτηση δεδομένα ροής σε πολλούς πελάτες. Ο σχεδιασμός επιβάλλει την εύρεση ενός μηχανισμού συντονισμού και δρομολόγησης όλων των επιμέρους εργασιών, ώστε να επιτευχθεί η μέγιστη απόδοση αλλά και να ικανοποιηθούν οι περιορισμοί πραγματικού χρόνου των αντικειμένων ροής.

Οι επιμέρους εργασίες αναφέρονται στην επεξεργασία των εισερχόμενων αιτήσεων, στην ανάκτηση δεδομένων από τα μέσα αποθήκευσης, στην αποταμίευση των δεδομένων στην προσωρινή μνήμη και στην αποστολή των δεδομένων στους πελάτες μέσω του δικτύου.

Βασικό χαρακτηριστικό του σχεδιασμού της υπηρεσίας ροής είναι η χρήση της πολυνηματικής τεχνολογίας μέσω της οποίας γίνεται εφικτή η αποδοτική παράλληλη εκτέλεση των ανεξάρτητων εργασιών. Επιγραμματικά η αρχιτεκτονική της υπηρεσίας ροής περιλαμβάνει:

1. το νήμα ζεύξης και επαφής (Stream Service Interface), το οποίο αποτελεί τον κόμβο εισόδου της υπηρεσίας και δέχεται τις αιτήσεις των πελατών
2. τον κεντρικό δρομολογητή, που οδηγεί την εκτέλεση των αιτήσεων συγκεντρώνοντας πληροφορία από τα υποσυστήματα αποθήκευσης
3. τον ελεγκτή πραγματικού χρόνου, υπεύθυνο για την παροχή χρονική πληροφορίας στο σύστημα
4. τους εξυπηρετητές δίσκων, οι οποίοι αποτελούν νήματα υπεύθυνα για την λειτουργία κάθε μέσου αποθήκευσης
5. και τέλος τον εξυπηρετητή μετάδοσης, ο οποίος αναλαμβάνει την αποστολή των δεδομένων των αντικειμένων ροής στους αιτούμενους πελάτες με τον κατάλληλο τρόπο.

Η δρομολόγηση βασίζεται στην περιοδική φύση των δεδομένων ροής και εξυπηρετεί ταυτόχρονα όλους του πελάτες οδηγώντας την εκτέλεση των αιτήσεων με κύκλους. Σε κάθε κύκλο ένας αριθμός από τμήματα δεδομένων πρέπει να ανακτηθούν για κάθε πελάτη από τους εξυπηρετητές δίσκου. Ο ελεγκτής πραγματικού χρόνου αναλαμβάνει τον χρονικό συντονισμό των εξυπηρετητών δίσκου, παρέχοντάς τους πληροφορία μέσω του κεντρικού δρομολογητή και καθορίζοντας την έναρξη των εργασιών κάθε κύκλου.

Το βασικό πλεονέκτημα που εμφανίζει ο πολυνηματικός σχεδιασμός της υπηρεσίας ροής είναι η αποφυγή ύπαρξης πολλών διεργασιών και συνεπώς ο εκμηδενισμός του μεγάλου κόστους επικοινωνίας με τεχνικές IPC μεταξύ των διεργασιών που υπάρχουν σε άλλους σχεδιασμούς στην βιβλιογραφία.

Εφαρμόζοντας στον εξυπηρετητή μετάδοσης διαφορετικές πολιτικές δρομολόγησης της αποστολής δεδομένων, καταγράψαμε μετρήσεις στην πλευρά του πελάτη. Τα πειράματα έδειξαν ότι η δρομολόγηση της αποστολής δεδομένων είναι πολύ σημαντική όσον αφορά την ποιότητα υπηρεσιών στον πελάτη. Η εφαρμογή της πολιτικής δρομολόγησης αποστολής δεδομένων συνεχόμενα σε τμήματα είχε σαν αποτέλεσμα την απώλεια μεγάλου ποσοστού πακέτων στην πλευρά του πελάτη, ενώ η εφαρμογή της τεχνικής πολύπλεξης των πακέτων διαφορετικών πελατών και την πολιτική EDF εμφάνισε πειραματικά καλύτερα αποτελέσματα.

Λαμβάνοντας υπόψη τα συμπεράσματα του προηγούμενου κεφαλαίου, ότι δηλαδή κάθε πελάτης έχει μια μέγιστη δυνατότητα λήψης δεδομένων, σχεδιάσαμε ένα νέο μηχανισμό δρομολόγησης της μετάδοσης. Ο μηχανισμός αυτός επιτρέπει την πολύπλεξη πακέτων διαφορετικών πελατών και είναι ικανός να οδηγήσει πολλαπλές ροές δεδομένων με διαφορετικούς ρυθμούς και μειωμένο jitter, χρησιμοποιώντας προθεσμίες για την αποστολή των πακέτων. Η αποτελεσματικότητα του μηχανισμού επιβεβαιώθηκε πειραματικά, καθώς επιτύχαμε να δρομολογήσουμε την μετάδοση 81 MPEG1 πελατών χωρίς προβλήματα, με τον εξυπηρετητή να τρέχει σε υπολογιστή Ultra One διασυνδεδεμένο με ATM δίκτυο.

Τα πειράματα που έλαβαν χώρα με ταυτόχρονη λειτουργία των μηχανισμών δρομολόγησης της ανάκτησης και μετάδοσης δεδομένων εμφάνισαν μειωμένη απόδοση. Έτσι ενώ τα πειράματα μετάδοσης έδειξαν ότι το συγκεκριμένο υλικό μπορεί να εξυπηρετεί την μετάδοση 81 πελατών και τα πειράματα ανάκτησης ανταποκρίθηκαν στις ανάγκες επεξεργασίας περίπου 70 χρηστών, εντούτοις τα συνολικά πειράματα της υπηρεσίας ροής έδειξαν μειωμένη απόδοση, η οποία περιορίζεται κάτω από τους 65 πελάτες.

Εφαρμόζοντας τον ολοκληρωμένο μηχανισμό δρομολόγησης, με χρήση προτεραιοτήτων στα νήματα και τεχνικές εκτέλεσης πραγματικού χρόνου, αποδεικνύεται πειραματικά ότι επιτυγχάνεται ο σωστός συνδυασμός της εκτέλεσης των επιμέρους εργασιών στην υπηρεσία ροής και το σύστημα ικανοποιεί την εξυπηρέτηση 70 MPEG1 πελατών, ανεξάρτητα από τον επεξεργαστικό φόρτο του συστήματος από άλλες μη κρίσιμες διεργασίες.

Ο εξυπηρετητής ροής παρουσίασε επίσης καλή συμπεριφορά πειραματικά και στις περιπτώσεις εξυπηρέτησης μεγάλου αριθμού πελατών μικρών απαιτήσεων (εξυπηρετητής ήχου), καθώς κατάφερε να ικανοποιήσει τις απαιτήσεις περίπου 2000 πελατών που ζητούν την αναπαραγωγή αντικειμένων ήχου με ρυθμό κατανάλωσης 28.8kbps.

➤ **Υποστήριξη Ετερογενών Πελατών με Αυτοπροσαρμοζόμενο Μηχανισμό Δρομολόγησης**



Η ανάγκη υποστήριξης πελατών με διαφορετικά χαρακτηριστικά προϋποθέτει ένα δυναμικό μηχανισμό δρομολόγησης και λήψης αποφάσεων. Σε αυτό το κεφάλαιο παρουσιάζεται ένας αυτοπροσαρμοζόμενος μηχανισμός δρομολόγησης που ενσωματώνεται στην υπηρεσία διαχείρισης δεδομένων ροής που παρουσιάστηκε στο προηγούμενο κεφάλαιο. Οι εργασίες που λαμβάνουν χώρα στο κεφάλαιο έχουν σαν στόχο να αξιοποιήσουν τα αποτελέσματα των πειραμάτων του 0ου και 6ου κεφαλαίου. Περιγράφονται οι παράμετροι που οδηγούν την δρομολόγηση και οι ενέργειες που εκτελούνται σε κάθε περίπτωση, ενώ ιδιαίτερη αναφορά γίνεται σε ζητήματα διαχείρισης της προσωρινής μνήμης του συστήματος.

7.1 Ροές δεδομένων και υλικό υποστήριξης

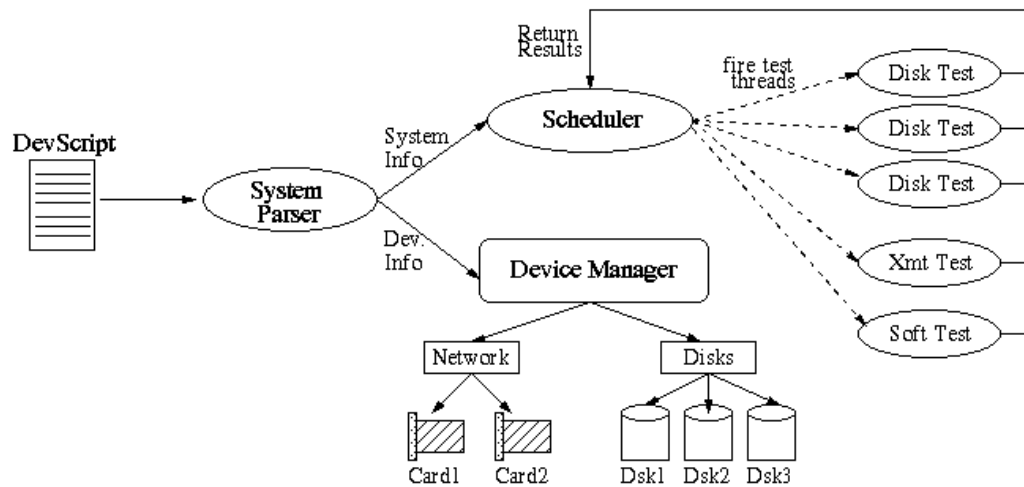
Ένα αποδοτικό σύστημα εξυπηρέτησης δεδομένων πολυμέσων καλείται να ενσωματώσει ένα δυναμικό μηχανισμό δρομολόγησης. Ο μηχανισμός αυτός πρέπει να είναι υπεύθυνος για την επεξεργασία των αιτήσεων των πελατών, για τον αποτελεσματικό συντονισμό των επιμέρους εργασιών ανάκτησης, αποταμίευσης και μετάδοσης δεδομένων, καθώς επίσης και για την συνεχή παρακολούθηση της δέσμευσης των διαθέσιμων πόρων του συστήματος.

Είναι προφανές από τα πειράματα που διενεργήθηκαν και περιγράφηκαν στα κεφάλαια 0 και 6, ότι το υλικό πάνω στο οποίο στηρίζονται οι λειτουργίες λογισμικού της υπηρεσίας ροής κατέχει σημαντικό ρόλο στην απόδοση όλου του συστήματος. Γίνεται επομένως εύκολα κατανοητό ότι ο μηχανισμός δρομολόγησης της υπηρεσίας ροής είναι απαραίτητο να γνωρίζει πληροφορίες σχετικές με τις δυνατότητες του υλικού, έτσι ώστε να οδηγεί κατάλληλα την εκτέλεση των επιμέρους εργασιών και τις ροές δεδομένων.

Στην συνέχεια αυτού του κεφαλαίου παρουσιάζεται το μοντέλο δυναμικής δρομολόγησης, που βασίζεται πάνω στον μηχανισμό του δρομολογητή εργασιών που αναπτύχθηκε στα πλαίσια της διατριβής.

7.2 Ρυθμίσεις συστήματος

Κατά την εγκατάσταση της υπηρεσίας ροής λαμβάνει χώρα η εκτέλεση εφαρμογών ελέγχου των ρυθμίσεων του τρέχοντος συστήματος.



Σχήμα 98: Ρυθμίσεις εγκατάστασης της υπηρεσίας

Στο σχήμα περιγράφονται γραφικά οι αρχικές ενέργειες της υπηρεσίας ροής που λαμβάνουν χώρα κατά την διαδικασία εγκατάστασης του λογισμικού. Ο αναλυτής του συστήματος προωθεί στον κεντρικό δρομολογητή και τον διαχειριστή των συσκευών την απαραίτητη πληροφορία. Στην συνέχεια ο δρομολογητής προκαλεί την δημιουργία νημάτων ελέγχου με στόχο την πειραματική λειτουργία των συσκευών υποστήριξης και την καταγραφή μετρήσεων. Τα αποτελέσματα των μετρήσεων επιστρέφονται στον δρομολογητή, ο οποίος αποκτά με αυτόν τον τρόπο μια εικόνα για τα φυσικά όρια απόδοσης του τρέχοντος συστήματος.

Όπως έχει αναφερθεί στην υποενότητα 4.2.3, ένα τμήμα λογισμικού, που αποτελεί τον αναλυτή του συστήματος, επεξεργάζεται ένα αρχείο script, το οποίο περιέχει πληροφορία σχετικά με τις διαθέσιμες συσκευές που προορίζονται για την υποστήριξη της υπηρεσίας ροής. Το αρχείο αυτό περιγράφει αναλυτικά τις συσκευές δίσκου και τα χαρακτηριστικά τους. Έτσι το σύστημα είναι σε θέση να ενημερωθεί για το όνομα πρόσβασης της συσκευής, την χωρητικότητά της, την ονομαστική της απόδοση, καθώς επίσης και για το αναγνωριστικό του διαύλου SCSI πάνω στον οποίο είναι προσαρμοσμένη η συσκευή αποθήκευσης. Το αρχείο αυτό έχει εμπλουτιστεί και με επιπρόσθετες πληροφορίες, τις οποίες οφείλει να εισάγει ο υπεύθυνος του συστήματος, ανάλογα με το υπάρχων υλικό (π.χ. δικτυακές συσκευές, διαθέσιμη κύρια μνήμη, μέγεθος αποταμιευτή επικοινωνίας, μέγεθος προσωρινής μνήμης συστήματος-heap, μεγέθη ουρών, χρονική τιμή κύκλου και υποκύκλων του συστήματος κ.α.).

Είναι γενικά αποδεκτό ότι δεν μπορούμε να βασιστούμε στην πληροφορία των ονομαστικών αποδόσεων των συσκευών (δίσκοι, κάρτες δικτύου) που παρέχονται από τους κατασκευαστές, καθώς, όπως μπορεί εύκολα να αποδειχθεί από πειράματα που έλαβαν χώρα σε αυτό το κεφάλαιο, ότι οι αποδόσεις των συσκευών κάτω από πραγματικές συνθήκες είναι χαμηλότερες, ιδιαίτερα όταν είναι αναγκαίος ο συνδυασμός λειτουργίας πολλαπλών πόρων ταυτόχρονα. Επίσης βασικό παράγοντα έρχεται να αποτελέσει το υπολογιστικό σύστημα πάνω στο οποίο εφαρμόζονται οι περιφερειακές συσκευές, ενώ καταλυτική παράμετρος σε αυτή την περίπτωση αναδεικνύεται η επεξεργαστική του ισχύς.

Στο Σχήμα 98 περιγράφονται γραφικά τα βήματα που ακολουθούνται κατά την διάρκεια της εγκατάστασης της υπηρεσίας ροής σε ένα υπολογιστικό σύστημα. Ο αναλυτής του συστήματος επεξεργάζεται το αρχείο DeviceScript και εισάγει την πληροφορία στις κατάλληλες δομές της υπηρεσίας ροής, μέσω των οποίων ενημερώνεται ο κεντρικός δρομολογητής με στοιχεία απαραίτητα για την λειτουργία του λογισμικού (π.χ. μεγέθη αποταμιευτών, ουρών, χρονικά όρια δρομολόγησης κ.α.) και ταυτόχρονα δημιουργούνται οι δομές που αντιπροσωπεύουν τις συσκευές δίσκου και δικτύου στον διαχειριστή συσκευών. Στην συνέχεια ο κεντρικός δρομολογητής δημιουργεί μια σειρά από νήματα, τα οποία εκτελούν ένα κώδικα ελέγχου όλων των συσκευών και καταγράφονται διάφορες μετρήσεις. Τα αποτελέσματα των μετρήσεων επιστρέφονται στον κεντρικό δρομολογητή που πλέον φέρει την κατάλληλη πληροφορία για να συντονίσει τις επιμέρους εργασίες. Οι εφαρμογές ελέγχου αναδεικνύουν την απόδοση της πρόσβασης σε πόρους του συστήματος σε πραγματικό περιβάλλον. Με αυτόν τον τρόπο ο δρομολογητής είναι σε θέση να εφαρμόσει ένα απλό μηχανισμό ελέγχου υποβολής αιτήσεων εξυπηρέτησης, ακολουθώντας μια ντετερμινιστική πολιτική για τις χειρότερες περιπτώσεις, ενώ ταυτόχρονα αλλάζει δυναμικά παραμέτρους του μηχανισμού ροής δεδομένων, ώστε να επιτευχθεί η μέγιστη δυνατή απόδοση.

Όσον αφορά την εύρεση της απόδοσης των δίσκων, το σύστημα έχει προγραμματιστεί ώστε να υπολογίζει αυτόματα την μέση απόδοση του κάθε δίσκου, όταν όλοι οι διαθέσιμοι δίσκοι εργάζονται κατά το μέγιστο δυνατό με δεδομένο το τρέχων σχήμα προσαρμογής στους διαύλους SCSI. Με αυτόν τον τρόπο ο μηχανισμός δρομολόγησης συνυπολογίζει τις καθυστερήσεις που προέρχονται από την ταυτόχρονη χρήση κοινών διαύλων.

Δεχόμαστε ότι ο χρόνος προσπέλασης ενός τμήματος δεδομένων σε ένα δίσκο ακολουθεί το ευρέως διαδεδομένο και πιο απλό μοντέλο:

$$T_R = T_a(cyl) + \frac{size_b}{T_d}$$

όπου T_R :ο χρόνος ανάκτησης ενός τμήματος δεδομένων από τον δίσκο

$T_a(cyl)$:ο χρόνος προσπέλασης των δεδομένων (seek + rotational delay)

$Size_b$:το μέγεθος του τμήματος δεδομένων

T_d :ο ρυθμός ανάκτησης δεδομένων της κεφαλής

Ο χρόνος T_a υπολογίζεται από μια συνάρτηση με παράμετρο την απόσταση της αναζήτησης (concave function, [Reum94]).

Στην φάση ελέγχου του υλικού δημιουργούνται νήματα από τον κεντρικό δρομολογητή, ένα για κάθε δίσκο. Κάθε νήμα υποβάλλει συνεχώς αιτήσεις ανάκτησης δεδομένων στον αντίστοιχο δίσκο, ενώ συγχρόνως καταγράφει τους χρόνους καθυστέρησης μιας αίτησης ανάκτησης δεδομένων συγκεκριμένου μεγέθους για κάθε δίσκο, σύμφωνα με τις τρέχουσες ρυθμίσεις και κάτω από συνθήκες υψηλού φόρτου εργασίας. Έχοντας σαν στόχο να προσεγγίσουμε όσο τον δυνατόν καλύτερα τις πραγματικές συνθήκες λειτουργίας των δίσκων κατά την διάρκεια της ρύθμισης, τα νήματα ελέγχου των δίσκων εργάζονται σε κύκλους. Για την εκτέλεση του ελέγχου παράγεται ένας αριθμός από υποαιτήσεις ανάκτησης. Οι υποαιτήσεις φέρουν σταθερό μέγεθος και η θέση των τμημάτων στο δίσκο

καθορίζεται με την βοήθεια μιας γεννήτριας τυχαίων αριθμών. Η υπηρεσία ροής σε κάθε κύκλο επιβάλλει την πολιτική SCAN στην εκτέλεση των υποαιτήσεων.

Υποθέτουμε επομένως ότι ο εξυπηρετητής επεξεργάζεται N αντικείμενα ροής σε κάθε δίσκο. Κατά την διάρκεια του κύκλου εξυπηρέτησης (T) ανακτάται ένα τμήμα δεδομένων μεγέθους S για κάθε ένα από τα N αντικείμενα ροής. Ο χρόνος T_{Di} που χρειάζεται ο κάθε δίσκος Di για να εξυπηρετήσει όλες τις αιτήσεις ανάκτησης των τμημάτων των αντικειμένων προκύπτει από την σχέση:

$$T_{Di} = \sum_{j=1}^N Ta(cyl_j) + \frac{N \times S}{T_{Ri}}$$

Αποδεικνύεται ότι το χειρότερο κόστος προσπέλασης δεδομένων με εφαρμογή του αλγορίθμου SCAN εμφανίζεται όταν τα τμήματα δεδομένων βρίσκονται διασκορπισμένα ομοιόμορφα στην επιφάνεια του δίσκου. Έτσι στην χειρότερη περίπτωση έχουμε κόστος προσπέλασης ίσο με $N \times Ta(\frac{TotCyl}{N})$. Συνεπώς ο ουσιαστικός ρυθμός ανάκτησης δεδομένων T_{RefDi} βρίσκεται με χρήση της παρακάτω εξίσωσης:

$$T_{RefDi} = \frac{N \times S}{N \times Ta(\frac{TotCyl}{N}) + \frac{N \times S}{T_{Ri}}} = \frac{S}{S + (Ta(\frac{TotCyl}{N}) \times T_{Ri})} \times T_{Ri}$$

Από την παραπάνω σχέση γίνεται εμφανές ότι όσο μεγαλώνει το μέγεθος των τμημάτων δεδομένων S , τόσο αυξάνει ο εφικτός ρυθμός ανάκτησης δεδομένων από τον δίσκο, ο οποίος τείνει ασυμπτωτικά προς το επίπεδο T_{Ri} της μέγιστης απόδοσης της συσκευής.

Ο κεντρικός δρομολογητής της υπηρεσίας ροής προσπαθεί μέσα από την πειραματική διαδικασία της εγκατάστασης του συστήματος να συλλέξει τιμές απόδοσης των δίσκων για διάφορες τιμές μεγέθους των τμημάτων δεδομένων.

Ο Πίνακας 4 συγκεντρώνει τις τιμές $T_{RefDi,bj}$ που προκύπτουν από τα νήματα ελέγχου των δίσκων και αντιπροσωπεύουν τους πραγματικούς ρυθμούς ανάκτησης των m δίσκων για ένα εύρος τιμών μεγέθους τμημάτων $j=1,...,n$.

$T_{RefD1,b1}$	$T_{RefD1,b2}$		$T_{RefD1,bn}$
.....			
$T_{RefDm,b1}$	$T_{RefDm,b2}$		$T_{RefDm,bn}$

Πίνακας 4: Πειραματικές μετρήσεις της απόδοσης των δίσκων

Οι τιμές αυτές επιστρέφονται στον κεντρικό δρομολογητή και παίζουν σημαντικό ρόλο στην προσπάθεια του μηχανισμού δρομολόγησης να παρέχει εγγυήσεις ποιότητας εξυπηρέτησης, υπολογίζοντας αν το υποσύστημα ανάκτησης δεδομένων μπορεί να αντεπεξέλθει σε νέα αύξηση φόρτου εργασίας.

Με στόχο να έχουμε μια καλύτερη εικόνα, αυτής της αυτοματοποιημένης ρύθμισης, τρέχουμε τα νήματα ελέγχου πάνω στο υλικό που έτρεχε η υπηρεσία ροής σε όλα τα

προηγούμενα πειράματα (Ultra One, 4 δίσκοι, 2 διαύλοι SCSI). Τα αποτελέσματα των μετρήσεων με τα οποία ενημερώνεται ο δρομολογητής παρουσιάζονται στον επόμενο πίνακα (Πίνακας 5). Ο πειραματικός έλεγχος έλαβε χώρα για μεγέθη τμημάτων δεδομένων από 8K έως 10M, ενώ ταυτόχρονα καταγράφεται το ποσοστό χρήσης του επεξεργαστή κατά την διάρκεια εκτέλεσης των νημάτων και το ποσοστό αξιοποίησης των διαύλων SCSI (σε σχέση με την ονομαστική τους απόδοση).

Block Size (Bytes)	Retrieval Rate per Disk (MBps)	SCSI Utilization (%)	CPU Usage(%)
1K	0.09	1.8	11
8K	0.6	12	10
16K	1.05	21	9
32K	1.6	32	8.5
64K	2.2	44	7.5
128K	2.5	50	6
256K	3.3	66	5.5
512K	3.7	74	5
1M	3.9	78	5
2M	4.05	81	5
4M	4.2	84	5
8M	4.25	85	5
10M	4.25	85	5

Πίνακας 5

Σε πραγματικές συνθήκες το υποσύστημα ανάκτησης μπορεί να δέχεται πολλαπλές αιτήσεις διαφορετικών μεγεθών τμημάτων που αντιστοιχούν σε αντικείμενα με διαφορετικούς ρυθμούς κατανάλωσης, τις οποίες καλείται να επεξεργαστεί. Σε κάθε κύκλο εξυπηρέτησης ένα ή περισσότερα τμήματα για κάθε πελάτη (διαφορετικού μεγέθους) πρέπει να ανακτηθούν. Για την ικανοποίηση των απαιτήσεων των αντικειμένων ροής θα πρέπει ο ρυθμός ανάκτησης δεδομένων από τον δίσκο να υπερέχει του αθροίσματος του απαραίτητου ρυθμού κατανάλωσης δεδομένων των ενεργών αντικειμένων ροής, δηλαδή :

$$\sum_{i=1}^N Cr_i \leq T_{RefD} = \frac{\sum_{i=1}^N S_{bi}}{T_D} = \frac{\sum_{i=1}^N S_{bi}}{N \times Ta + \frac{\sum_{i=1}^N S_{bi}}{T_R}} = \frac{T_R \times \sum_{i=1}^N S_{bi}}{N \times Ta T_R + \sum_{i=1}^N S_{bi}} =$$

$$= \frac{N \times T_R \times Msize}{N \times Ta \times T_R + N \times Msize} = \frac{Msize}{Msize + Ta \times T_R} \times T_R = T_{RefD, Msize} \Rightarrow$$

$$\Rightarrow \sum_{i=1}^N Cri \leq T_{RefD, Msize}$$

όπου Cr_i : ο ρυθμός κατανάλωσης για το αντικείμενο i

T_R : ο ρυθμός ανάκτησης δεδομένων από τον δίσκο D

Sb_i : το μέγεθος του τμήματος b_i

$Msize$: ο μέσος όρος των μεγεθών των N τμημάτων.

$T_{RefD, Msize}$: ο ρυθμός ανάκτησης του δίσκου D τμημάτων δεδομένων μεγέθους $Msize$

Συνεπώς ο δρομολογητής διατηρώντας τον μέσο όρο των τμημάτων που πρέπει να ανακτηθούν σε κάθε κύκλο για κάθε δίσκο και γνωρίζοντας τους ρυθμούς κατανάλωσης του αντικειμένου ροής μπορεί να αποφασίσει με την βοήθεια του πίνακα τιμών (Πίνακας 4) αν οι συσκευές δίσκων μπορούν να ικανοποιήσουν τις απαιτήσεις εξυπηρέτησης.

Τα επόμενα βήματα στις αρχικές ρυθμίσεις εγκατάστασης είναι η εύρεση της μέγιστης ικανότητας μετάδοσης δεδομένων, καθώς επίσης και σε πιο βαθμό ο επεξεργαστής μπορεί να ικανοποιήσει τις απαιτήσεις των μηχανισμών ανάκτησης, μετάδοσης δεδομένων και δρομολόγησης των εργασιών. Έτσι ο κεντρικός δρομολογητής δημιουργεί ένα νέο νήμα ελέγχου, το οποίο αναλαμβάνει να μεταδίδει δεδομένα αρχίζοντας από τον μικρότερο μέχρι τον μέγιστο ρυθμό. Ταυτόχρονα καταγράφεται το ποσοστό χρήσης του επεξεργαστή. Ο επόμενος πίνακας αποτελεί ένα παράδειγμα καταγραφής μετρήσεων για τον μηχανισμό μετάδοσης για ένα τον σταθμό εργασίας Ultra One.

Ρυθμός αποστολής δεδομένων (Mbps)	Ποσοστό χρήσης του επεξεργαστή (%)
14	10
25	20
45	35
65	49
122	96

Από τις πειραματικές τιμές, που καταγράφονται στον παραπάνω πίνακα, διαπιστώνουμε ότι υπάρχει μια σχεδόν γραμμική σχέση μεταξύ του ρυθμού μετάδοσης και του ποσοστού χρήσης του επεξεργαστή.

Οι πειραματικές ρυθμίσεις τελειώνουν με ένα άλλο νήμα ελέγχου, αντικειμενικός στόχος του οποίου είναι η εύρεση των επεξεργαστικών απαιτήσεων των τμημάτων λογισμικού της υπηρεσίας ροής και κυρίως του μηχανισμού δρομολόγησης, αν και στην ενότητα 6.11.5 διαπιστώθηκε πειραματικά ότι το μεγαλύτερο ποσοστό επεξεργαστικής ισχύος δεσμεύεται από την χρήση των υπηρεσιών του συστήματος για την ανάκτηση αλλά κυρίως για την αποστολή πακέτων δεδομένων στους πελάτες.

Ένα ακόμα παράγοντα που εξετάζει το νήμα ελέγχου λογισμικού είναι η ακρίβεια στους χρονικούς κύκλους που μπορεί να προσφέρει το εκάστοτε υπολογιστικό σύστημα.

7.3 Η πολιτική δρομολόγησης για την υποστήριξη ετερογενών πελατών

Ο μηχανισμός δρομολόγησης που σχεδιάστηκε και αναπτύχθηκε στα πλαίσια της διατριβής αναδείχθηκε αρκετά αποτελεσματικός, σύμφωνα με τα πειράματα που προηγήθηκαν σε άλλα κεφάλαια. Η υποενότητα αυτή περιέχει τις απαραίτητες προσθήκες και βελτιώσεις που εμπλουτίζουν τον υπάρχοντα μηχανισμό δρομολόγησης, μετατρέποντας τον τελευταίο σε ένα αποδοτικό και δυναμικά προσαρμοζόμενο διαχειριστή του συστήματος.

Σύμφωνα με τα όσα έχουν παρουσιαστεί στις προηγούμενες ενότητες η υπηρεσία ροής είναι ικανή να υποστηρίζει μεγάλο αριθμό πελατών που ζητούν την αναπαραγωγή διαφόρων αντικειμένων ροής, ενώ ταυτόχρονα παρέχει εγγυήσεις ποιότητας πραγματικού χρόνου, εφόσον οι απαιτήσεις επεξεργασίας των αιτήσεων κυμαίνονται μέσα στα φυσικά όρια του υλικού υποστήριξης.

Στην ενότητα 5.3 έγιναν πειράματα με στόχο να εξετάσουμε την μέγιστη ικανότητα μετάδοσης δεδομένων διαφόρων υπολογιστικών συστημάτων. Στην συνέχεια (ενότητες 6.4.2,6.4.3,6.4.4) μελετήθηκε πειραματικά η εφαρμογή στον εξυπηρετητή μετάδοσης διαφορετικών πολιτικών δρομολόγησης και καταγράφηκε η απόδοσή τους. Οι μεγάλες διαφοροποιήσεις στην ποιότητα υπηρεσιών στην πλευρά του πελάτη, οι οποίες προκαλούνταν από την εφαρμογή διαφορετικών πολιτικών μετάδοσης, οφείλονται κυρίως στα χαρακτηριστικά του πελάτη. Πειράματα που έλαβαν χώρα στην ενότητα 5.4 ανάδειξαν το γεγονός ότι κάθε πελάτης ανάλογα με το υλικό που χρησιμοποιεί (επεξεργαστή, δικτυακή υποδομή κ.α.), είναι σε θέση να δεχθεί δεδομένα χωρίς απώλειες μέχρι ένα μέγιστο ρυθμό άφιξης πακέτων, ο οποίος ποικίλει από πελάτη σε πελάτη. Έχοντας σαν δεδομένο το συμπέρασμα αυτό σχεδιάστηκε ο κατάλληλος μηχανισμός δρομολόγησης της μετάδοσης δεδομένων, ο οποίος λαβαίνει υπόψη του τον θεμιτό ρυθμό μετάδοσης για κάθε αντικείμενο ροής. Ο μηχανισμός οδηγείται με την χρήση χρονικών προθεσμιών για την αποστολή των πακέτων, ενώ ταυτόχρονα επιτρέπεται η πολύπλεξη πακέτων δεδομένων διαφορετικών πελατών. Με τον τρόπο αυτό επιτυγχάνεται η θεμιτή ροή μετάδοσης προς τον πελάτη με σταθερό ρυθμό και μειωμένο jitter.

Σημαντική σημασία στον σχεδιασμό της υπηρεσίας ροής είχε η μελέτη της απόδοσης του υποσυστήματος ανάκτησης δεδομένων, ενώ ουσιαστική συνεισφορά αποτελεί η προτεινόμενη συνολική αντιμετώπιση του ζητήματος της δρομολόγησης εργασιών με εγγυήσεις πραγματικού χρόνου (ενότητα 6.10), μέσω της εφαρμογής της οποίας επιτυγχάνεται ο ιδανικός συνδυασμός της λειτουργίας όλων των τμημάτων λογισμικού, ώστε να έχουμε την μεγαλύτερη δυνατή απόδοση και την πλήρη αξιοποίηση των διαθέσιμων φυσικών πόρων.

Από τα προηγούμενα γίνεται εύκολα κατανοητό ότι η απόδοση της υπηρεσίας διαχείρισης δεδομένων ροής επηρεάζεται σε μεγάλο βαθμό τόσο από τα χαρακτηριστικά

υλικού του εξυπηρετητή, αλλά και από τα αντίστοιχα χαρακτηριστικά των πελατών που ζητούν υπηρεσίες. Το λογισμικό δρομολόγησης του συστήματος επομένως, είναι απαραίτητο να εφοδιαστεί με την ανάλογη πληροφορία. Ήδη στην προηγούμενη υποενότητα περιγράφηκε ο τρόπος με τον οποίο το σύστημα αυτόματα ελέγχει το διαθέσιμο υλικό υποστήριξης, κατά την διάρκεια της εγκατάστασης και συγκεντρώνει χρήσιμες πληροφορίες σχετικά με τις φυσικές δυνατότητες του υπολογιστικού συστήματος.

Όσον αφορά την δυνατότητα προσαρμογής του δρομολογητή σε ανοιχτά περιβάλλοντα, το πρώτο βήμα σε αυτή την προσπάθεια είναι η συγκέντρωση των χαρακτηριστικών των ετερογενών πελατών και η εύρεση των πραγματικών ικανοτήτων τους. Οι περιπτώσεις που μπορούμε να συναντήσουμε είναι οι ακόλουθες:

1. *Πελάτες με μεγάλη χωρητικότητα αποθήκευσης δεδομένων σε δίσκους ή/και προσωρινούς αποταμιευτές*
2. *Πελάτες με μικρούς διαθέσιμους χώρους αποταμίευσης δεδομένων*
3. *Πελάτες με διαφορετική δικτυακή υποδομή*
4. *Πελάτες με διαφορετική επεξεργαστική ισχύ*
5. *Πελάτες με διαφορετικό υλικό ή/και λογισμικό αποκωδικοποίησης και αναπαραγωγής των αντικειμένων ροής.*

Σύμφωνα με τα αποτελέσματα των πειραμάτων, ο εξυπηρετητής μετάδοσης πρέπει να αποστέλλει πακέτα στο δίκτυο με ένα συγκεκριμένο μέγιστο ρυθμό, διαφορετικά αυξάνονται σημαντικά οι πιθανότητες απώλειας δεδομένων. Ευνόητο είναι ότι ο δρομολογητής μετάδοσης πρέπει να γνωρίζει αυτόν τον ρυθμό αποστολής, ο οποίος είναι διαφορετικός για κάθε πελάτη. Για τον σκοπό αυτό, κάθε πελάτης που επιθυμεί να εξυπηρετηθεί από την υπηρεσία διαχείρισης δεδομένων ροής, πρέπει να μεταφέρει μια εφαρμογή από τον εξυπηρετητή και να την εγκαταστήσει τοπικά. Μέσω της εφαρμογής ο χρήστης εισάγει τα χαρακτηριστικά υλικού του πελάτη, ενώ ταυτόχρονα ένα νήμα του εξυπηρετητή αναλαμβάνει την επικοινωνία με την εφαρμογή.

Απώτερος στόχος της εφαρμογής αυτής αποτελεί η εύρεση του μέγιστου ρυθμού δεδομένων που μπορεί να δεχθεί ο πελάτης χωρίς να παρατηρούνται απώλειες πακέτων, ενώ ταυτόχρονα να είναι δυνατή η αποταμίευση των δεδομένων στην κύρια και δευτερεύουσα μνήμη, η αποκωδικοποίησή τους και η αναπαραγωγή τους στις συσκευές εξόδου του πελάτη (βλ. πειράματα ενότητας 5.4). Με άλλα λόγια το νήμα ελέγχου του εξυπηρετητή αποστέλλει δεδομένα ενός πειραματικού αντικειμένου ροής αρχίζοντας με μικρό ρυθμό μετάδοσης και αυξάνοντάς τον σταδιακά. Στην άλλη πλευρά ο πελάτης δέχεται τα δεδομένα, τα αποταμιεύει και τα παρουσιάζει καταγράφοντας συγχρόνως μετρήσεις σχετικές με τον τρέχοντα ρυθμό άφιξης των δεδομένων, το ποσοστό χρήσης του επεξεργαστή, τις παρατηρούμενες απώλειες πακέτων και τον ρυθμό αναπαραγωγής του μέσου (π.χ. καρέ ανά δευτερόλεπτο- fps). Μετά το πέρας αυτής της διαδικασίας επιστρέφεται στον εξυπηρετητή ως αποτέλεσμα ο μέγιστος ρυθμός άφιξης δεδομένων, κατά τον οποίο καταγράφηκαν μηδαμινές απώλειες πακέτων, ενώ ο ρυθμός παρουσίασης του

μέσου παρέμεινε ανεπηρέαστος. Στο παράρτημα Γ περιγράφεται με μεγαλύτερη λεπτομέρεια η εφαρμογή που αναπτύχθηκε και τρέχει στην πλευρά των πελατών.

Όλα τα αποτελέσματα των μετρήσεων αποθηκεύονται στον εξυπηρετητή για κάθε πελάτη και οργανώνονται από τον διαχειριστή αρχείων πελατών (Client Profile Manager). Η πληροφορία που περιέχεται στις εγγραφές των πελατών επιτρέπει στον δρομολογητή του συστήματος να οδηγήσει τους μηχανισμούς προανάκτησης (prefetching mechanisms) κατά την εξυπηρέτηση των αντικειμένων ροής, καθώς γνωρίζει πλέον τον μέγιστο ρυθμό μετάδοσης με τον οποίο μπορούν να αποσταλούν δεδομένα στον κάθε πελάτη και το μέγεθος των δεδομένων που κάθε πελάτης δύναται να αποταμιεύσει. Η προανάκτηση δεδομένων είναι ενδιαφέρουσα καθώς επιτρέπει στην υπηρεσία ροής να μεταφέρει δεδομένα του αντικειμένου με ρυθμούς πολλαπλάσιους των πραγματικών απαιτήσεων αναπαραγωγής. Το γεγονός αυτό προϋποθέτει μεγάλο αποταμιευτικό χώρο στους πελάτες, ενώ από την άλλη πλευρά αποφέρει σημαντικά πλεονεκτήματα, καθώς σε περιόδους μειωμένου φόρτου εργασίας οι αιτήσεις των πελατών εξυπηρετούνται πιο γρήγορα, με άμεση απόρροια την ταχύτερη αποδέσμευση των πόρων του συστήματος (επεξεργαστή, μνήμη, δίσκοι) και την αύξηση της συνολικής απόδοσης της υπηρεσίας ροής σε μεγάλο χρονικό διάστημα. Επίσης η προανάκτηση δεδομένων έχει σαν αποτέλεσμα την αποταμίευση σημαντικού μεγέθους δεδομένων τοπικά στον πελάτη, το οποίο είναι ικανό να απορροφήσει τυχόν διακυμάνσεις στην απόδοση του δικτύου ή του εξυπηρετητή, χωρίς αρνητικές επιδράσεις στην ποιότητα αναπαραγωγής του αντικειμένου ροής.

Αν υποθέσουμε ότι δεν επιχειρούμε την ενεργοποίηση των μηχανισμών προανάκτησης δεδομένων, αλλά ακολουθούμε τον κανονικό αλγόριθμο παραγωγής από τους δίσκους και σε αυτήν την περίπτωση είναι θεμιτή η αποστολή δεδομένων με τον μέγιστο ρυθμό (που ο πελάτης μπορεί να δεχθεί), ώστε να έχουμε καλύτερη διαχείριση μνήμης στον εξυπηρετητή. Στην επόμενη ενότητα εξετάζονται με μεγαλύτερη λεπτομέρεια ζητήματα διαχείρισης της προσωρινής μνήμης του συστήματος.

7.4 Αποδοτική διαχείριση μνήμης στον εξυπηρετητή ροής

7.4.1 Απαιτήσεις προσωρινής αποταμίευσης δεδομένων

Κατά την ανάπτυξη της υπηρεσίας διαχείρισης δεδομένων ροής ενσωματώσαμε ένα ευρέως γνωστό σχήμα δρομολόγησης της ανάκτησης δεδομένων. Σύμφωνα με το σχήμα αυτό η εξυπηρέτηση των αντικειμένων ροής γίνεται σε κύκλους, ενώ επιβάλλεται στην λειτουργία των δίσκων η πολιτική SCAN με στόχο την βελτιστοποίηση της απόδοσης των συσκευών αποθήκευσης με την ελαχιστοποίηση του κόστους αναζήτησης (disk seek overhead).

Η διαχείριση της μνήμης του προσωρινού αποταμιευτή αποτελεί ένα σημαντικό πρόβλημα όπως παρουσιάζεται στο [ChMol97]. Αν και σε κάθε κύκλο ανακτώνται S bytes από κάθε αντικείμενο ροής, τελικά χρειαζόμαστε διπλάσιο χώρο αποταμίευσης για να αντιμετωπίσουμε τις διαφοροποιήσεις στους χρόνους ανάκτησης των τμημάτων. Το γεγονός

αυτό συμβαίνει γιατί, εξαιτίας της πολιτικής SCAN, σε ένα κύκλο εξυπηρέτησης το τμήμα δεδομένων ενός αντικειμένου μπορεί να ανακτηθεί πρώτο, ενώ στον επόμενο κύκλο τελευταίο. Για την αντιμετώπιση του παραπάνω προβλήματος η υπηρεσία ροής δεν αρχίζει την μετάδοση δεδομένων μόλις ενός τμήμα ανακτηθεί, αντίθετα η αποστολή δεδομένων καθυστερεί μέχρι την έναρξη του επόμενου κύκλου εξυπηρέτησης. Με τον τρόπο αυτό επιτυγχάνεται αποταμίευση των απαραίτητων δεδομένων, ώστε να εξασφαλιστεί η αναπαραγωγή του μέσου ανεξάρτητα από τις χρονικές στιγμές εξυπηρέτησης του αντικειμένου στους επόμενους κύκλους. Καταλήγουμε συνεπώς σε ένα σχήμα διπλού αποταμιευτή, οπότε για την ταυτόχρονη εξυπηρέτηση N πελατών χρειαζόμαστε διαθέσιμο χώρο:

$$Buf_size = 2 \times N \times S$$

Στο [ChMol97] υπολογίζεται το μέγεθος της μνήμης που χρειάζεται για την εξυπηρέτηση N πελατών σύμφωνα με την ακόλουθη σχέση:

$$Mem_{min} = \frac{2 \times N^2 \times T_{seek} \left(\frac{TotCyl}{N} \right) \times TR \times DR}{TR - (DR \times N)}$$

όπου: TR: Ο ρυθμός ανάκτησης δεδομένων από τον δίσκο

T_{seek} (cyl): Η συνάρτηση που υπολογίζει το κόστος αναζήτησης

N: Ο αριθμός των πελατών που εξυπηρετούνται

DR: Ο ρυθμός κατανάλωσης των αντικειμένων ροής

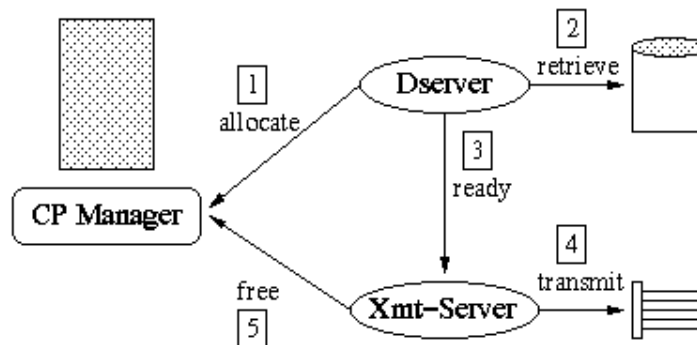
Από την παραπάνω σχέση προκύπτει ότι ο παρονομαστής αυξάνει εκθετικά, ενώ ο παρονομαστής τείνει στο μηδέν, καθώς το N αυξάνει προς το όριο TR/DR. Γίνεται επομένως κατανοητό ότι στην προσπάθεια να αξιοποιήσουμε την μέγιστη απόδοση του υποσυστήματος ανάκτησης, ώστε να εξυπηρετήσουμε περισσότερους πελάτες, το μέγεθος του αναγκαίου χώρου προσωρινής αποταμίευσης εκτινάσσεται σε υπερβολικά και απαγορευτικά επίπεδα.

Για την αντιμετώπιση αυτού του προβλήματος προτείνεται στο [ChMol97] ένα σχήμα διαμοιρασμού του χώρου αποταμίευσης, ώστε να όταν ελευθερώνεται χώρος από μία αίτηση να μπορεί να χρησιμοποιηθεί από μία άλλη, ενώ ταυτόχρονα εφαρμόζονται εναλλακτικά οι πολιτικές Fixed-Strech και GSS. Ως αποτέλεσμα αναγκάζονται να οδηγούν το σύστημα ανάκτησης σε χαμηλότερα επίπεδα απόδοσης με στόχο την μείωση των απαιτήσεων μνήμης που αναδεικνύονται σε πρωτεύων πρόβλημα στους εξυπηρετητές πολυμέσων.

Στο κεφάλαιο 7 έλαβαν χώρα πειράματα απόδοσης του εξυπηρετητή ροής της διατριβής σε ένα σταθμό εργασίας Ultra One εξοπλισμένο με ATM κάρτα δικτύου, δύο δίαυλους SCSI-2 στους οποίους εφαρμόζονται 4 μαγνητικοί δίσκοι. Η χρονική διάρκεια του κύκλου εξυπηρέτησης ήταν 1 sec και το μέγεθος του τμήματος ανάκτησης 192K.

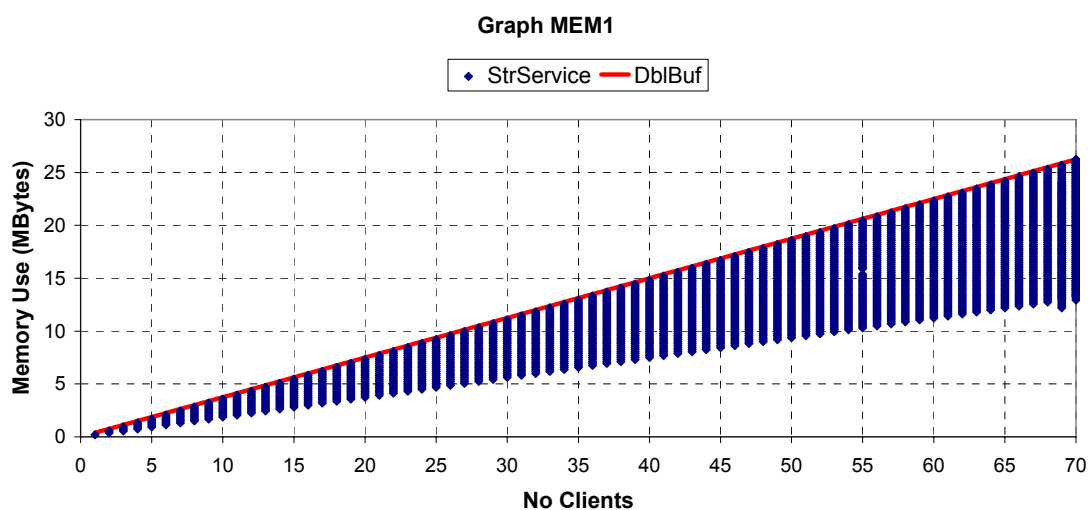
Επαναλαμβάνουμε το ίδιο πείραμα καταγράφοντας μετρήσεις στον διαχειριστή του αποταμιευτή επικοινωνίας, ώστε να αποκτήσουμε μια εικόνα της χρήσης του χώρου του

αποταμιευτή σε πραγματικές συνθήκες. Το σχήμα δέσμευσης και αποδέσμευσης προσωρινής μνήμης που ακολουθείται από την υπηρεσία ροής παρουσιάζεται στο Σχήμα 99.



Σχήμα 99: Ο μηχανισμός δέσμευσης/αποδέσμευσης χώρου στον αποταμιευτή επικοινωνίας

Τα αποτελέσματα των πειραμάτων έδειξαν ότι το σύστημα είναι σε θέση να εξυπηρετήσει ταυτόχρονα 70 πελάτες, ενώ στο ΣΧΗΜΑ η καμπύλη δέσμευσης χώρου έχει τις κορυφές της πάνω στην γραμμή του σχήματος διπλού αποταμιευτή, γεγονός που επιβεβαιώνει την ανάγκη ύπαρξης χώρου μεγέθους $2 \times 70 \times 192K = 26.25M$ για την υποστήριξη 70 πελατών.



Σχήμα 100: Χρήση προσωρινής μνήμης στην υπηρεσία ροής δεδομένων
 Στο σχήμα διακρίνεται το μέγεθος δεδομένων που είναι απαραίτητο σε συνάρτηση με τον αριθμό των ενεργών πελατών. Η υιοθέτηση σχήματος διπλού αποταμιευτή (DbIBuf) οδηγεί σε γραμμική αύξηση των απαιτήσεων μνήμης όσο εισάγονται νέες αιτήσεις. Χρησιμοποιώντας τον μηχανισμό της υπηρεσίας ροής (ΣΧΗΜΑ) παίρνουμε μετρήσεις για το μέγεθος δέσμευσης σε σχέση με τον τρέχοντα αριθμό πελατών που αναπαρίστανται με σημεία. Παρά τις παρατηρούμενες διακυμάνσεις οι υψηλότερες τιμές κείτονται πάνω στην καμπύλη DbIBuf.

Από τα πειράματα προέκυψε ότι το σύστημα δεν μπορούσε να εξυπηρετήσει περισσότερους πελάτες λόγω φαινομένων συμφόρησης στο υποσύστημα ανάκτησης. Η απόδοση των συσκευών αποθήκευσης, όπως αναφέρθηκε σε προηγούμενη υποενότητα,

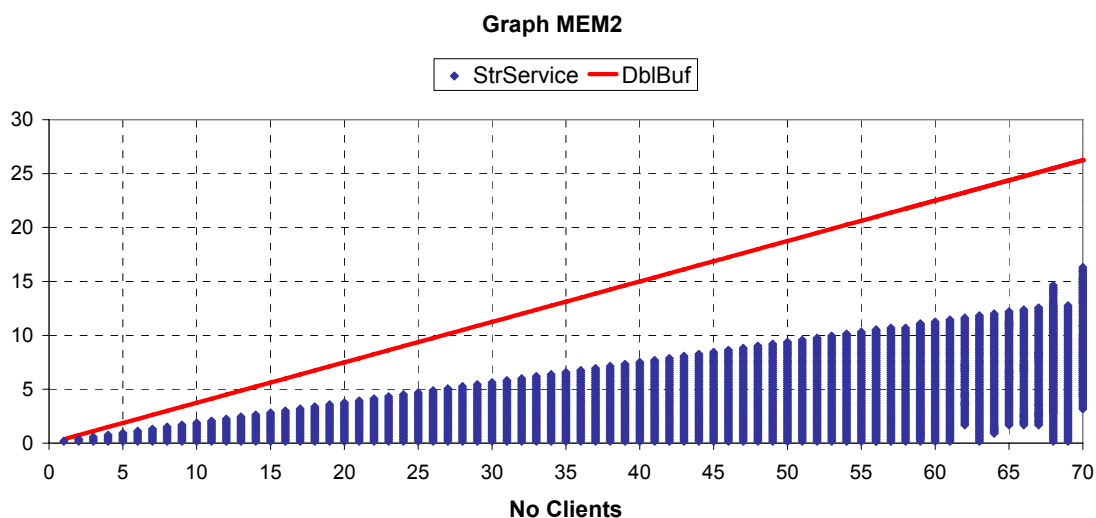
είναι δυνατόν να αυξηθεί επεκτείνοντας την διάρκεια του κύκλου εξυπηρέτησης και κατά συνέπεια το μέγεθος των τμημάτων δεδομένων που ανακτώνται.

Αυξάνοντας όμως το μέγεθος των τμημάτων στα πλαίσια του πειράματος από 192K σε 1M, το υποσύστημα ανάκτησης είναι σε θέση να υποστηρίξει μέχρι 83 πελάτες (σύμφωνα με τις μετρήσεις - Πίνακας 5). Αντίθετα το σύστημα με την τρέχουσα αλλαγή εμφανίζει δραστική μείωση της απόδοσης στους 30 πελάτες, καθώς γρήγορα εμφανίζονται, όπως είναι αναμενόμενο, προβλήματα διαθέσιμου χώρου στον αποταμιευτή επικοινωνίας (60Mbytes).

Για να ήταν εφικτή η υποστήριξη 83 πελατών, θα έπρεπε να είχαμε διαθέσιμα τουλάχιστον 166 Mbytes μνήμης, ενώ ενδεικτικό της ραγδαίας αύξησης των απαιτήσεων προσωρινής μνήμης όσο προσεγγίζουμε τα όρια δυνατοτήτων των συσκευών αποθήκευσης είναι το γεγονός ότι με την βέλτιστη αξιοποίηση του τρέχοντος υποσυστήματος ανάκτησης – που σύμφωνα με τις μετρήσεις δύναται να υποστηρίξει 90 πελάτες με τμήμα δεδομένων μεγέθους 8M- θα χρειαζόταν τουλάχιστον 1440 Mbytes μόνο για τις ανάγκες του αποταμιευτή επικοινωνίας.

7.4.2 Μείωση των απαιτήσεων χώρου σε προσωρινή μνήμη

Μια λύση στο μεγάλο πρόβλημα των απαιτήσεων σε προσωρινή μνήμη είναι η αποδοχή σχημάτων που υποχρησιμοποιούν το υποσύστημα ανάκτησης, αλλά από την άλλη πλευρά βελτιώνουν την διαχείριση του χώρου στον αποταμιευτή επικοινωνίας ([ChMol97].



Σχήμα 101: Χρήση προσωρινής μνήμης στην υπηρεσία ροής δεδομένων
Από την παρατήρηση του σχήματος διακρίνουμε την σημαντική μείωση δέσμευσης αποταμιευτικού χώρου στον εξυπηρετητή ροής όταν η καθυστέρηση έναρξης της κατανάλωσης (για λόγους κάλυψης των διαφοροποιήσεων στους χρόνους ανάκτησης των τμημάτων) λαμβάνει χώρα στην πλευρά πελατών με αρκετό χώρο αποθήκευσης δεδομένων (CIBuf).

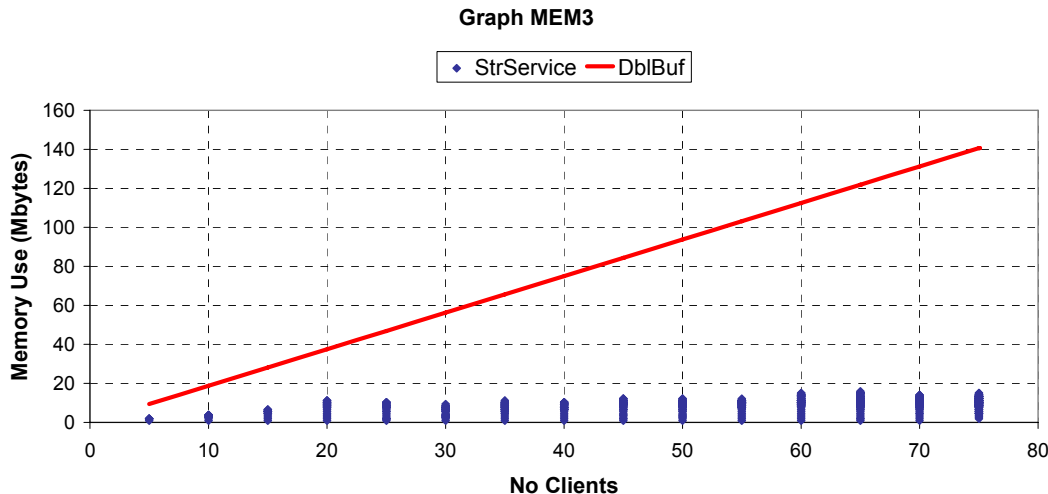
Μια άλλη εναλλακτική λύση είναι να εκμεταλλευτούμε τις δυνατότητες των πελατών σε αποθήκευση και λήψη δεδομένων, με απώτερο στόχο να μειώσουμε τις απαιτήσεις προσωρινής αποταμίευσης στην πλευρά του εξυπηρετητή. Η ιδέα αυτή βασίζεται στο γεγονός ότι πλέον είναι συνηθισμένο φαινόμενο τα συστήματα πελατών να αποτελούνται από ιδιαίτερα ισχυρό εξοπλισμό με μεγάλες δυνατότητες αποθήκευσης δεδομένων σε

πρωτεύουσα και δευτερεύουσα μνήμη. Από την άλλη πλευρά οι συνεχείς βελτιώσεις στην ταχύτητα των δικτύων προετοιμάζουν την μελλοντική εικόνα των διανεμημένων συστημάτων με ισχυρούς πελάτες, οι οποίοι θα έχουν πρόσβαση μέσω ταχύτατων δικτύων στους παροχείς υπηρεσιών.

Ένα πρώτο βήμα είναι αντί η υπηρεσία ροής να καθυστερεί την έναρξη της μετάδοσης δεδομένων για το πρώτο τμήμα του αντικειμένου μέχρι την λήξη του τρέχοντος κύκλου, να μεταβιβάζει αυτήν την υπευθυνότητα στους πελάτες που έχουν αρκετό χώρο αποταμίευσης. Σε αυτή την περίπτωση ο εξυπηρετητής ροής μαζί με τα πρώτα δεδομένα θα πρέπει να αποστείλει και ένα χρονικό σημείο που θα ενημερώνει τον πελάτη για τον χρόνο έναρξης της αναπαραγωγής του μέσου.

Εφαρμόζοντας αυτήν την τεχνική (ClBuf) στην υπηρεσία ροής καταγράφουμε μετρήσεις στον διαχειριστή του αποταμιευτή επικοινωνίας, οι οποίες εμφανίζονται γραφικά στο Σχήμα 101. Είναι πλέον εμφανές ότι εκμεταλλευόμενοι τις δυνατότητες αποταμίευσης των πελατών μπορούμε άμεσα να μειώσουμε στο μισό τις απαιτήσεις σε χώρο προσωρινής μνήμης στον εξυπηρετητή.

Ακολουθώντας την ίδια λογική, αν εκτός από τους αποταμιευτές των πελατών προσπαθήσουμε να εκμεταλλευτούμε την υπολογιστική ισχύ τους και τον δικτυακό τους εξοπλισμό μπορούμε να επιτύχουμε δραστική μείωση στον απαιτούμενο χώρο αποταμίευσης στην πλευρά του εξυπηρετητή ροής, αφήνοντας επομένως ανοιχτό το πεδίο για πλήρη αξιοποίηση του υποσυστήματος ανάκτησης και των υπόλοιπων πόρων.



Σχήμα 102 Χρήση προσωρινής μνήμης στην υπηρεσία ροής δεδομένων

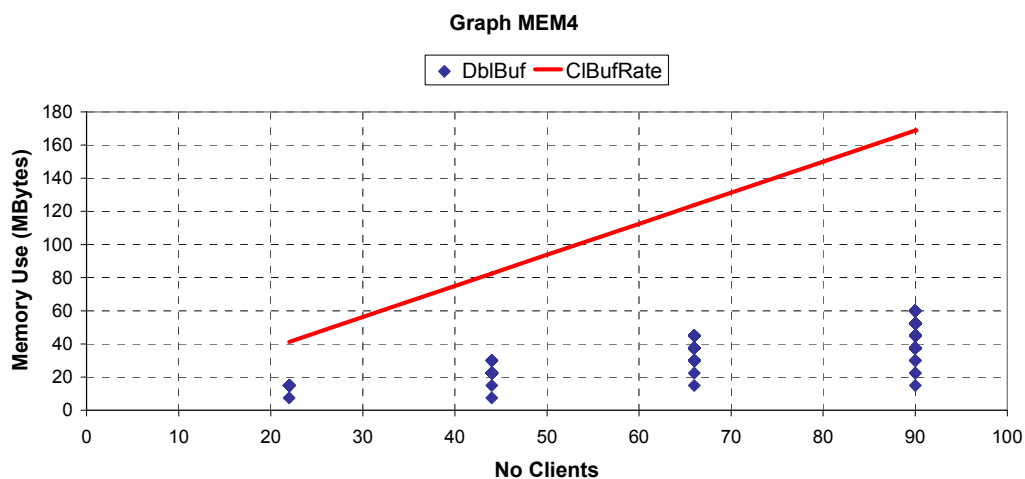
Η άμεση αποστολή των δεδομένων των διαθέσιμων τμημάτων των αντικειμένων στον αποταμιευτή επικοινωνίας με τον μέγιστο αριθμό που μπορούν να λαμβάνουν οι πελάτες (100Mbps), έχει σαν αποτέλεσμα την εντυπωσιακή μείωση του απαιτούμενου χώρου αποταμίευσης στην πλευρά του εξυπηρετητή ροής. Χαρακτηριστικό είναι ότι ακολουθώντας το κανονικό σχήμα εξυπηρέτησης των αντικειμένων ροής και προσεγγίζοντας τους 80 πελάτες, απαιτούνται τουλάχιστον 160 Mbytes στο αποταμιευτή επικοινωνίας (DbIBuf), αντίθετα υιοθετώντας το προτεινόμενο σχήμα (ClBufRate) οι απαιτήσεις περιορίζονται στο επίπεδο των 20 Mbytes.

Έτσι ο εξυπηρετητής μαζί με τον μηχανισμό επιβολής άμεσης μετάδοσης μόλις τα τμήματα δεδομένων γίνονται διαθέσιμα, προσπαθεί να αποστείλει τα δεδομένα αυτά με τον

μέγιστο ρυθμό που ο πελάτης μπορεί να δεχθεί χωρίς προβλήματα, ώστε να ελευθερωθεί όσο το δυνατόν ταχύτερα ο αντίστοιχος χώρος που έχει δεσμευθεί με αντικειμενικό στόχο την βελτιστοποίηση της διαχείρισης μνήμης στον αποταμιευτή επικοινωνίας. (Αξίζει να σημειωθεί ότι στην πραγματικότητα ο ρυθμός αποστολής στους πελάτες μπορεί να είναι μικρότερος του μέγιστου που μπορούν να δεχθούν, καθώς ο μέγιστος ρυθμός αποστολής δεδομένων του εξυπηρετητή διαιρείται με τον αριθμό των τμημάτων που ταυτόχρονα πρέπει να μεταδοθούν κάθε στιγμή).

Στο παρακάτω πείραμα υποθέτουμε ότι έχουμε διαθέσιμο ένα δίκτυο από πελάτες με δυνατότητα λήψης 100 Mbps και αρκετό χώρο αποταμίευσης σε πρωτεύουσα και δευτερεύουσα μνήμη. Εφαρμόζοντας τις τεχνικές που περιγράφηκαν προηγουμένως τρέχουμε τον εξυπηρετητή ροής και αυξάνουμε συνεχώς τους πελάτες, ώστε να ερευνήσουμε τα όρια του συστήματος. Τα αποτελέσματα έδειξαν ότι το σύστημα κατάφερε να ικανοποιήσει την εξυπηρέτηση 79 πελατών με 100% χρήση του επεξεργαστή, 97.1% αξιοποίηση του δικτυακού εξοπλισμού και 88.2% αξιοποίηση του υποσυστήματος ανάκτησης (τα ποσοστά προκύπτουν από τις μέγιστες πειραματικές επιδόσεις).

Οι μετρήσεις που καταγράφηκαν στον διαχειριστή του αποταμιευτή επικοινωνίας δείχνουν δραστική μείωση του απαιτούμενου χώρου προσωρινής μνήμης σε σύγκριση με το κανονικό σχήμα εξυπηρέτησης των ροών δεδομένων (Σχήμα 102). Όπως αποδεικνύεται πειραματικά, εμπόδιο στην μεγαλύτερη αύξηση της απόδοσης αποτελεί η επεξεργαστική ισχύς και η δικτυακή υποδομή του τρέχοντος συστήματος, με αποτέλεσμα να οδηγούμαστε σε υποαξιοποίηση του υποσυστήματος ανάκτησης.

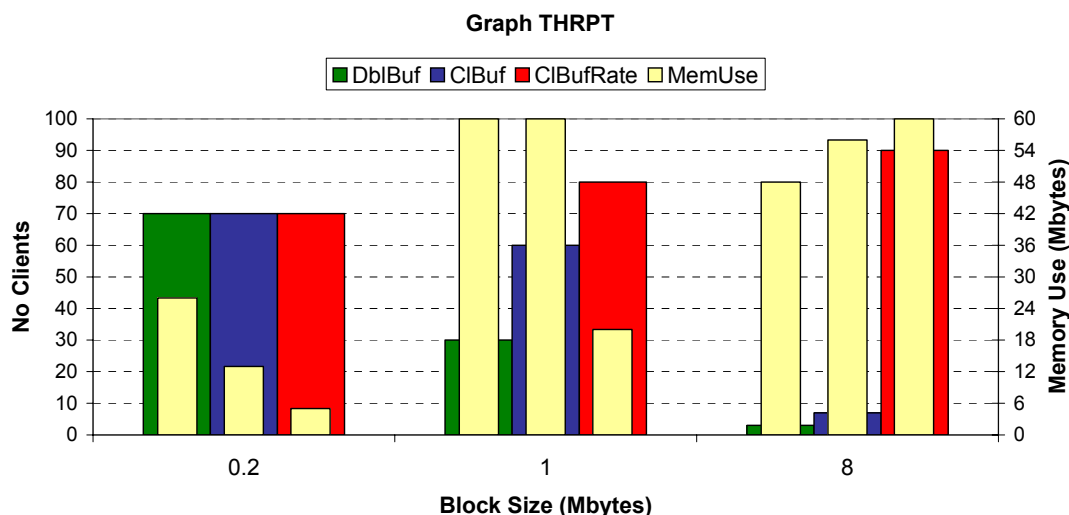


Σχήμα 103: Χρήση προσωρινής μνήμης στην υπηρεσία ροής δεδομένων
Αυξάνοντας το τμήμα ανάκτησης δεδομένων σε ~8Mbytes επιτυγχάνουμε καλή αξιοποίηση των δίσκων του συστήματος, ενώ επιβάλλοντας το προτεινόμενο σχήμα δρομολόγησης (ClBufRate) είναι δυνατή η λειτουργία του συστήματος καθώς οι απαιτήσεις μνήμης περιορίζονται στα 60Mbytes αντί των 168Mbytes που θα ήταν αναγκαία για την εφαρμογή του κανονικού σχήματος εξυπηρέτησης με διπλό αποταμιευτή (DbIBuf).

Στην συνέχεια υποθέτουμε ότι τα προβλήματα αυτά επιλύονται με ισχυρότερο επεξεργαστή ή πολυεπεξεργαστικό σύστημα και πολλαπλές συσκευές δικτύου. Επιβάλλουμε την δημιουργία τμημάτων δεδομένων μεγέθους 8M, που πειραματικά επιφέρει

την μεγαλύτερη απόδοση (Πίνακας 5) και παρακολουθούμε τον διαχειριστή του αποταμιευτή επικοινωνίας, συγκεντρώνοντας μετρήσεις οι οποίες παρουσιάζονται στο Σχήμα 103.

Στο Σχήμα 104 γίνεται μια σύγκριση των αποδόσεων (throughput) που επιτυγχάνονται από την υπηρεσία διαχείρισης ροής με διαφορετικά μεγέθη τμημάτων δεδομένων, υποθέτοντας ότι δεν έχουμε προβλήματα μετάδοσης. Η υπηρεσία ροής τρέχει σε σταθμό Ultra One, ενώ ο αποταμιευτής επικοινωνίας έχει διαθέσιμα 60 Mbytes μνήμης.



Σχήμα 104: Αποδόσεις της υπηρεσίας διαχείρισης δεδομένων ροής

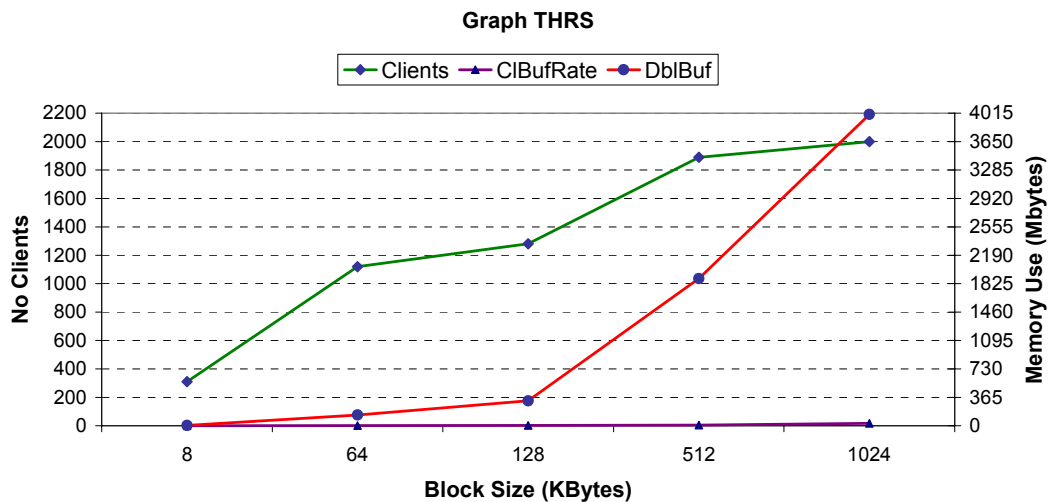
Στο γράφημα παρουσιάζονται οι πειραματικές αποδόσεις του εξυπηρετητή ροής για διαφορετικά μεγέθη τμημάτων δεδομένων (0.2K, 1M, 8M). Στην υπηρεσία ροής εφαρμόζονται τρεις διαφορετικές τεχνικές δρομολόγησης: το κανονικό σχήμα εξυπηρέτησης (DbIBuf), η τεχνική άμεσης μετάδοσης των τμημάτων επιβάλλοντας αποταμίευση στον πελάτη (ClIBuf) και τέλος ένα σχήμα (ClIBufRate) άμεσης μετάδοσης με τον μέγιστο ρυθμό που οι πελάτες μπορούν να δεχθούν (100Mbps). Τα αποτελέσματα δείχνουν ότι αυξάνοντας το μέγεθος του τμήματος ανάκτησης (ώστε να αυξήσουμε την απόδοση των δίσκων) οι δύο πρώτες τεχνικές (DbIBuf, ClIBuf) εμφανίζουν τελικά μείωση στην μέγιστη απόδοση λόγω περιορισμών στην δέσμευση προσωρινής μνήμης. Θετική συμπεριφορά παρουσιάζει μόνο το σχήμα ClIBufRate το οποίο και επιτυγχάνει να εξυπηρετήσει ικανοποιητικά 90 πελάτες με τμήμα ανάκτησης 8M.

Στο τέλος του 6ου κεφαλαίου όταν εξετάστηκε η συμπεριφορά του συστήματος στην περίπτωση που πρέπει να εξυπηρετήσει ένα μεγάλο αριθμό από αντικείμενα ροής με μικρές απαιτήσεις (π.χ. αντικείμενα ήχου), γρήγορα αναδείχθηκε το γεγονός ότι η προσωρινή μνήμη αποταμίευσης αποτελεί σοβαρό εμπόδιο αύξησης της απόδοσης του συστήματος. Έτσι ο εξυπηρετητής ροής αναγκάζεται να λειτουργήσει σε πολύ χαμηλότερα επίπεδα απόδοσης από αυτά που μπορεί να υποστηρίξει το υποσύστημα μετάδοσης.

Αν υποθέσουμε ότι έχουμε αποθηκεύσει στην υπηρεσία ροής αντικείμενα ήχου με απαιτήσεις αναπαραγωγής 64 Kbps, τότε το Σχήμα 105 παρουσιάζει την σχέση της απόδοσης του συστήματος (throughput) με το μέγεθος της διαθέσιμης προσωρινής μνήμης. Είναι εμφανές ότι χρησιμοποιώντας το κανονικό σχήμα εξυπηρέτησης για να επιτύχουμε μια ικανοποιητική απόδοση (σε σχέση με τις δυνατότητες του υποσυστήματος ανάκτησης) χρειάζεται να είναι διαθέσιμα υπερβολικά ποσά προσωρινής μνήμης.

Αν υποθέσουμε ότι η υπηρεσία ροής καλείται να εξυπηρετήσει ένα μεγάλο αριθμό από πελάτες με καλή δικτυακή υποδομή (π.χ. δυνατότητα λήψης δεδομένων με ρυθμό 10 Mbps)

και κάθε πελάτης έχει αρκετή διαθέσιμη μνήμη (>2Mbytes), τότε ακολουθώντας τον προτεινόμενο μηχανισμό δρομολόγησης της υπηρεσίας ροής είναι εφικτή η προσέγγιση της μέγιστης απόδοσης με ταυτόχρονο περιορισμό των απαιτήσεων αποταμίευσης σε χαμηλά επίπεδα (<20 Mbytes μέγεθος αποταμίευσης για 2000 πελάτες) (Σχήμα 105 CIBufRate). Αξίζει να σημειωθεί ότι η απόδοση στο κανονικό σύστημα με τους περιορισμούς στον εξοπλισμό δικτύου και τον επεξεργαστή μειώνεται στους 1900 πελάτες.

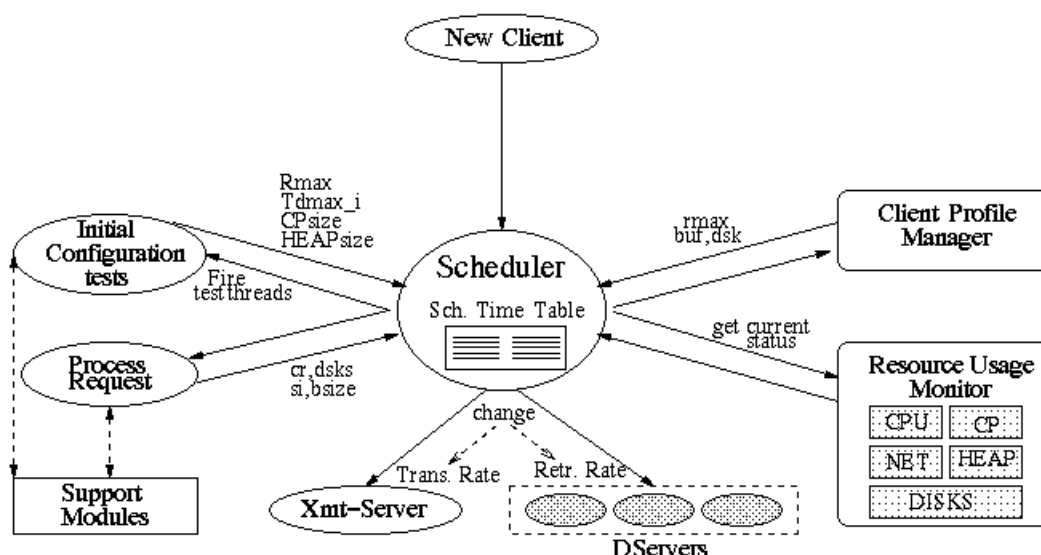


Σχήμα 105: Απόδοση του εξυπηρετητή ροής και απαιτήσεις προσωρινής αποταμίευσης

7.5 Ο αυτοπροσαρμοζόμενος μηχανισμός δρομολόγησης

Στο Σχήμα 106 παρουσιάζονται σε γενικές γραμμές οι αλληλεπιδράσεις του κεντρικού δρομολογητή της υπηρεσίας ροής με τα υπόλοιπα τμήματα λογισμικού και οι ενέργειες που λαμβάνουν χώρα. Στο σχήμα επίσης διακρίνονται ο διαχειριστής αρχείων των πελατών και το τμήμα εκείνο του λογισμικού που είναι υπεύθυνο για την συνεχή παρακολούθηση της χρήσης των πόρων του υπολογιστικού συστήματος (Resource Monitor).

Όπως αναφέρθηκε στην προηγούμενη υποενότητα, κατά την εγκατάσταση της υπηρεσίας διαχείρισης δεδομένων ροής ο κεντρικός δρομολογητής δημιουργεί μια σειρά από νήματα που ελέγχουν τις συσκευές υποστήριξης του εξυπηρετητή και την απόδοση της πρόσβασης των πόρων του υπολογιστικού συστήματος. Με την βοήθεια του αρχείου των συσκευών (Device script) και του λογισμικού υποστήριξης τα νήματα αυτά επιστρέφουν τελικά στον δρομολογητή την απαραίτητη πληροφορία στην οποία εμπεριέχεται ο πίνακας των δυνατών ρυθμών μετάδοσης από τον εξυπηρετητή, η μέγιστη απόδοση των αποθηκευτικών μέσων ανάλογα με το μέσο μέγεθος των τμημάτων ανάκτησης και τα αντίστοιχα ποσοστά δέσμευσης της επεξεργαστικής ισχύος.



Σχήμα 106: Το σχήμα δυναμικής δρομολόγησης

Όλα αυτά τα δεδομένα οδηγούνται και οργανώνονται στο ελεγκτή των πόρων του συστήματος. Επιπρόσθετα τα αρχικά αυτά νήματα ελέγχου ενημερώνουν τον κεντρικό δρομολογητή για το διαθέσιμο μέγεθος του προσωρινού αποταμιευτή του συστήματος (Heap) και του αποταμιευτή επικοινωνίας (Com. Pool). Μετά το πέρας της φάσης εγκατάστασης του λογισμικού και των αρχικών ρυθμίσεων, η κεντρική οντότητα δρομολόγησης έχει πλέον μια γενική εικόνα των επιδόσεων που μπορούν να επιτευχθούν με τις υπάρχουσες ρυθμίσεις.

Στην συνέχεια κάθε πελάτης που επιθυμεί να τύχει εξυπηρέτησης από το σύστημα, πρέπει να ακολουθήσει την απαραίτητη διαδικασία εγγραφής στο σύστημα. Έτσι ο πελάτης, όπως έχει ήδη αναφερθεί, τρέχει τοπικά μια ειδική εφαρμογή μέσω της οποίας ο δρομολογητής δέχεται τα τεχνικά χαρακτηριστικά του πελάτη (είδος επεξεργαστή, μέγεθος προσωρινής αποταμίευσης, μέγεθος δευτερεύουσας μνήμης, ρυθμός εγγραφής στους δίσκους, δικτυακή υποδομή, εξοπλισμό αναπαραγωγής δεδομένων ροής κ.α.), ενώ το πιο σημαντικό πλεονέκτημα είναι ότι με την βοήθεια της εφαρμογής διαπιστώνονται σε πραγματικό περιβάλλον οι δυνατές επιδόσεις του πελάτη με τις τρέχουσες ρυθμίσεις. Συνεπώς, όταν η διαδικασία εγγραφής για κάθε πελάτη ολοκληρώνεται, ο διαχειριστής πελατών (Client Profile Manager) ενημερώνεται με νέα στοιχεία, όπως κωδικός νέου πελάτη, μέγιστη δυνατότητα ρυθμού λήψης δεδομένων και διαθέσιμος χώρος αποταμίευσης δεδομένων στην πρωτεύουσα και δευτερεύουσα μνήμη.

Στο Σχήμα 106 μπορεί κανείς να διακρίνει γραφικά τις αλληλεπιδράσεις που συμβαίνουν κατά την είσοδο στην υπηρεσία ροής μιας νέας αίτησης αναπαραγωγής ενός μέσου (αναλυτική περιγραφή έγινε στο κεφάλαιο 6). Ο κεντρικός δρομολογητής αποτελεί, όπως είναι φυσικό, τον τελικό αποδέκτη της νέας αίτησης. Στην συνέχεια ο δρομολογητής αναλύοντας τα δεδομένα της αίτησης προκαλεί την δημιουργία ενός νήματος για την επεξεργασία της. Το νήμα επεξεργασίας σε συνεργασία με τα τμήματα λογισμικού υποστήριξης, συγκεντρώνει τις κατάλληλες πληροφορίες, οι οποίες τελικά επιστρέφονται

στον κεντρικό δρομολογητή. Οι πληροφορίες αυτές αναφέρονται στον απαραίτητο ρυθμό κατανάλωσης του μέσου, τον αριθμό και τα αναγνωριστικά των δίσκων στους οποίους έχει διασπαστεί η αποθήκευση του αντικειμένου, το μέγεθος του τμήματος δεδομένων αποθήκευσης, καθώς επίσης και στον αντίστοιχο πίνακα πρόσβασης του αντικειμένου ροής. Ο δρομολογητής με βάση το αναγνωριστικό του πελάτη ανακτά τα στοιχεία που χαρακτηρίζουν τον πελάτη από τον διαχειριστή πελατών. Σε αυτό ο σημείο ελέγχεται αν ο διαθέσιμος χώρος στην προσωρινή μνήμη του συστήματος (heap) είναι αρκετός για τις ανάγκες επεξεργασίας της αίτησης, ώστε να καταστεί εφικτή η συνέχιση της εξυπηρέτησης της αίτησης.

Έχοντας συγκεντρώσει όλα αυτά τα δεδομένα ο κεντρικός δρομολογητής είναι πλέον σε θέση να οδηγήσει την ροή δεδομένων της νέας αίτησης και να πάρει αποφάσεις αλλάζοντας δυναμικά της παραμέτρους εξυπηρέτησης των υπόλοιπων ενεργών ροών δεδομένων, όπως θα δούμε στην συνέχεια. Έτσι με βάση τα στοιχεία που υπάρχουν, το δρομολογητής υπολογίζει τον ιδανικό ρυθμό μετάδοσης που πρέπει να εφαρμόσει για το συγκεκριμένο νέο πελάτη, σύμφωνα με τον μαθηματικό τύπο που προκύπτει από την ακόλουθη διαδικασία. Αν υποθέσουμε ότι μεταδίδουμε τα δεδομένα του αντικειμένου ροής με ρυθμό μεγαλύτερο από τις απαιτήσεις κατανάλωσης, τότε για την αποφυγή προβλημάτων θα πρέπει ο χρόνος αναπαραγωγής του μέσου να είναι μικρότερος ή ίσος με τον χρόνο που μεσολαβεί μέχρι να γεμίσουν οι διαθέσιμοι χώροι αποταμίευσης του πελάτη λόγω της διαφοράς στους ρυθμούς κατανάλωσης και μετάδοσης. Συνεπώς πρέπει να ισχύει:

$$\frac{strm_size}{Cr} \leq \frac{Buf_size + Dsk_size}{R_{send} - Cr} \Rightarrow$$

$$R_{send} \leq \frac{Cr(Buf_size + Dsk_size + strm_size)}{strm_size}$$

όπου *strm_size* : το συνολικό μέγεθος του αντικειμένου ροής

Cr : ο ρυθμός κατανάλωσης δεδομένων

Buf_size : το διαθέσιμο μέγεθος προσωρινής μνήμης

Dsk_size : το διαθέσιμο μέγεθος αποθήκευσης σε τοπικούς δίσκους

Rsend : ο ρυθμός αποστολής δεδομένων

Με βάση τον παραπάνω τύπο προκύπτει ο ρυθμός μετάδοσης δεδομένων για τον νέο πελάτη σύμφωνα με τον σχέση:

$$R_{cl} = \frac{Cr(Buf_size + Dsk_size + strm_size)}{strm_size} \quad \text{Αν } R_{max} > R_{send}$$

$$R_{MAX} \quad \text{Διαφορετικά}$$

Σχέση 1

Πρέπει να σημειωθεί ότι η παραπάνω σχέση υπολογίζει τον μέγιστο ρυθμό δεδομένων που ο πελάτης είναι σε θέση να δεχθεί συνεχόμενα σε ολόκληρη την διάρκεια εξυπηρέτησης, ενεργοποιώντας τους μηχανισμούς προανάκτησης στον εξυπηρετητή ροής.

Αν ο δρομολογητής του συστήματος αποφασίσει να διατηρήσει τον κανονικό ρυθμό ανάκτησης δεδομένων από του δίσκους, τότε η μετάδοση δεδομένων μπορεί να γίνει με ρυθμό $R_{cl\max}$ (εφόσον ο χώρος αποταμίευσης στον πελάτη είναι αρκετός).

Αν το μέγεθος το τμήματος μετάδοσης S είναι αρκετά μεγάλο ώστε ο πελάτης να μην διαθέτει χώρο $2S$ στην πρωτεύουσα μνήμη του για να υποστηρίξει την αναπαραγωγή του μέσου, θα πρέπει να αποθηκεύσει δεδομένα και στον δίσκο που τυχόν διαθέτει. Έτσι ο μέγιστος ρυθμός λήψης προκύπτει από την ακόλουθη σχέση :

$$\frac{R_{cl\max}}{TR} \quad \text{Αν} \quad TR > R_{cl\max} \\ \text{Διαφορετικά}$$

Σχέση 2

Ο διαχειριστής αρχείων των πελατών συνεπώς, πρέπει να διατηρεί τις τιμές του ρυθμού λήψης δεδομένων για κάθε πελάτη σε κάθε περίπτωση. Στην συνέχεια ακολουθεί ένα παράδειγμα του πίνακα στοιχείων που αποθηκεύονται για κάθε πελάτη.

Περίπτωση	Εφικτός ρυθμός λήψης δεδομένων
$S < 16 \text{ Mbytes}$	$R1 \ (R_{cl\max})$
$S > 16 \text{ Mbytes}$	$R2 \dots (\text{Σχέση 2})$
Προανάκτηση (prefetching)	$R3 \dots (\text{Σχέση 1})$

Το επόμενο βήμα του δρομολογητή είναι να εξετάσει σε συνεργασία με τον ελεγκτή των πόρων του συστήματος (Resource Monitor), αν οι δίσκοι, στους οποίους έχει αποθηκευτεί το αντικείμενο ροής, μπορούν να αντεπεξέλθουν στην αύξηση του φόρτου εργασίας που θα προκύψει από την επεξεργασία της νέας αίτησης.

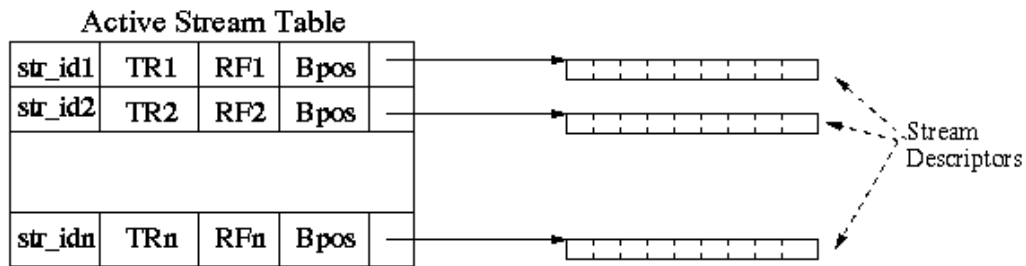
Η αποθήκευση των αντικειμένων ροής στον εξυπηρετητή γίνεται, κατά κανόνα, ακολουθώντας την coarse striping τεχνική, ώστε σε κάθε κύκλο να μπορεί να ανακτηθεί ένα τμήμα δεδομένων από κάθε δίσκο με κυκλική σειρά, μέσα από ένα σύνολο δίσκων που καθορίζεται από τον αντίστοιχο φάκελο του αντικειμένου. Έτσι ο διαχειριστής τοποθέτησης δεδομένων (βλ. Σχήμα 28) καθορίζει το μέγεθος του τμήματος ανάκτησης να είναι $Cr * \text{SYS_ROUND}$.

Σε αυτό το στάδιο της ανάπτυξης επιτρέπεται η προανάκτηση δεδομένων με ρυθμούς που αποτελούν ακέραια πολλαπλάσια του ρυθμού κατανάλωσης του αντικειμένου. Έτσι επιτρέπουμε σε πελάτες με αυξημένες δυνατότητες σε αποταμίευση και λήψη δεδομένων, να δέχονται ταχύτερα τα δεδομένα (για λόγους που σχολιάστηκαν στην ενότητα 7.3) εκμεταλλευόμενοι τον μηχανισμό παράλληλης ανάκτησης (δηλαδή επιβάλλοντας fine striping τεχνική).

Ο δρομολογητής οδηγεί την εξυπηρέτηση των αιτήσεων με την βοήθεια του πίνακα ενεργών ροών και σε συνεργασία με τον διαχειριστή πελατών και του ελεγκτή των πόρων του συστήματος. Ο πίνακας των ενεργών ροών παρουσιάζεται στο Σχήμα 107. Κάθε εγγραφή του πίνακα αυτού περιέχει το αναγνωριστικό του αντικειμένου ροής που

εξυπηρετείται, τον τρέχοντα ρυθμό μετάδοσης δεδομένων στον πελάτη (TR_i) και τον παράγοντα ανάκτησης (retrieval factor –RF). Αυτός ο παράγοντας καθορίζει τον ακέραιο πολλαπλασιαστή του ρυθμού κατανάλωσης του αντικειμένου, με τον οποίο προκύπτει ο τρέχων ρυθμός ανάκτησης από τους δίσκους ($RF \cdot Cr$).

Τέλος οι εγγραφές του πίνακα ενεργών ροών περιέχουν δείκτες στις δομές των περιγραφών των αντίστοιχων αντικειμένων ροής, που όπως έχει περιγραφεί στο κεφάλαιο 4 διατηρούνται από το υποσύστημα αποθήκευσης στην προσωρινή μνήμη του συστήματος (heap).



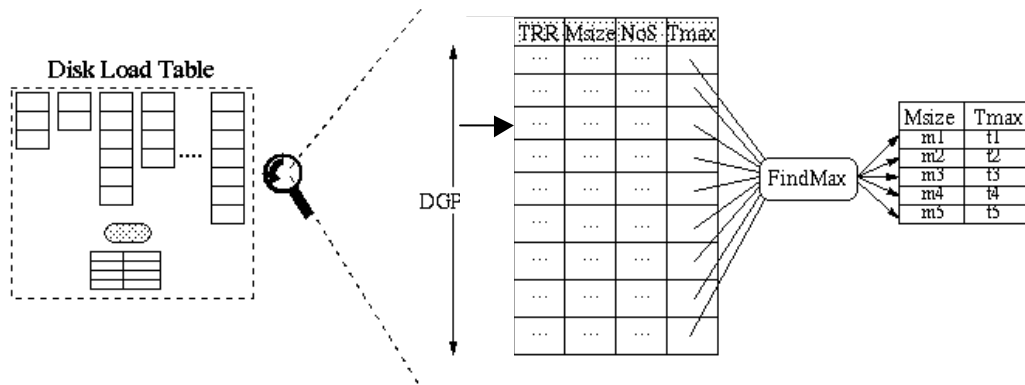
Σχήμα 107: Ο πίνακας ενεργών ροών

Η δομή των ενεργών αντικειμένων ροής αποτελεί ένα πίνακα κατακερματισμού (hashing table) με βάση το αναγνωριστικό του αντικειμένου. Κάθε αντικείμενο ροής που εξυπηρετείται από το σύστημα χαρακτηρίζεται από τον ρυθμό μετάδοσης των δεδομένων στον πελάτη (TR), το παράγοντα ανάκτησης (RF) και το στάδιο της εξυπηρέτησης ($Bpos$, επόμενο τμήμα δεδομένων).

Χρησιμοποιώντας τον πίνακα ενεργών ροών ο κεντρικός δρομολογητής έχει άμεση εικόνα κάθε χρονική στιγμή των παραμέτρων που χαρακτηρίζουν κάθε διαφορετική ροή δεδομένων που είναι ενεργή στο σύστημα. Έτσι μεταβάλλοντας ο κεντρικός δρομολογητής δυναμικά τις παραμέτρους των ενεργών αντικειμένων, αυτόματα αλλάζει και ο τρόπος εξυπηρέτησής τους από την υπηρεσία ροής.

Σε κάθε κύκλο εξυπηρέτησης ο ελεγκτής πραγματικού χρόνου μέσω του δρομολογητή εγγράφει τις υποαιτήσεις δίσκου στον κατάλληλο χώρο επικοινωνίας (DSR-Channel). Ο κεντρικός δρομολογητής δημιουργεί τις υποαιτήσεις αυτές χρησιμοποιώντας τον πίνακα των ενεργών ροών με τους αντίστοιχους περιγραφείς αντικειμένων.

Ο ελεγκτής των πόρων του συστήματος διατηρεί επίσης μια δομή, με στόχο την παρακολούθηση του φόρτου εργασίας των δίσκων. Η δομή αυτή καλείται πίνακας φόρτου εργασίας των δίσκων (Disk Load Table) και περιγράφεται στο επόμενο σχήμα (Σχήμα 108).



Σχήμα 108: Πίνακας φόρτου εργασίας δίσκων

Ο κύκλος του συστήματος είναι σταθερός και κοινός για όλους τους εξυπηρετητές δίσκου. Η περίοδος εξυπηρέτησης ενός αντικειμένου ροής από τον ίδιο δίσκο εξαρτάται από το εύρος των δίσκων στους οποίους έχει διασπαστεί η αποθήκευση, το μέγεθος του τμήματος δεδομένων ανάκτησης, καθώς και από τον ρυθμό ανάκτησης από τους δίσκους, όπως προκύπτει από την ακόλουθη σχέση :

$$\text{Period} = \frac{\text{no_disks}}{\text{RF}}$$

Σχέση 3

Το μέγεθος του δρομολογίου μπορεί να ποικίλει από δίσκο σε δίσκο καθώς είναι δυναμικά προσαρμοζόμενο στο ελάχιστο κοινό πολλαπλάσιο των περιόδων όλων των αντικειμένων ροής που εξυπηρετούνται από τον δίσκο και καλείται κοινή περίοδος δίσκου (Disk Global Period- DGP). Ένας δείκτης (Current Round Pointer- CPR) σε κάθε δρομολόγιο υποδεικνύει την εγγραφή που αντιστοιχεί στον τρέχοντα κύκλο που διανύεται από το σύστημα και προκύπτει από την ακόλουθη ισότητα:

$$\text{CRP} = \text{SYS_ROUND} \bmod \text{DGP}$$

Κάθε φορά που ζητείται η εξυπηρέτηση ενός νέου αντικειμένου ροής, ο κεντρικός δρομολογητής ζητάει από τον ελεγκτή των πόρων να ελέγξει αν είναι δυνατή η εξυπηρέτηση της νέας αίτησης, παρέχοντάς του ταυτόχρονα πληροφορία σχετική με το τμήμα δεδομένων ανάκτησης του αντικειμένου, τα αναγνωριστικά των δίσκων στους οποίους έχει διασπαστεί η αποθήκευση και τον τρέχοντα παράγοντα ανάκτησης. Στην συνέχεια ο ελεγκτής των πόρων υπολογίζοντας την περίοδο εξυπηρέτησης του αντικειμένου για κάθε δίσκο, προσθέτει στις γραμμές του πίνακα που αντιπροσωπεύουν τους κύκλους κατά τους οποίους το νέο αντικείμενο πρέπει να εξυπηρετείται σύμφωνα με την περίοδό του, τον ρυθμό ανάκτησης του αντικειμένου (1η στήλη). Ταυτόχρονα ενημερώνεται (2η στήλη) ο νέος μέσος όρος των τμημάτων που ανακτούνται σε εκείνο τον κύκλο ($\text{newMsize} = \frac{\text{NoStreams} \cdot \text{Msize} + \text{bl_size}}{\text{NoStreams} + 1}$). Με την βοήθεια της κατάλληλης συνάρτησης βρίσκουμε την μέγιστη απόδοση του δίσκου ($T_{\text{max_rate}}$) για τον πλησιέστερο μέσο όρο μεγέθους τμημάτων με χρήση του πίνακα που προέκυψε πειραματικά από την διαδικασία εγκατάστασης του συστήματος (Πίνακας 5). Αν ο αριθμός

που υπολογίστηκε στην δεύτερη στήλη του πίνακα παρακολούθησης του φόρτου των δίσκων είναι μεγαλύτερος του αντίστοιχου T_{max_rate} , αυτό σημαίνει ότι τον δεδομένο χρονικό κύκλο ο συγκεκριμένος δίσκος πιθανότατα δεν θα καταφέρει να αντεπεξέλθει στις απαιτήσεις ανάκτησης, δημιουργώντας έτσι ανεπιθύμητες καθυστερήσεις. Ο ελεγκτής των πόρων σε τέτοιες περιπτώσεις ενημερώνει τον κεντρικό δρομολογητή ότι λόγω φόρτου εργασίας του υποσυστήματος μετάδοσης η αίτηση δεν μπορεί να εξυπηρετηθεί.

Στους στόχους των μελλοντικών επεκτάσεων του συστήματος είναι ο ελεγκτής των πόρων να δοκιμάζει να μετακινήσει την εξυπηρέτηση του αντικειμένου σε άλλους κύκλους και να εξετάζει πάλι την σχέση ($TRR < T_{max_rate}$).

Χρησιμοποιώντας τον αυτοπροσαρμοζόμενο μηχανισμό δρομολόγησης επιτυγχάνουμε την καλύτερη δυνατή αξιοποίηση των πόρων του συστήματος. Σε περιπτώσεις μικρού και μεσαίου φόρτου εργασίας ο προτεινόμενος μηχανισμός δρομολόγησης είναι σε θέση να ενεργοποιήσει τεχνικές προανάκτησης δεδομένων και λαμβάνοντας υπόψη τα χαρακτηριστικά των πελατών οδηγεί την μετάδοση δεδομένων με τον κατάλληλο ρυθμό. Η αντιμετώπιση αυτή έχει, όπως είναι φυσικό, ως αποτέλεσμα την γρήγορη εξυπηρέτηση αιτήσεων αναπαραγωγής πολυμέσων, όπου αυτό είναι δυνατόν και επομένως την ταχύτερη αποδέσμευση των κρίσιμων πόρων του συστήματος.

Από την άλλη πλευρά στις περιπτώσεις εκείνες όπου η προσωρινή μνήμη είναι πιο κρίσιμη για τον εξυπηρετητή, ο δρομολογητής αποφασίζει να ανακτά δεδομένα με τον κανονικό ρυθμό από τους δίσκους, επιβάλλοντας τον μέγιστο ρυθμό αποστολής δεδομένων για κάθε πελάτη.

7.6 Περίληψη

Η ανάγκη υποστήριξης πελατών με διαφορετικά χαρακτηριστικά προϋποθέτει ένα αυτοπροσαρμοζόμενο μηχανισμό δρομολόγησης και λήψης αποφάσεων.

Ο μηχανισμός αυτός στην φάση εγκατάστασης του συστήματος προκαλεί την δημιουργία νημάτων ελέγχου με στόχο την πειραματική λειτουργία των συσκευών υποστήριξης και την καταγραφή μετρήσεων. Τα αποτελέσματα των μετρήσεων επιτρέφονται στον κεντρικό δρομολογητή, ο οποίος αποκτά με αυτόν τον τρόπο μια εικόνα για τα φυσικά όρια απόδοσης του τρέχοντος συστήματος.

Όσον αφορά την δυνατότητα προσαρμογής του δρομολογητή σε ανοιχτά περιβάλλοντα, το πρώτο βήμα σε αυτή την προσπάθεια είναι η συγκέντρωση των χαρακτηριστικών των ετερογενών πελατών και η εύρεση των πραγματικών ικανοτήτων τους. Αυτό γίνεται μέσω μιας εφαρμογής που χρησιμοποιεί ο πελάτης για να εγγραφεί στην υπηρεσία ροής, ώστε να μπορεί να τύχει εξυπηρέτησης. Μετά το πέρας αυτής της διαδικασίας εγγραφής, ο δρομολογητής του εξυπηρετητή ενημερώνεται για τον μέγιστο ρυθμό, με τον οποίο μπορούν να αποσταλούν δεδομένα στον πελάτη χωρίς προβλήματα, και για τον διαθέσιμο χώρο αποταμίευσης του. Όλα αυτά τα στοιχεία συγκεντρώνονται και οργανώνονται από τον διαχειριστή αρχείων των πελατών.

Η πληροφορία που περιέχεται στις εγγραφές των πελατών επιτρέπει στον δρομολογητή να οδηγήσει τους μηχανισμούς προανάκτησης.

Ένα από τα πιο σημαντικά προβλήματα που εμποδίζουν την αύξηση της απόδοσης στους εξυπηρετητές ροής είναι το περιορισμένο μέγεθος της προσωρινής μνήμης αποταμίευσης που διαθέτουν. Είναι γνωστό αλλά και αποδεικνύεται πειραματικά ότι αυξάνοντας τον κύκλο εξυπηρέτησης του συστήματος και κατά συνέπεια το μέγεθος των τμημάτων που ανακτώνται από τους δίσκους αυξάνεται σημαντικά η απόδοση των μέσων αποθήκευσης. Αυτή η τεχνική όμως έχει ως βαρύ τίμημα την ραγδαία αύξηση των απαιτήσεων σε προσωρινή μνήμη ώστε να καταστεί εφικτή η εξυπηρέτηση πολλών πελατών. Το γεγονός αυτό επομένως μας αναγκάζει να οδηγούμε την λειτουργία των δίσκων σε χαμηλότερα επίπεδα απόδοσης ώστε να μειώσουμε τις ανάγκες προσωρινής αποταμίευσης.

Στο κεφάλαιο αυτό δείξαμε ότι είναι εφικτή η ελαχιστοποίηση των επιπτώσεων στην προσωρινή αποταμίευση, από την αύξηση του μεγέθους των τμημάτων ανάκτησης, αν ο δρομολογητής της υπηρεσίας εκμεταλλευτεί τις δυνατότητες για λήψη δεδομένων και αποταμίευση των πελατών, σύμφωνα με τις πληροφορίες που του παρέχει ο διαχειριστής αρχείων των πελατών. Έτσι επιβάλλοντας την αποταμίευση των δεδομένων στην πλευρά των πελατών και αποστέλλοντας δεδομένα με τον μέγιστο δυνατό αριθμό, μειώνονται δραστικά οι απαιτήσεις αποταμίευσης και καθίσταται δυνατή η πλήρη αξιοποίηση των μέσων αποθήκευσης του συστήματος.

Τα παραπάνω συμπεράσματα προέκυψαν από τα αποτελέσματα πειραμάτων, σύμφωνα με τα οποία με τον νέο μηχανισμό δρομολόγησης το σύστημα του εξυπηρετητή (Ultra One, ATM, 4 δίσκοι, 2 SCSI) είναι σε θέση να τροφοδοτεί με MPEG1 δεδομένα 80 ισχυρούς πελάτες διασυνδεδεμένους σε ATM δίκτυο ή 1900 πελάτες με τον κατάλληλο εξοπλισμό (Ethernet ή ADSL modems) που ζητούν την αναπαραγωγή αντικειμένων ήχου με απαιτήσεις 64Kbps.

➤ Κλιμάκωση της Υπηρεσίας ροής



Στο κεφάλαιο αυτό εξετάζονται ζητήματα επέκτασης της υπηρεσίας αποθήκευσης δεδομένων ροής. Η κλιμάκωση έχει ιδιαίτερη σημασία για ένα εξυπηρετητή πολυμέσων, καθώς τον καθιστά ικανό να προσαρμόζεται ανάλογα με τις πραγματικές συνθήκες στις οποίες πρέπει να εφαρμοστεί. Αυτή η προσαρμογή γίνεται είτε με ελάττωση των πόρων σε ένα περιβάλλον μικρών απαιτήσεων είτε με αύξηση των πόρων και επέκταση του συστήματος τις περισσότερες φορές. Στην συνέχεια του κεφαλαίου εξετάζονται τρεις βασικές περιπτώσεις επέκτασης της υπηρεσίας ροής σε τρία διαφορετικά υπολογιστικά περιβάλλοντα. Μεγαλύτερη έμφαση δίνεται στην επέκταση του εξυπηρετητή σε διανεμημένο περιβάλλον με την χρήση πολλών υπολογιστικών μηχανών που φέρουν τα στιγμιότυπα των εξυπηρετητών.

8.1 Κλιμάκωση συστήματος

Η ανάπτυξη ενός υπολογιστικού συστήματος που θα μπορεί από μόνο του να αποθηκεύσει πολλά μέσα συνεχόμενης ροής και θα είναι ικανό να εξυπηρετεί ένα μεγάλο αριθμό πελατών είναι πολύ δύσκολη εργασία αν αναλογιστεί κανείς τις απαιτήσεις σε εύρος μετάδοσης, αποθηκευτικού χώρου και πραγματικού χρόνου που συνοδεύουν την πληροφορία πολυμέσων. Έτσι το πρόβλημα που εμφανίζεται είναι ότι εφόσον έχουμε συστήματα τα οποία μπορούν να διαχειριστούν ένα μικρό αριθμό από αντικείμενα ροής και να εξυπηρετήσουν μερικούς πελάτες, πώς μπορούμε να τοποθετήσουμε νέους πόρους ώστε κλιμακώνοντας το σύστημα να το καταστήσουμε ικανό να διαχειριστεί μεγαλύτερο ποσό δεδομένων και αριθμό πελατών.

Η κλιμάκωση στην πληροφορική σημαίνει ότι όχι μόνο όλα τα τμήματα του υπολογιστικού συστήματος (υλικό και λογισμικό) συνεχίζουν να λειτουργούν σωστά καθώς γίνεται η κλιμάκωση, αλλά συγχρόνως εκμεταλλεύονται πλήρως αυτήν την ανάπτυξη. Έτσι ένα σύστημα εξυπηρέτησης πολυμέσων με δυνατότητα κλιμάκωσης μπορεί να διαχειριστεί μια αύξηση στον αριθμό των χρηστών με την κατάλληλη προσθήκη πόρων χωρίς να εμφανίζονται προβλήματα και φαινόμενα συμφόρησης. Το σύστημα πρέπει να αντιμετωπίζει την αύξηση σε απαιτήσεις χωρίς αλλαγές στην δομή του και στους εφαρμοζόμενους αλγορίθμους.

Οι πιο σημαντικές παράμετροι που εμφανίζουν προβλήματα στην κλιμάκωση ενός συστήματος είναι:

- Η αύξηση της βάσης των χρηστών (οδηγεί σε προβλήματα στα υποσυστήματα αποθήκευσης και δικτύου)

- Το μέγεθος δεδομένων των μέσων (έχει επιπτώσεις στο δίκτυο και στην διαθεσιμότητα αποθηκευτικού χώρου)
- Η μη ομοιόμορφη κατανομή των απαιτήσεων των αιτήσεων για ένα χρονικό διάστημα στους διαθέσιμους διαφορετικούς πόρους του ίδιου τύπου (επεξεργαστές, δίσκους, κάρτες δικτύου).

Οι εξυπηρετητές πολυμέσων και ειδικά τα συστήματα λογισμικού πρέπει να αναπτύσσονται με τέτοιο τρόπο ώστε να είναι ανεξάρτητα από περιορισμούς στους πόρους και ευέλικτα για να εγγυηθούν την κλιμάκωση, διατηρώντας τον χρόνο απόκρισης, την διαθεσιμότητα και την αξιοπιστία σε ικανοποιητικά επίπεδα.

Ο πολυνηματικός σχεδιασμός της υπηρεσίας αποθήκευσης δεδομένων ροής έγινε με τον κατάλληλο τρόπο, ώστε να υπάρχει η δυνατότητα για αποδοτική επέκταση του συστήματος με την αύξηση των πόρων. Η εύκολη προσαρμογή του λογισμικού χωρίς χρονοβόρες ή ριζικές αλλαγές σε περιβάλλοντα διαφορετικής κλίμακας μεγέθους, είναι ζωτικής σημασίας. Ο κατάλληλος διαχωρισμός των εργασιών που επιτεύχθηκε στην υπηρεσία ροής και ιδιαίτερα η χρήση νημάτων διευκολύνουν τα σενάρια κλιμάκωσης της υπηρεσίας σε διάφορα περιβάλλοντα.

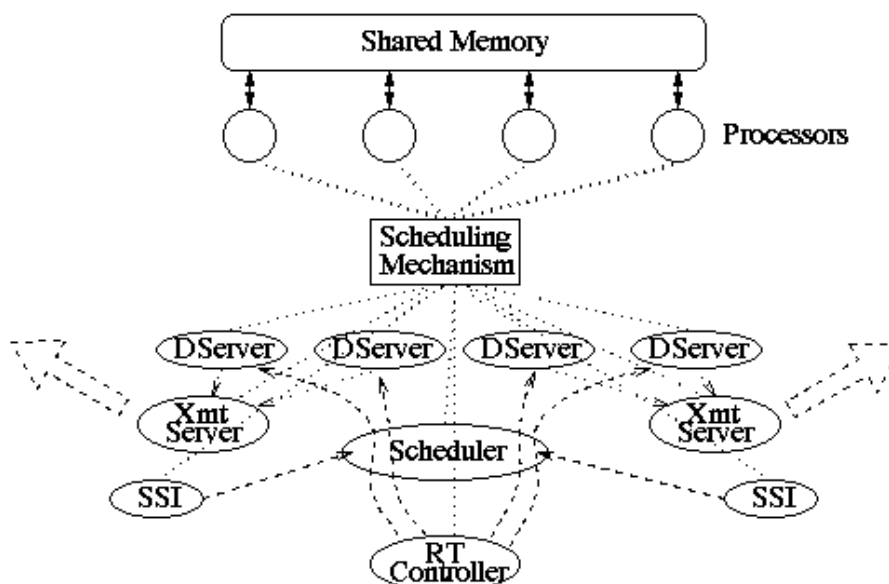
Σε αυτό το κεφάλαιο παρουσιάζονται τρία διαφορετικά σενάρια και τρόποι για αποδοτική επέκταση του συστήματος της διατριβής. Το πρώτο αναφέρεται στην εφαρμογή του λογισμικού σε πολυεπεξεργαστικό σύστημα διαμοιραζόμενης μνήμης, το δεύτερο σε περιβάλλοντα παράλληλων μηχανών ευρείας κλίμακας και τέλος εξετάζουμε πιο διεξοδικά την περίπτωση επέκτασης του συστήματος σε διανεμημένο περιβάλλον, χρησιμοποιώντας ομάδες από σταθμούς εργασίας. Ιδιαίτερο ενδιαφέρον παρουσιάζει η τελευταία περίπτωση καθώς παρουσιάζει μεγαλύτερη δυσκολία, αλλά εμφανίζει ένα σημαντικό πλεονέκτημα, αυτό του μειωμένου κόστους υλικού για την υποστήριξη του εξυπηρετητή πολυμέσων.

Όπως είναι φυσικό η δυνατότητα κλιμάκωσης της υπηρεσίας ροής εξαρτάται από την προσαρμογή του λογισμικού του συστήματος, όταν είναι αναγκαία η προσθήκη επιπλέον πόρων για την αντιμετώπιση αυξημένου φόρτου εργασίας. Οι πόροι αυτοί αναφέρονται στην κύρια μνήμη, τα αποθηκευτικά μέσα, τις συσκευές δικτύου και τους διαθέσιμους επεξεργαστές.

8.2 Κλιμάκωση του συστήματος σε πολυεπεξεργαστικό περιβάλλον διαμοιραζόμενης μνήμης.

Τα πολυεπεξεργαστικά συστήματα διαμοιραζόμενης μνήμης είναι υπολογιστικά συστήματα τα οποία επιτρέπουν την αξιοποίηση ενός αριθμού από επεξεργαστές. Βασικό χαρακτηριστικό αυτής της αρχιτεκτονικής είναι ότι οι επεξεργαστές έχουν διαμοιραζόμενη κοινή μνήμη. Ένα γεγονός πολύ σημαντικό καθώς επιτρέπει σε διαφορετικά νήματα μιας διεργασίας να επεξεργάζονται σε διαφορετικούς επεξεργαστές και να έχουν πρόσβαση στον κοινό χώρο μνήμης της διεργασίας. Έτσι με αυτό τον τρόπο αποφεύγεται το μεγάλο κόστος

Για να επιτευχθεί η επέκταση της υπηρεσίας ροής, ένας εφικτός τρόπος είναι η αύξηση των νημάτων που ασχολούνται με την εκτέλεση μιας εργασίας. Για παράδειγμα, φυσικό είναι η πρόσθεση καινούργιων αποθηκευτικών μέσων να συνοδεύεται αυτόματα με την αντίστοιχη αύξηση των νημάτων-εξυπηρετητών δίσκου, που αναλαμβάνουν την λειτουργία των νέων δίσκων. Όπως επίσης φυσικό είναι η προσθήκη μιας νέας συσκευής δικτύου να οδηγεί στην δημιουργία ενός νέου νήματος εξυπηρετητή μετάδοσης, σε μια προσπάθεια αύξησης της ρυθμοαπόδοσης του συστήματος. Η αύξηση των διαθέσιμων πόρων συνεπάγεται λογικά την εμφάνιση νέων επεξεργαστικών αναγκών για την διαχείρισή τους. Έτσι η ύπαρξη περισσότερων του ενός επεξεργαστή είναι πάντα απαραίτητη. Ιδιαίτερα για την κάλυψη των αναγκών μετάδοσης, που όπως είδαμε από τα πειράματα στο προηγούμενο κεφάλαιο χρειάζονται αυξημένη επεξεργαστική ισχύ.



Υποστήριξη Υπηρεσιών Ροής

Στο κεφάλαιο 6 περιγράφηκε ότι ο ελεγκτής πραγματικού χρόνου ζητάει συνεχώς χρονικά σημεία από το σύστημα για την διατήρηση των κύκλων εξυπηρέτησης, αναλαμβάνει τον συντονισμό των νημάτων δίσκου/μετάδοσης και την εγγραφή των υποαιτήσεων δίσκου στο κανάλι επικοινωνίας (DSR-Channel).

Όπως ο ελεγκτής πραγματικού χρόνου, έτσι και ο κεντρικός δρομολογητής δεν εμφανίζεται διασπασμένος στο σχήμα επέκτασης. Οι απαιτήσεις του για επεξεργαστική ισχύ είναι πολύ μικρές καθώς περιορίζονται στην συλλογή των αιτήσεων, δημιουργία νέων νημάτων εξυπηρέτησης και διατήρηση της πληροφορίας στους πίνακες δρομολόγησης και επεξεργασμένων αιτήσεων. Συνεπώς δεν αποτελεί σημείο το οποίο εμποδίζει την αύξηση της απόδοσης και επομένως δεν συντελείται κάποια αλλαγή σε αυτό.

8.3 Εφαρμογή & Κλιμάκωση του συστήματος σε περιβάλλοντα παράλληλης επεξεργασίας

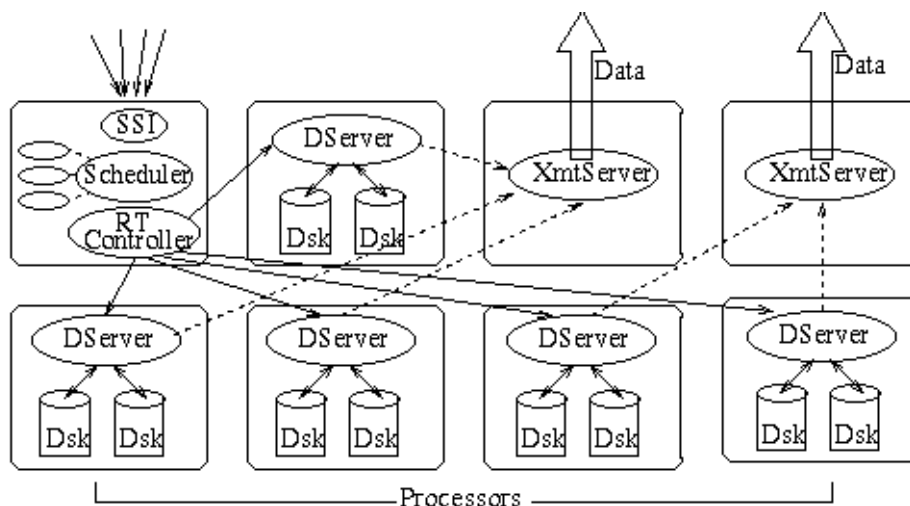
Τα παράλληλα συστήματα όπως και οι σειριακοί υπολογιστές, λειτουργούν εκτελώντας εντολές πάνω σε δεδομένα. Μια ροή από εντολές (ο αλγόριθμος) υποδεικνύει στον υπολογιστή πως πρέπει να ενεργήσει σε κάθε βήμα. Μια ροή από δεδομένα (η είσοδος του αλγορίθμου), επηρεάζεται από τις εντολές αυτές. Υπάρχουν τέσσερα υπολογιστικά μοντέλα, τα οποία εξαρτώνται από την ύπαρξη μιας ή περισσότερων από τις ροές αυτές [AKL96]. Έτσι τα παράλληλα συστήματα διακρίνονται σε αρχιτεκτονικές μονής ροής εντολών και μονής ροής δεδομένων (SISD), πολλαπλής ροής εντολών και μονής ροής δεδομένων (MISD), μονής ροής εντολών και πολλαπλής ροής δεδομένων (SIMD) και τέλος πολλαπλής ροής εντολών και πολλαπλής ροής δεδομένων (MIMD). Η τελευταία κλάση είναι η πιο γενική, η πιο ισχυρή και συναντάται περισσότερο συχνά. Οι επεξεργαστές μιας παράλληλης μηχανής MIMD, φέρουν την δικιά τους μονάδα ελέγχου, τοπική μνήμη και μονάδες υπολογισμών και λογικής. Κάθε επεξεργαστής λειτουργεί κάτω από τον έλεγχο μιας ροής εντολών που υποβάλλεται από την μονάδα ελέγχου. Συνεπώς οι επεξεργαστές εκτελούν συνήθως διαφορετικά προγράμματα πάνω σε διαφορετικά δεδομένα. Ένα γεγονός που τυπικά σημαίνει ότι οι επεξεργαστές λειτουργούν ασύγχρονα. Οι επεξεργαστές συνδέονται με ένα εσωτερικό δίκτυο υψηλών ταχυτήτων. Αυτή είναι η κύρια διαφοροποίησή τους από τα πολυεπεξεργαστικά συστήματα, στα οποία η επικοινωνία των επεξεργαστών λαμβάνει χώρα μέσω της κοινοποιημένης μνήμης.

Αυτή η διαφορά σε συνδυασμό με το γεγονός ότι ο προγραμματισμός στα παράλληλα συστήματα είναι διαφορετικός από την φύση του, καθιστά την εφαρμογή της υπηρεσίας ροής μια δύσκολη και πολύπλοκη διαδικασία. Για παράδειγμα η εφαρμογή προϋποθέτει τον απαραίτητο διαχωρισμό μεταξύ των εννοιών διεργασία-επεξεργαστής. Ένας ασύγχρονος αλγόριθμος είναι μια συλλογή διεργασιών, μερικές από τις οποίες εκτελούνται ταυτόχρονα σε ένα αριθμό από διαθέσιμους επεξεργαστές.

Αρχικά όλοι οι επεξεργαστές είναι ελεύθεροι. Το παράλληλο πρόγραμμα αρχίζει την εκτέλεσή του σε κάποιον επεξεργαστή, η επιλογή του οποίου γίνεται αυθαίρετα. Αμέσως μετά δημιουργεί ένα αριθμό από υπολογιστικές εργασίες ή διαφορετικά διεργασίες που πρέπει να

εκτελεστούν. Εφόσον δημιουργείται μια διεργασία πρέπει να ανατεθεί σε ένα επεξεργαστή. Η επικοινωνία των διεργασιών που εκτελούνται ταυτόχρονα, συντελείται με την βοήθεια ειδικού λογισμικού και μέσω του εσωτερικού δικτύου διασύνδεσης των επεξεργαστών. Η μεταφορά συνεπώς της υπηρεσίας ροής σε παράλληλη μηχανή, προϋποθέτει την πληροφορία ανάθεσης διεργασιών σε συγκεκριμένους επεξεργαστές, η οποία θα παραχθεί μετά από την μελέτη των επικοινωνιακών αναγκών και της χρησιμοποιούμενης τοπολογίας του εσωτερικού δικτύου.

Στην γενική περίπτωση κάθε νήμα μπορεί να αποτελέσει μια διεργασία που τρέχει σε έναν επεξεργαστή. Ο διαχωρισμός όμως των νημάτων και η μετατροπή τους σε διεργασίες χρειάζεται προσοχή. Πρέπει δε να λαμβάνει χώρα μόνο σε περιπτώσεις όπου η συνύπαρξη νημάτων μεταφράζεται σε αυξημένη επεξεργαστική επιβάρυνση, καθώς ο διαχωρισμός εμφανίζει νέο φορτίο στην λειτουργία του συστήματος με το επιπρόσθετο επικοινωνιακό κόστος που απαραίτητα χρειάζεται για την επικοινωνία των διαχωρισμένων νημάτων. Στον σχεδιασμό είναι επίσης αναγκαίο να γίνει και η σύγκριση των αποτελεσμάτων μιας τέτοιας ενέργειας, δηλαδή με άλλα λόγια πρέπει να αποφασιστεί σε ποιες περιπτώσεις το επιπρόσθετο επικοινωνιακό κόστος θα επιφέρει μεγαλύτερη ευελιξία στην δυνατότητα επέκτασης του συστήματος, παρέχοντας ισορροπία στο υπολογιστικό φορτίο των επεξεργαστών.



Στο σχήμα παρουσιάζεται η εφαρμογή της υπηρεσίας αποθήκευσης ροής δεδομένων σε μια παράλληλη μηχανή, μετά από αρκετές αλλαγές σε πολλά τμήματα λογισμικού και ιδιαίτερα στους εξυπηρετητές μετάδοσης. Οι αλλαγές αυτές για την μεταφορά του εξυπηρετητή πολυμέσων σε παράλληλη μηχανή έγινε στα πλαίσια του προγράμματος SICMA για την υποστήριξη μιας πειραματικής εφαρμογής (trial). Η πειραματική εφαρμογή αποτελούσε ένα πρόγραμμα τήλε-εκπαίδευσης το οποίο παρείχε πληροφορία πολυμέσων (video) σε δεκάδες φοιτητές υποστηρίζοντας τα τρέχοντα μαθήματά τους και έλαβε χώρα στο πανεπιστήμιο Limburg στο Hasselt του Βελγίου για δύο μήνες. Για τις ανάγκες και τις απαιτήσεις της συγκεκριμένης πειραματικής εφαρμογής, η αρχιτεκτονική αυτή αποδείχθηκε ικανοποιητική. Γίνεται όμως εύκολα κατανοητό ότι εμφανίζει περιορισμένες δυνατότητες επέκτασης. Η συνεχής αύξηση των επεξεργαστών που ελέγχουν την μετάδοση δεδομένων

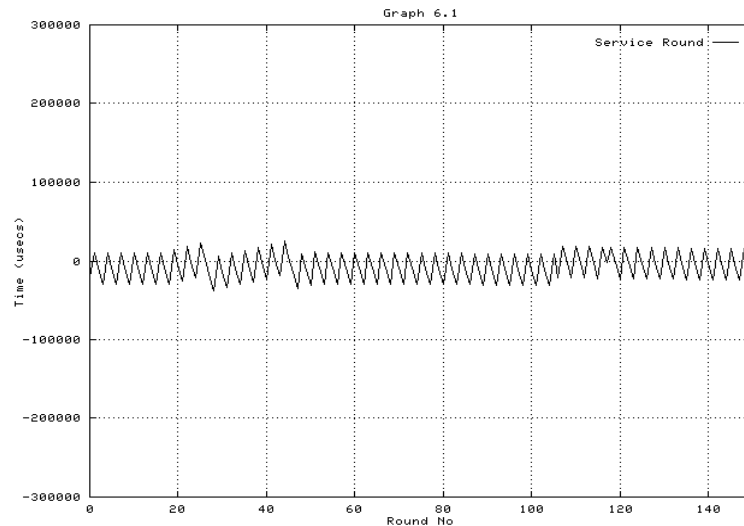
και τα μέσα αποθήκευσης είναι ο τρόπος επέκτασης του συστήματος για την αντιμετώπιση αυξημένου φόρτου εργασίας. Αυτή η διαδικασία είναι λογικό να οδηγήσει τον κόμβο δρομολόγησης σε φαινόμενα συμφόρησης, ώστε να αποτελεί εμπόδιο στην συνολική απόδοση. Για τον λόγο αυτό χρειάζεται μεγαλύτερη παραλληλοποίηση του μηχανισμού δρομολόγησης, με ιδέες που μπορούν να βασιστούν στο σενάριο της επέκτασης στα πολυεπεξεργαστικά συστήματα διαμοιραζόμενης μνήμης. Περισσότερη όμως έρευνα στους τρόπους επέκτασης της υπηρεσίας ροής σε παράλληλα συστήματα δεν αποτέλεσε στόχο της παρούσας διατριβής.

8.4 Κλιμάκωση του συστήματος σε διανεμημένο περιβάλλον

Η κλιμάκωση του συστήματος, ώστε να μπορεί να ανταποκριθεί στην εξυπηρέτηση μεγάλου αριθμού χρηστών, γίνεται με την διασπορά σε δίκτυο πολλών στιγμιότυπων εξυπηρετητών. Η υπηρεσία διαχείρισης δεδομένων ροής έχει σχεδιαστεί ώστε να μπορεί να συνεργάζεται και με άλλους εξυπηρετητές. Η ιδέα είναι να συγχρονίσουμε τους κύκλους όλων των εξυπηρετητών, ώστε ο κάθε εξυπηρετητής να εργάζεται σχεδόν ανεξάρτητα με στόχο να περιορίσουμε τον όγκο των δεδομένων που χρειάζονται να μεταφέρονται μεταξύ των κόμβων του συστήματος (όπως παρατηρείται σε πολλούς σχεδιασμούς στην βιβλιογραφία) και περιορίζουν την απόδοση του συστήματος σε συνολικό επίπεδο. Έτσι στην συνέχεια εφαρμόζονται τεχνικές striping σε επίπεδο δικτύου επιτυγχάνουμε ισορροπία φόρτου και κατά συνέπεια αποδοτική κλιμάκωση.

Η επιτυχία του μηχανισμού συγχρονισμού πολλών εξυπηρετητών βασίζεται στην ακρίβεια διατήρησης των κύκλων εξυπηρέτησης μέσω του εφαρμοζόμενου μηχανισμού δρομολόγησης που περιγράφηκε στο προηγούμενο κεφάλαιο. Τα επόμενα διαγράμματα αποδεικνύουν το γεγονός αυτό. Όπως περιγράφηκε στο 6ο κεφάλαιο ο μηχανισμός δρομολόγησης της υπηρεσίας ροής ακολουθεί ένα σχήμα κύκλων εξυπηρέτησης. Σε κάθε κύκλο του συστήματος λαμβάνουν χώρα ένα σύνολο από εργασίες που καθορίζονται από τον κεντρικό δρομολογητή. Η έναρξη των κύκλων εξυπηρέτησης προσδιορίζονται από τον ελεγκτή πραγματικού χρόνου (RT-Controller), ο οποίος ελέγχει περιοδικά σε υποκύκλους (η χρονική διάρκεια των οποίων αποτελεί υποπολλαπλάσιο του κανονικού κύκλου) τα χρονικά σημεία που λαμβάνονται από το ρολόι του συστήματος και αποφασίζει πότε πρέπει να αρχίσει ο επόμενος κύκλος εξυπηρέτησης. Επιπλέον σε κάθε υποκύκλο ο ελεγκτής πραγματικού χρόνου ενεργοποιεί τον εξυπηρετητή μετάδοσης (αν είναι αδρανής) και τον ενημερώνει για το τρέχων χρονικό σημείο ώστε να δρομολογήσει κατάλληλα την μετάδοση δεδομένων.

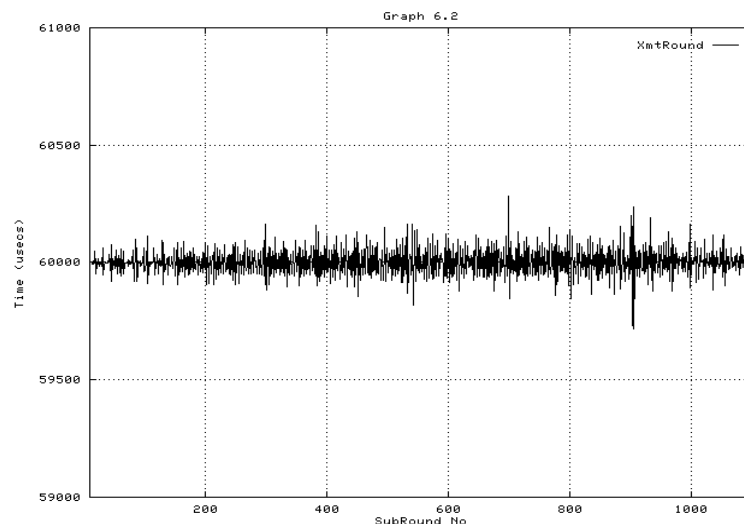
Με στόχο να υπολογίσουμε την ακρίβεια της λειτουργίας του παραπάνω μηχανισμού των κύκλων εξυπηρέτησης κάτω από πραγματικές συνθήκες, καταγράψαμε μετρήσεις εσωτερικά στην υπηρεσία ροής κατά την διάρκεια ενός πειράματος. Σύμφωνα με το πείραμα αυτό, τρέχουμε την υπηρεσία ροής σε ένα υπολογιστή Ultra One και σταδιακά εισάγουμε αιτήσεις πελατών για αναπαραγωγή μέσων μέχρι τους 70 πελάτες (μέγιστη απόδοση σύμφωνα με το πείραμα της ενότητας 6.11).



Σχήμα 109: Χρονικές αποκλίσεις στον καθορισμό των κύκλων του συστήματος
Στο σχήμα εικονίζεται γραφικά η χρονική απόκλιση του θεωρητικά υπολογισμένου χρόνου έναρξης του νέου κύκλου εξυπηρέτησης από τον χρόνο που πρακτικά καθορίζεται από τον ελεγκτή πραγματικού χρόνου κατά την διάρκεια του πειράματος. Η γραφικά παράσταση που προκύπτει, εμφανίζει καλή ακρίβεια, καθώς οι αποκλίσεις κυμαίνονται περίπου στο επίπεδο των 10msec.

Στο Σχήμα 109 εμφανίζονται οι χρονικές αποκλίσεις που παρουσιάζονται στον RT-Controller στον καθορισμό της έναρξης των κύκλων εξυπηρέτησης, ενώ στο Σχήμα 110 περιγράφονται οι αποκλίσεις στον καθορισμό των υποκύκλων λειτουργίας του εξυπηρετητή μετάδοσης. Και στα δυο διαγράμματα καταγράφονται μετρήσεις που επιβεβαιώνουν την καλή ακρίβεια λειτουργίας.

Το γεγονός αυτό επιτρέπει να υλοποιηθεί ένα σχήμα συγχρονισμένων εξυπηρετητών ροής σε διανεμημένο περιβάλλον, καθώς ο καθορισμός των εσωτερικών κύκλων λειτουργίας του συστήματος είναι αρκετά ακριβής και παραμένει σταθερός, τόσο στην διάρκεια του χρόνου, όσο και σε συνθήκες υψηλού φόρτου εργασίας.



Σχήμα 110: Χρονικές αποκλίσεις στον καθορισμό των υπο-κύκλων του συστήματος
Στο σχήμα εικονίζεται γραφικά η χρονική διάρκεια όλων των υποκύκλων που καθορίζονται από τον ελεγκτή πραγματικού χρόνου κατά την διάρκεια του πειράματος. Παρατηρώντας την γραφική παράσταση συμπεραίνουμε ότι οι αποκλίσεις από την κανονική διάρκεια (60msec) είναι πολύ μικρές (< 250 nsec), ώστε να εξασφαλίζεται η χρονική ακρίβεια λειτουργίας του εξυπηρετητή μετάδοσης.

8.4.1 Συγχρονισμός Υπολογιστών μέσω δικτύου

Στην προηγούμενη ενότητα αναφέρθηκε ότι η βασική ιδέα του σχεδιασμού επέκτασης της υπηρεσίας αποθήκευσης δεδομένων ροής είναι ο συγχρονισμός των μηχανισμών δρομολόγησης των επιμέρους εξυπηρετητών-στιγμιότυπων που βρίσκονται διασυνδεδεμένοι σε τοπικό δίκτυο. Στην συνέχεια αυτής της ενότητας παρουσιάζεται το ζήτημα του χρονικού συντονισμού δικτυακών υπολογιστών.

Η βασική κλίμακα χρόνου που χρησιμοποιείται από τα περισσότερα κράτη του κόσμου είναι ο παγκόσμιος χρόνος συντονισμού (Universal Coordinated Time UTC), ο οποίος βασίζεται στην περιστροφή της γης γύρω από τον άξονά της και το Γρηγοριανό ημερολόγιο. Η κλίμακα UTC υιοθετείται από διάφορα μέσα συμπεριλαμβανομένων των συστημάτων πλοήγησης με ραδιοκύματα και δορυφορική υποστήριξη, modems και φορητά ρολόγια.

Ειδικοί λήπτες είναι διαθέσιμοι για τον συγχρονισμό συσκευών λαμβάνοντας χρονική πληροφορία. Για λόγους κόστους και ευκολίας, δεν είναι δυνατός ο εξοπλισμός κάθε υπολογιστή με τους κατάλληλους λήπτες. Ωστόσο είναι εφικτό να εξοπλιστούν κάποιοι υπολογιστές, οι οποίοι θα ενεργούν ως πρωτεύοντες εξυπηρετητές χρόνου και θα συγχρονίζουν ένα μεγάλο αριθμό από δευτερεύοντες εξυπηρετητές και πελάτες. Για την επίτευξη του παραπάνω στόχου είναι απαραίτητη η χρήση του κατάλληλου πρωτοκόλλου επικοινωνίας. Σχετικά πρωτόκολλα είναι το NTP (Network Time Protocol) και το DTSS (Digital Time Synchronization Protocol).

Το πρωτόκολλο συγχρονισμού καθορίζει την διαφορά του τρέχοντος χρόνου του εξυπηρετητή, όπως διατηρείται από το ρολόι του συστήματος, σε σχέση με τον χρόνο του πελάτη. Τα διάφορα πρωτόκολλα συντονισμού που χρησιμοποιούνται, παρέχουν διαφορετικούς τρόπους συντονισμού, αλλά όλα ακολουθούν την ίδια γενική ιδέα. Κατ' απαίτηση ο εξυπηρετητής στέλνει ένα μήνυμα, το οποίο περιλαμβάνει το τρέχων χρονικό σημείο και ο πελάτης καταγράφει αυτό το χρονικό σημείο μόλις λάβει το μήνυμα. Για μεγαλύτερη ακρίβεια ο πελάτης χρειάζεται να μετρήσει την καθυστέρηση μετάδοσης εξυπηρετητή-πελάτη, ώστε να υπολογίσει αξιόπιστα την διαφορά χρόνου σε σχέση με τον εξυπηρετητή. Εφόσον όμως δεν είναι δυνατόν να καθορίσει την καθυστέρηση αποστολής μηνύματος μιας κατεύθυνσης, το πρωτόκολλο υπολογίζει τον συνολικό χρόνο κυκλικής μετάδοσης ενός μηνύματος (εξυπηρετητής –πελάτης -εξυπηρετητής) και υποθέτει ότι οι χρόνοι μετάδοσης είναι στατιστικά ισοδύναμοι για κάθε κατεύθυνση. Στην γενική περίπτωση αυτή η υπόθεση είναι μια χρήσιμη προσέγγιση και ιδιαίτερα όταν έχουμε να συγχρονίσουμε υπολογιστές διασυνδεδεμένους σε τοπικό δίκτυο.

Τα πρωτόκολλα συγχρονισμού λειτουργούν με διαφορετικούς τρόπους, εξαρτώμενους από τον σχεδιασμό. Ο τρόπος λειτουργίας (mode) πελάτη- εξυπηρετητή ή αλλιώς αφέντη-σκλάβου υποστηρίζονται και από το NTP και από το DTSS.

8.4.2 Πρωτόκολλο SNTP

Το πρωτόκολλο Simple Network Time Protocol (SNTP) αποτελεί ένα υποσύνολο του NTP και χρησιμοποιείται για τον χρονικό συντονισμό πολλών υπολογιστών μέσω του διαδικτύου. Το SNTP μπορεί να λειτουργήσει και στους δύο τρόπους επικοινωνίας σημείο

με σημείο και σημείο με πολλά σημεία (unicast, multicast). Επίσης μπορεί να λειτουργήσει χρησιμοποιώντας IP multicast όπου αυτό είναι διαθέσιμο. Το πρωτόκολλο SNTP δεν επιφέρει καμία αλλαγή στις τρέχουσες υλοποιήσεις (ή specifications) του NTP πρωτοκόλλου, αλλά κυρίως αποσαφηνίζει συγκεκριμένα χαρακτηριστικά σχεδιασμού του NTP που επιτρέπουν την λειτουργία του με ένα πιο απλό τρόπο (με χρήση RPC), παρέχοντας παρόμοια ακρίβεια και αξιοπιστία με το πρωτόκολλο UDP/TIME που περιγράφεται στο RFC-868.

Το πρωτόκολλο SNTP προτείνεται για την κάλυψη αναγκών συγχρονισμού υπολογιστών υποδικτύων, ακολουθώντας την ακρίβεια που επιτυγχάνεται με την χρήση του NTP (1-50msec).

8.4.3 Η επέκταση της υπηρεσίας ροής

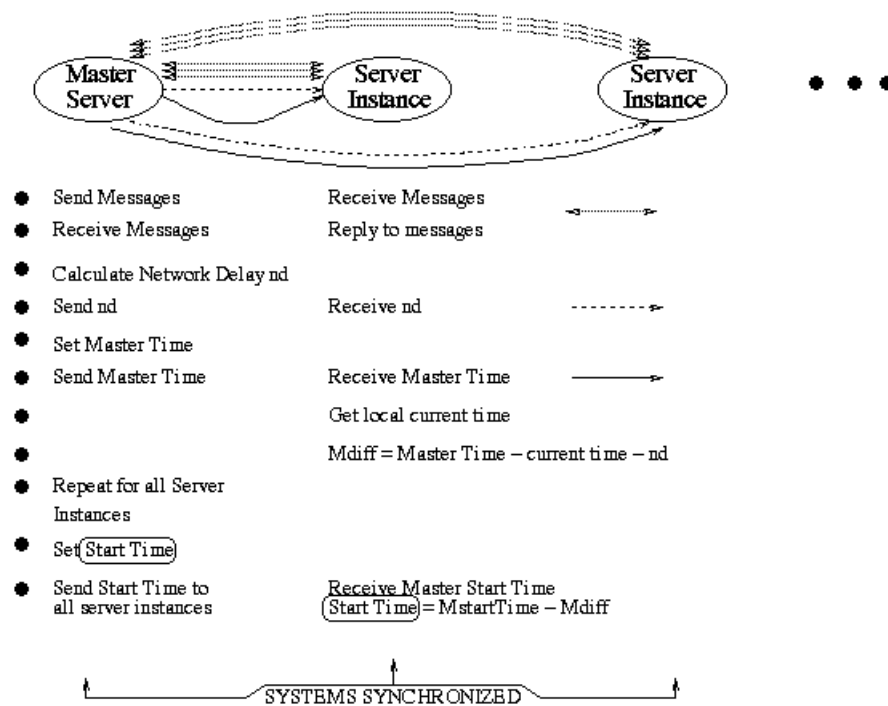
Στα πλαίσια της διατριβής έγιναν τα πρώτα βήματα εφαρμογής ενός σχήματος επέκτασης της υπηρεσίας ροής σε διανεμημένο περιβάλλον, ώστε να καταστήσουμε τον εξυπηρετητή ροής ικανό να προσαρμόζεται σε εφαρμογές διαφορετικών απαιτήσεων.

Όπως περιγράφηκε στις προηγούμενες υποενότητες η επέκταση μπορεί να πραγματοποιηθεί χρησιμοποιώντας ένα αριθμό από εξυπηρετητές ροής διασυνδεδεμένους σε δίκτυο, η λειτουργία των οποίων συντονίζεται χρονικά. Στα επόμενα σχήματα περιγράφεται γραφικά ο τρέχων μηχανισμός συντονισμού των κύκλων εξυπηρέτησης όλων των κόμβων εξυπηρετητών και παρουσιάζεται ένα παράδειγμα με τρεις εξυπηρετητές.

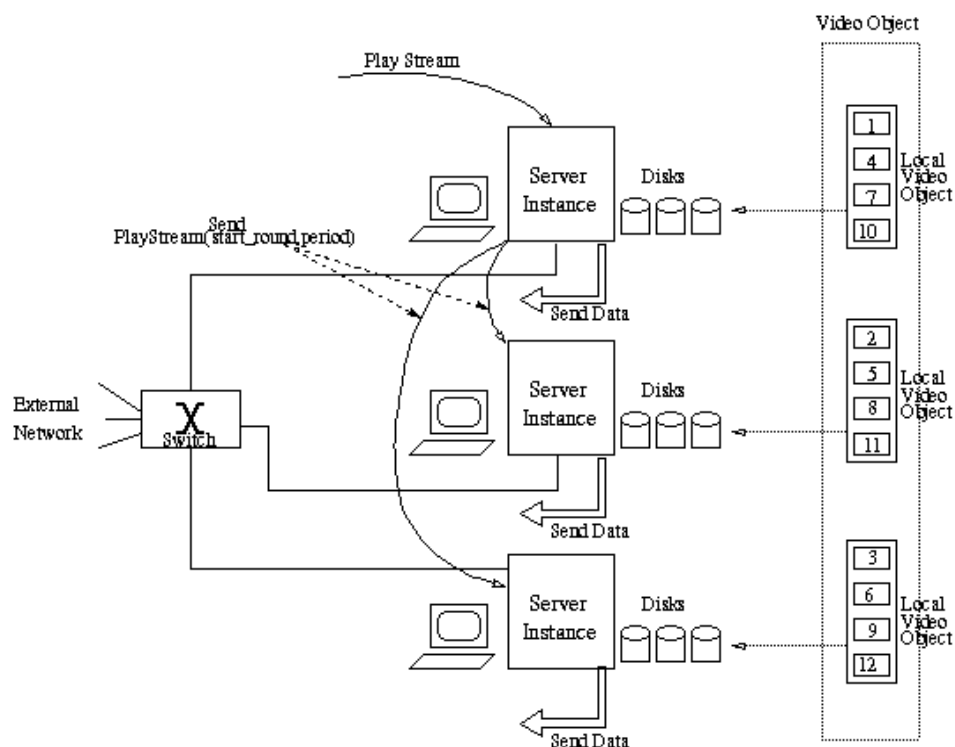
Στο Σχήμα 111 περιγράφεται με απλό τρόπο ο συντονισμός των κύκλων επεξεργασίας των εξυπηρετητών στο δίκτυο με την χρήση ενός κεντρικού εξυπηρετητή (master server). Χρησιμοποιώντας τρεις εξυπηρετητές (Σχήμα 112) μπορούμε να αποθηκεύσουμε ένα αντικείμενο ροής διασκορπισμένα σε όλους τους εξυπηρετητές με κυκλική ανάθεση των τμημάτων. Κάθε εξυπηρετητής αναγνωρίζει τα τμήματα που αποθηκεύονται σαν τμήματα ενός τοπικού αντικειμένου με περίοδο εξυπηρέτησης $NoServers * SYS_ROUND$.

Έτσι σύμφωνα με το σχήμα αυτό, ο κεντρικός εξυπηρετητής που δέχεται την αίτηση καθορίζει τον κύκλο έναρξης της εξυπηρέτησης για κάθε εξυπηρετητή. Στην συνέχεια το διανεμημένο σύστημα αρχίζει να υποστηρίζει την αναπαραγωγή του αντικειμένου, ανακτώντας και μεταδίδοντας ένα τμήμα δεδομένων από ένα εξυπηρετητή σε κάθε κύκλο. Ο χρονικός συντονισμός των εξυπηρετητών επιτρέπει την ανεξάρτητη λειτουργία τους χωρίς επιπρόσθετο επικοινωνιακό κόστος.

Μερικά πειράματα που έλαβαν χώρα πάνω σε αυτό το σχήμα επέκτασης έδειξαν ικανοποιητικά αποτελέσματα. Περισσότερη όμως έρευνα πάνω στο αντικείμενο αυτό αποτελεί στόχο μελλοντικών εργασιών.



Σχήμα 111: Διαδικασία συντονισμού των κύκλων εξυπηρέτησης σε ένα cluster από εξυπηρετητές



Σχήμα 112 Αρχιτεκτονική κλιμάκωσης της υπηρεσίας εξυπηρέτησης δεδομένων συνεχόμενης ροής σε διανεμημένο περιβάλλον

8.5 Περίληψη

Ο πολυνηματικός σχεδιασμός της υπηρεσίας διαχείρισης δεδομένων ροής προέβλεψε την δυνατότητα αποδοτικής επέκτασης του συστήματος, ως αποτέλεσμα της αύξησης των πόρων. Η προσαρμογή του λογισμικού σε τέτοιες αλλαγές πρέπει να είναι άμεση και χωρίς ριζικές μετατροπές.

Στο κεφάλαιο αυτό περιγράφηκαν τρία διαφορετικά σενάρια κλιμάκωσης της υπηρεσίας ροής. Το πρώτο αναφέρεται στην εφαρμογή του λογισμικού σε πολυεπεξεργαστικό σύστημα διαμοιραζόμενης μνήμης, το δεύτερο σε περιβάλλοντα παράλληλων μηχανών, ενώ στο τρίτο εξετάζεται με μεγαλύτερη λεπτομέρεια η περίπτωση κλιμάκωσης του εξυπηρετητή ροής της διατριβής σε διανεμημένο περιβάλλον με την χρήση ομάδων εξυπηρετητών.

Ο σχεδιασμός και η ανάπτυξη της υπηρεσίας ροής περιλαμβάνει την δυνατότητα συνεργασίας με άλλα στιγμιότυπα εξυπηρετητών. Η κλιμάκωση του συστήματος επιτυγχάνεται με την διασπορά πολλών εξυπηρετητών στο δίκτυο. Η βασική ιδέα είναι να συγχρονίσουμε τους κύκλους όλων των εξυπηρετητών, ώστε ο κάθε εξυπηρετητής να εργάζεται σχεδόν ανεξάρτητα. Απώτερος στόχος αυτού του σχήματος αποτελεί ο περιορισμός του όγκου δεδομένων που χρειάζονται να μεταφέρονται μεταξύ των κόμβων του συστήματος, όπως παρατηρείται σε πολλές προσεγγίσεις στην βιβλιογραφία. Εφαρμόζοντας τεχνικές striping σε επίπεδο δικτύου επιτυγχάνουμε καλή ισορροπία φορτίου και κατά συνέπεια αποδοτική κλιμάκωση. Αξίζει να σημειωθεί ότι στην προσπάθεια συντονισμού των κύκλων εξυπηρέτησης όλων των εξυπηρετητών με αυτούς του κεντρικού εξυπηρετητή μπορούν να εφαρμοστούν γνωστά πρωτόκολλα που χρησιμοποιούνται για τον συντονισμό των ρολογιών των υπολογιστών μέσω δικτύου, όπως είναι το NTP.

. Η επιτυχία του μηχανισμού συγχρονισμού πολλών εξυπηρετητών βασίζεται στην ακρίβεια διατήρησης των κύκλων εξυπηρέτησης μέσω του εφαρμοζόμενου μηχανισμού δρομολόγησης που περιγράφηκε στο 6ο κεφάλαιο και καταγράφηκε πειραματικά.

➤ **Ανακεφαλαίωση και μελλοντικές επεκτάσεις**



Αυτό το κεφάλαιο αποτελεί μια προσπάθεια για ανακεφαλαίωση όλων όσων περιγράφηκαν και μελετήθηκαν στην παρούσα μεταπτυχιακή διατριβή. Επίσης γίνεται μια σύντομη αναφορά στους άμεσους στόχους βελτίωσης του συστήματος, καθώς και στις δυνατές μελλοντικές επεκτάσεις των εργασιών που έλαβαν χώρα και παρουσιάστηκαν στα προηγούμενα κεφάλαια.

Τα θέματα που εξετάστηκαν και περιγράφηκαν στην παρούσα εργασία αφορούν την ανάπτυξη μιας αποτελεσματικής αρχιτεκτονικής για την υποστήριξη υπηρεσιών ροής πολυμέσων σε ανοιχτά περιβάλλοντα ετερογενών πελατών.

Η σύγχρονη έκρηξη της πληροφορικής και της τεχνολογίας έχει δημιουργήσει τις προϋποθέσεις για την εμφάνιση νέων τύπων εφαρμογών με ανάγκες για την αποδοτική διαχείριση της πληροφορίας πολυμέσων. Η μεταπτυχιακή διατριβή συνέβαλε στην ανάγκη ανάπτυξης συστημάτων λογισμικού που θα αναλαμβάνουν την οργάνωση της αποθήκευσης και την προώθηση των δεδομένων στους πελάτες μέσω δικτύου. Η διατριβή ασχολήθηκε ιδιαίτερα με τους μηχανισμούς δρομολόγησης που πρέπει να εφαρμοστούν, ώστε να έχουμε τα καλύτερα δυνατά αποτελέσματα απόδοσης αλλά και αξιοποίησης των πόρων του συστήματος. Προσθετικά η αρχιτεκτονική του συστήματος προβλέπει την κλιμάκωση του εξυπηρετητή, καθώς το σύστημα λογισμικού έχει την ικανότητα επέκτασης σε περιπτώσεις αύξησης των πόρων για την κάλυψη μεγάλου αριθμού πελατών.

Οι περισσότερες εργασίες της διατριβής έλαβαν χώρα στα πλαίσια του ευρωπαϊκού προγράμματος SICMA, οι στόχοι και τα αποτελέσματα του οποίου περιγράφονται περιληπτικά στο 1ο κεφάλαιο. Ο κυριότερος τεχνικός στόχος του προγράμματος ήταν η ανάπτυξη ενός μεγάλης κλίμακας πληροφοριακού εξυπηρετητή πολυμέσων που θα είχε την δυνατότητα αποθήκευσης, διαχείρισης και μετάδοσης δεδομένων πολυμέσων σε μεγάλο αριθμό πελατών, ακολουθώντας τα πρωτόκολλα που υιοθετεί το πρότυπο DAVIC.

Η υπηρεσία διαχείρισης δεδομένων ροής προέκυψε από την ανάπτυξη και τις αλλαγές κατά την διάρκεια των δύο επιτυχημένων δημόσιων πειραματικών εφαρμογών που διεξήχθησαν στο μουσείο φυσικής ιστορίας του Λονδίνου και στο πανεπιστήμιο LUC του Βελγίου.

Εκτενής αναφορά έγινε στο 2ο κεφάλαιο στην σχετική έρευνα της περιοχής που ασχολήθηκε η παρούσα διατριβή. Εξετάστηκαν τα χαρακτηριστικά των σύγχρονων μαγνητικών δίσκων και οι επικρατέστερες πολιτικές δρομολόγησης δίσκου. Με αρκετή λεπτομέρεια παρουσιάστηκαν τα χαρακτηριστικά και οι ιδιαιτερότητες των ψηφιακών δεδομένων συνεχόμενης ροής, όπως επίσης οι τρόποι τοποθέτησης των δεδομένων πολυμέσων στις συσκευές αποθήκευσης και οι πολιτικές δρομολόγησης.

Απαραίτητη επίσης για την ανάπτυξη της υπηρεσίας ροής ήταν η συγκέντρωση βιβλιογραφικών αναφορών για τις πολιτικές δρομολόγησης των διεργασιών, τις αρχές συμπεριφοράς πραγματικού χρόνου και για τις πολιτικές μετάδοσης δεδομένων. Στην συνέχεια της σχετικής έρευνας παρατέθηκαν ζητήματα ανάπτυξης λογισμικού για εξυπηρετητές αποθήκευσης, ενώ ιδιαίτερη βαρύτητα δόθηκε στην έννοια, τις τεχνικές και τους τρόπους κλιμακωτής επέκτασης των εξυπηρετητών πολυμέσων.

Στο 3ο κεφάλαιο παρουσιάστηκε μια γενική περιγραφή της αρχιτεκτονικής και της λειτουργικότητας του πληροφορικού συστήματος Kydonia, με την βοήθεια του οποίου αποκτήθηκε σημαντική τεχνογνωσία. Με μεγαλύτερη λεπτομέρεια περιγράφηκε ο εξυπηρετητής αποθήκευσης και ο αρχικός σχεδιασμός δρομολόγησης των αιτήσεων. Στόχος του κεφαλαίου αυτού ήταν να αναδείξει τα μειονεκτήματα του αρχικού συστήματος εξυπηρέτησης πολυμέσων, που αποτέλεσε την πρώτη βάση, στην τεχνική ανάπτυξη αλλά και στον σχεδιασμό. Τα προβλήματα αυτά που αναδείχθηκαν από πειραματικές εφαρμογές του συστήματος έρχεται να καλύψει η παρούσα διατριβή, προβάλλοντας μια ολοκληρωμένη αντιμετώπιση του ζητήματος της υποστήριξης υπηρεσιών ροής σε ανοιχτά περιβάλλοντα.

Το επόμενο κεφάλαιο αποτέλεσε μια προσπάθεια για την τεχνική περιγραφή των υποσυστημάτων λογισμικού που αναπτύχθηκαν για την υποστήριξη της υπηρεσίας διαχείρισης δεδομένων ροής. Καταγράφηκαν οι ανάγκες του εξυπηρετητή ροής και διαχωρίστηκαν τα ανεξάρτητα επίπεδα εργασιών που προκύπτουν από την εξυπηρέτηση αιτήσεων αναπαραγωγής αντικειμένων συνεχόμενης ροής. Στο κεφάλαιο αυτό περιγράφηκαν το σχήμα υλικού των συσκευών αποθήκευσης, ο τρόπος προσπέλασής τους, όπως επίσης ο διαχειριστής των συσκευών αποθήκευσης, οι λειτουργίες που υποστηρίζει και ο μηχανισμός αποθήκευσης των αντικειμένων ροής από το αντίστοιχο υποσύστημα.

Οι ανάγκες προσωρινής αποταμίευσης δεδομένων του εξυπηρετητή καλύπτονται από τους διαχειριστές του αποταμιευτή επικοινωνίας, ο οποίος αποσκοπεί στην αποταμίευση των τμημάτων δεδομένων των αντικειμένων που ανακτούνται από τους δίσκους, και της προσωρινής μνήμης του συστήματος, που αποβλέπει στην ικανοποίηση της ανάγκης διαχείρισης μνήμης κατά την λειτουργία των υποσυστημάτων λογισμικού υποστήριξης.

Τέλος στο 4ο κεφάλαιο περιγράφηκε το μοντέλο διαχείρισης των αιτήσεων και υπήρξε αναφορά στο τμήμα λογισμικού υποστήριξης της επικοινωνίας και της μετάδοσης δεδομένων.

Ο σχεδιασμός και η ανάπτυξη του μηχανισμού δρομολόγησης που οδηγεί την υπηρεσία διαχείρισης δεδομένων ροής, παρουσιάστηκε με μεγάλη λεπτομέρεια στο 5ο κεφάλαιο. Ο μηχανισμός υιοθέτησε τα πλεονεκτήματα της πολυνηματικής τεχνολογίας και εφάρμοσε αρχές ανάπτυξης λογισμικού πραγματικού χρόνου. Ο προτεινόμενος μηχανισμός δρομολόγησης αποτελεί μια ολοκληρωμένη αντιμετώπιση του προβλήματος υποστήριξης χρονικά ευαίσθητων δεδομένων, καθώς αναλαμβάνει αποτελεσματικά τον συντονισμό και την σωστή εκτέλεση των διεργασιών που λαμβάνουν χώρα από την έναρξη της ανάκτησης, την προσωρινή αποταμίευση, μέχρι την μετάδοση των δεδομένων στους αιτούμενους πελάτες.

Σε αυτό το κεφάλαιο περιγράφηκε μια πληθώρα από πειράματα που διεξήχθησαν με στόχο την ανάλυση και εύρεση των απαιτήσεων εξυπηρέτησης των αντικειμένων ροής, τον ρόλο των χαρακτηριστικών υλικού των πελατών αλλά και του εξυπηρετητή στην ανάπτυξη και λειτουργία του λογισμικού του συστήματος και κυρίως την παρακολούθηση της απόδοσης των επιλογών σχεδιασμού και τεχνικής εφαρμογής που προτείνονται στα πλαίσια της μεταπτυχιακής διατριβής.

Όλα τα αποτελέσματα που προέκυψαν από την καταγραφή των πειραματικών μετρήσεων σε πελάτες και εξυπηρετητή, αξιολογούνται και συνδυάζονται. Από αυτήν την προσπάθεια προέκυψε ένα σχήμα δυναμικής δρομολόγησης που μπορεί να εφαρμοστεί στην υπηρεσία ροής, ώστε να έχουμε την μεγαλύτερη δυνατή απόδοση και την βέλτιστη αξιοποίηση των διαθέσιμων πόρων του συστήματος.

Ιδιαίτερη σημασία έχουν οι εργασίες που έλαβαν χώρα για θέματα κλιμάκωσης της υπηρεσίας διαχείρισης δεδομένων ροής και παραθέτονται στο 6ο κεφάλαιο. Ο πολυνηματικός σχεδιασμός της υπηρεσίας αποθήκευσης δεδομένων ροής έγινε με τον κατάλληλο τρόπο, ώστε να υπάρχει η δυνατότητα για αποδοτική επέκταση του συστήματος με την αύξηση των πόρων. Η εύκολη προσαρμογή του λογισμικού χωρίς χρονοβόρες ή ριζικές αλλαγές σε περιβάλλοντα διαφορετικής κλίμακας μεγέθους, είναι ζωτικής σημασίας. Ο κατάλληλος διαχωρισμός των εργασιών που επιτεύχθηκε στην υπηρεσία ροής και ιδιαίτερα η χρήση νημάτων διευκολύνουν τα σενάρια κλιμάκωσης της υπηρεσίας σε διάφορα περιβάλλοντα.

Στο 6ο κεφάλαιο παρουσιάστηκαν τρία διαφορετικά σενάρια και τρόποι για αποδοτική επέκταση του συστήματος της διατριβής. Το πρώτο αναφέρεται στην εφαρμογή του λογισμικού σε πολυεπεξεργαστικό σύστημα διαμοιραζόμενης μνήμης, το δεύτερο σε

περιβάλλοντα παράλληλων μηχανών ευρείας κλίμακας και τέλος εξετάστηκε πιο διεξοδικά η περίπτωση επέκτασης του συστήματος σε διανεμημένο περιβάλλον, χρησιμοποιώντας ομάδες από σταθμούς εργασίας.

9.1 Μελλοντικές επεκτάσεις

Η ανάπτυξη της υπηρεσίας διαχείρισης δεδομένων ροής έδειξε πειραματικά θετικά αποτελέσματα στην προσπάθεια αντιμετώπισης των ευαίσθητων στις χρονικές καθυστερήσεις αντικειμένων. Η υπηρεσία ροής εφαρμόστηκε σε σταθμούς εργασίας Sparc20 και Ultra1 με λειτουργικό σύστημα Solaris 2.5.x .

Ένας από τους στόχους των μελλοντικών επεκτάσεων είναι η μεταφορά της αρχιτεκτονικής που σχεδιάστηκε και η αποτελεσματική εφαρμογή της σε άλλες πλατφόρμες και ιδιαίτερα στην πιο δημοφιλή Intel – Windows NT. Μια τέτοια προσπάθεια προϋποθέτει την λεπτομερή μελέτη των υπηρεσιών και των διαφορετικών χαρακτηριστικών των δύο λειτουργικών συστημάτων, όσον αφορά την πολυνηματική εφαρμογή και τις δυνατότητες δρομολόγησης πραγματικού χρόνου.

Μια άλλη σημαντική μελλοντική επέκταση που είναι απαραίτητη είναι η ανάπτυξη και η υιοθέτηση των πρωτοκόλλων μετάδοσης δεδομένων πραγματικού χρόνου, που έχουν προταθεί για την αποστολή δεδομένων συνεχόμενης ροής και την μεταφορά μηνυμάτων ελέγχου μέσω διαδικτύου (Multimedia over IP, RTP, RTSP κ.α.).

Πολλά ζητήματα είναι επίσης εκείνα που μπορούν να αποτελέσουν αντικείμενο μελλοντικής μελέτης και έρευνας στον τομέα της κλιμακωτής επέκτασης της υπηρεσίας ροής. Στην παρούσα διατριβή προτάθηκε η δυνατότητα κλιμάκωσης του συστήματος με την χρήση στιγμιότυπων εξυπηρετητών σε διανεμημένο επίπεδο. Η συνεργασία των εξυπηρετητών αυτών επιτεύχθηκε με τον κατάλληλο συγχρονισμό των χρονικών κύκλων λειτουργίας αυτών.

Πειράματα έδειξαν ότι ο παραπάνω σχεδιασμός της επέκτασης μπορεί να λειτουργήσει ικανοποιητικά, αλλά υπάρχει ανάγκη για βαθύτερη μελέτη και πειραματισμό. Πιο πολύπλοκη αντιμετώπιση χρειάζεται πιθανώς όταν είναι αναπόφευκτη η χρήση εξυπηρετητών διαφορετικών δυνατοτήτων, καθώς απαιτείται ιδιαίτερη προσοχή στον τρόπο τοποθέτησης των δεδομένων σε επίπεδο δικτύου και στην ανάπτυξη του μηχανισμού ισορροπίας του φόρτου εργασίας στους κόμβους-εξυπηρετητές του διανεμημένου συστήματος.

9.2 Συμπεράσματα – Συνεισφορά

Η αποδοτική διαχείριση δεδομένων πολυμέσων αποτελεί βασική ανάγκη για την υποστήριξη των σύγχρονων εφαρμογών. Η διατριβή έχει να παρουσιάσει την ανάπτυξη ενός πραγματικού συστήματος, το οποίο είναι ικανό να παρέχει υπηρεσίες δεδομένων ροής σε μεγάλο αριθμό πελατών.

Πειράματα έδειξαν ότι η μετάδοση δεδομένων από τον εξυπηρετητή εμφανίζει αυξημένες επεξεργαστικές ανάγκες, αναδεικνύοντας τον επεξεργαστή σε πιθανό σημείο συμφόρησης του συστήματος. Επίσης η τεχνολογία και ο εξοπλισμός των πελατών έχει άμεσες επιπτώσεις στην δυνατότητά τους για λήψη δεδομένων, με αποτέλεσμα ετερογενείς πελάτες να δημιουργούν διαφορετικές απαιτήσεις στον εξυπηρετητή.

Η ανάπτυξη του συστήματος ακολούθησε τον σχεδιασμό της υπηρεσίας ροής που προϋποθέτει χρήση της πολυνηματικής τεχνολογίας και εφαρμόζει αρχές προγραμματισμού πραγματικού χρόνου. Ο μηχανισμός δρομολόγησης των επιμέρους εργασιών (επεξεργασίας αιτήσεων, ανάκτησης, αποταμίευσης και μετάδοσης δεδομένων) που προέκυψε, αποδείχθηκε ικανός πειραματικά να συντονίζει κατάλληλα την λειτουργία του συστήματος παρέχοντας εγγυήσεις ποιότητας υπηρεσιών.

Η υποστήριξη ετερογενών πελατών σε ανοιχτά περιβάλλοντα, έγινε δυνατή μέσω του αυτοπροσαρμοζόμενου δρομολογητή, ο οποίος συγκεντρώνει τα χαρακτηριστικά των πελατών που ζητούν υπηρεσίες από το σύστημα και παράγει ένα σχήμα δρομολόγησης που εξαρτάται από τα χαρακτηριστικά των ενεργών πελατών. Τα ισχυρά συστήματα και ο εξοπλισμός που διαθέτουν πλέον οι σύγχρονοι πελάτες, μπορούν να αξιοποιηθούν από τον εφαρμοζόμενο μηχανισμό δρομολόγησης είτε για την ενεργοποίηση μηχανισμών προανάκτησης, είτε για την μείωση των απαιτήσεων προσωρινής αποταμίευσης.

Πειράματα που έλαβαν χώρα έδειξαν ότι ο εξυπηρετητής ροής αξιοποίησε σε μεγάλο βαθμό τους διαθέσιμους πόρους του συστήματος για να εμφανίσει την μέγιστη απόδοση που το υλικό ήταν σε θέση να υποστηρίξει.

➤ ΠΑΡΑΡΤΗΜΑ Α

Ανάπτυξη πειραματικής εφαρμογής διαδικτύου

Στο παράρτημα αυτό περιγράφεται η εφαρμογή διαδικτύου που αναπτύχθηκε για λόγους πειραματισμού, αξιοποίησης αλλά και επιβεβαίωσης της ορθής λειτουργίας της υπηρεσίας αποθήκευσης δεδομένων συνεχόμενης ροής. Η εφαρμογή αυτή εκμεταλλεύεται τις υπάρχουσες τεχνολογίες διαδικτύου για την παρουσίαση πολυμέσων (video και ήχου) στην πλευρά του πελάτη σε συνεργασία με τον εξυπηρετητή πολυμέσων .

To Java Media Framework

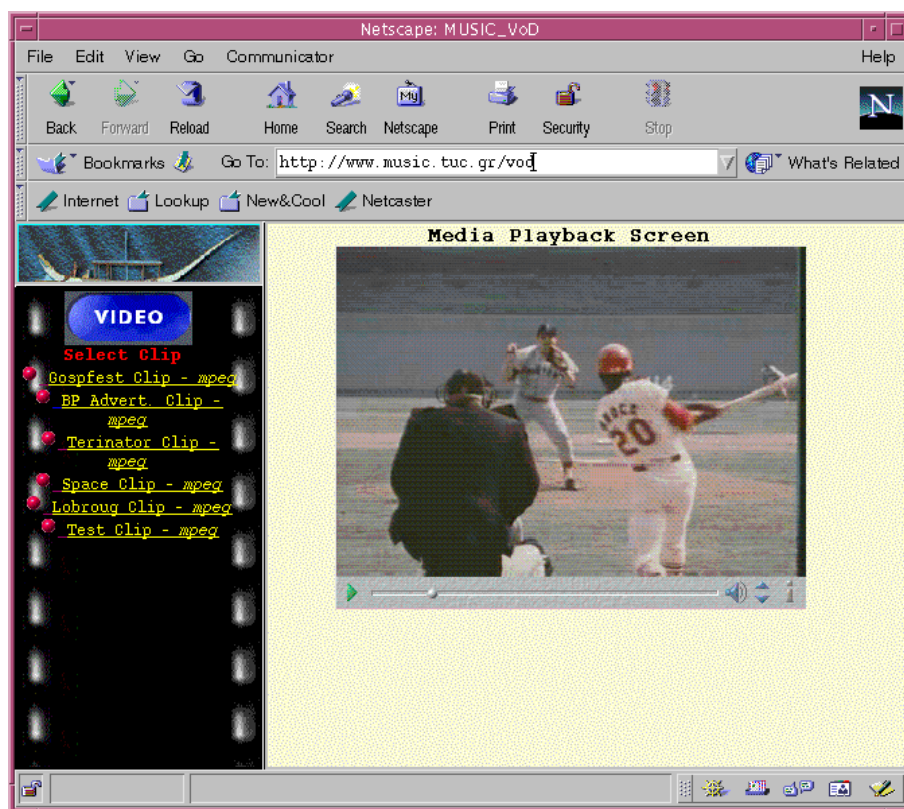
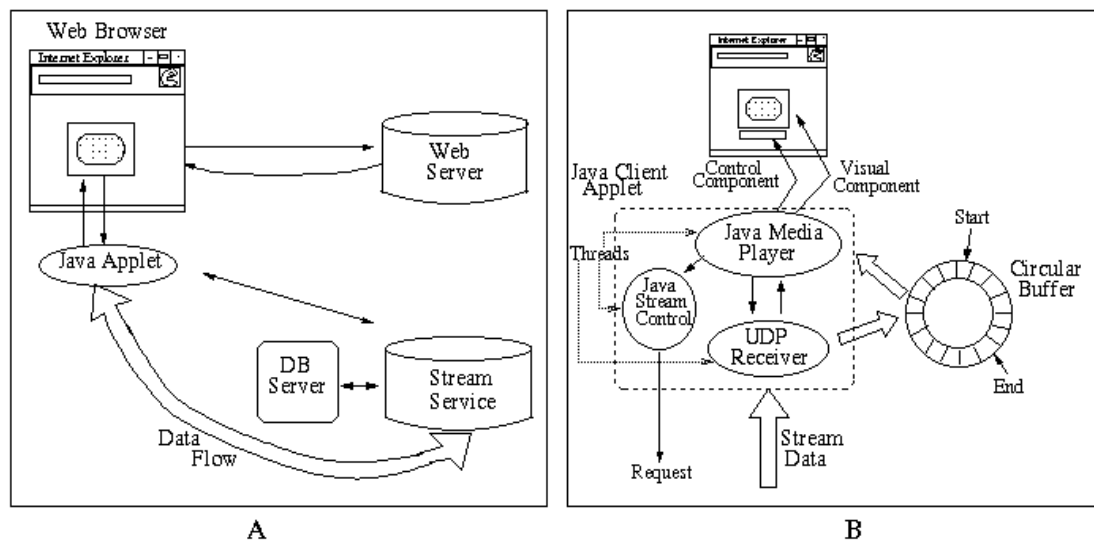
Το Java[™] Media Framework API (JMF) είναι μια συλλογή από κλάσεις που καθιστούν δυνατή την παρουσίαση δεδομένων πολυμέσων μέσα σε εφαρμογές Java ή applets. Το JMF καθορίζει μια ενοποιημένη αρχιτεκτονική, ένα πρωτόκολλο μηνυμάτων και ένα σύστημα προγραμματισμού για όλες τις ενέργειες (playback, capture, conferencing) πάνω σε συμπιεσμένα δεδομένα ήχου, video και MIDI σε όλες τις πλατφόρμες υποστήριξης της τεχνολογίας Java. Το JMF είναι ένα μέρος του Java Media API, που αποτελείται από ένα σύνολο από βιβλιοθήκες κλάσεων που παρέχουν δυνατότητες σε περιοχές όπως 2D και 3D γραφικά, animation, αναγνώριση και σύνθεση φωνής, ψηφιακή τηλεφωνία, συνεργασία μέσω υπολογιστών και παρουσίαση πολυμέσων, με αποτέλεσμα να εμπλουτίζονται οι Java εφαρμογές με ισχυρές ικανότητες επικοινωνίας και διαχείρισης μέσων.

Το JMF προσφέρεται σε τρία πακέτα: Player, Capture και Conference. Το πρώτο πακέτο, που χρησιμοποιήθηκε για την ανάπτυξη της εφαρμογής του παραρτήματος, παρέχει ένα πλαίσιο εργασίας ανεξάρτητο της πλατφόρμας. Έχει σχεδιαστεί για να υποστηρίζει πολλούς τύπους μέσων, συμπεριλαμβανομένων των MPEG1, MPEG2, Quick Time, AVI, WAV, AU και MIDI.

Χρησιμοποιώντας τους Java Media Players ο προγραμματιστής είναι σε θέση να συγχρονίσει και να παρουσιάσει χρονικά ευαίσθητα μέσα από διαφορετικές πηγές.

Περιγραφή της εφαρμογής

Στο σχήμα που ακολουθεί περιγράφεται η λειτουργία της εφαρμογής που αναπτύχθηκε. Η πρόσβαση του χρήστη επιτρέπεται μέσω HTML σελίδων που έχουν δημιουργηθεί στον εξυπηρετητή ιστού (web server) και εμφανίζονται στο σχήμα Α. Μετά την επιλογή του αντικειμένου video που ο χρήστης επιθυμεί την αναπαραγωγή του, ο εξυπηρετητής ιστού μεταφέρει στον πελάτη ένα java applet, μέσω του οποίου γίνεται δυνατή η επικοινωνία του πελάτη με τον εξυπηρετητή ροής



C

Χρησιμοποιώντας το JMF δίνουμε την δυνατότητα της παρουσίασης πολυμέσων στην πλευρά του πελάτη μέσα στον browser που χρησιμοποιεί. Οι αιτήσεις δημιουργούνται ύστερα από την κατάλληλη αλληλεπίδραση του χρήστη και αποστέλλονται στον εξυπηρετητή με την βοήθεια του TCP/IP πρωτοκόλλου. Στην συνέχεια ο εξυπηρετητής επεξεργάζεται την αίτηση και δημιουργεί ένα μονοπάτι ροής δεδομένων προς τον πελάτη που τρέχει το java applet. Στην εικόνα B του σχήματος μπορεί κανείς να διακρίνει πιο αναλυτικά την λειτουργία του applet. Η εφαρμογή δημιουργεί τρία νήματα, τα οποία είναι

υπεύθυνα για την λήψη των δεδομένων που αποστέλλει ο εξυπηρετητής, την αποταμίευση των δεδομένων, την παρουσίαση του μέσου στις συσκευές εξόδου του πελάτη και για την αλληλεπίδραση με τον χρήστη. Η μεταφορά των δεδομένων μεταξύ των νημάτων λήψης δεδομένων και παρουσίασης γίνεται με την χρήση ενός κυκλικού προσωρινού αποταμιευτή.

➤ Παράρτημα Β

Στο σχήμα εμφανίζεται η εφαρμογή αλληλεπίδρασης με την υπηρεσία διαχείρισης δεδομένων ροής.



Η εφαρμογή αυτή αναπτύχθηκε χρησιμοποιώντας την τεχνολογία Java, ενώ για την σύνδεσή της με το λογισμικό του εξυπηρετητή έγινε χρήση του Java Native Interface (JNI). Μέσω αυτής της εφαρμογής είναι δυνατόν ο χρήστης να εκτελέσει ένα αριθμό από λειτουργίες πάνω στον εξυπηρετητή ροής, όπως έναρξη, παύση και τερματισμός λειτουργίας της υπηρεσίας διαχείρισης δεδομένων ροής. Μέσω ειδικών παραθύρων είναι δυνατή η ρύθμιση των παραμέτρων λειτουργίας του εξυπηρετητή και η εισαγωγή των συσκευών υλικού υποστήριξης του συστήματος. Επίσης η εφαρμογή οδηγεί την αποθήκευση νέων αντικειμένων ροής στο σύστημα (κυρίως μέσω CD-ROM).

Το χαρακτηριστικό της εφαρμογής χρήσης του εξυπηρετητή είναι ότι παρέχει ένα σύνολο από παράθυρα παρακολούθησης, που εικονίζουν γραφικά διάφορες παραμέτρους που έχουν σχέση με την απόδοση της υπηρεσίας ροής. Έτσι σε πραγματικό χρόνο μπορούμε να παρακολουθούμε συνεχώς πληροφορίες, όπως τον αριθμό των ενεργών πελατών που εξυπηρετούνται κάθε χρονική στιγμή, τον φόρτο εργασίας των συσκευών αποθήκευσης, τα ποσοστά χρήσης του επεξεργαστή, καθώς και την διαθέσιμη μνήμη στον αποταμιευτή επικοινωνίας.

➤ Παράρτημα Γ

Στο παράρτημα αυτό περιγράφεται η εφαρμογή που τρέχει στην πλευρά του πελάτη ώστε να ενημερωθεί ο κεντρικός δρομολογητής της υπηρεσίας ροής για τα χαρακτηριστικά και τις δυνατότητες λήψης και αποταμίευσης δεδομένων του πελάτη .

The screenshot shows a software window titled 'Form' with a menu bar containing 'File' and 'Options'. The window is divided into several sections:

- Client Stream Configuration Application:** Includes a computer icon and a 'Stream Server' section with fields for 'IP Address' (147.27.1.22) and 'Port No' (8000).
- Client Communication:** Includes fields for 'IP Address' (147.27.1.125), 'Port No' (5500), 'Computer Name' (hermes), and 'Packet Size' (4096).
- Client Configuration:** Includes fields for 'Buffer Size (MBytes)' (32), 'CPU Type' (Pentium II 450 M), 'Network Type' (Ethernet 10Mbps), 'Disk Size (MBytes)' (1024), and 'Decoder' (Software).
- Monitor:** Displays various performance metrics with progress bars:
 - Packet Loss: 4.5936395
 - Arr. Bit Rate (Mbps): 5.140625
 - Total CPU usage: 73
 - App. CPU usage: 7
 - Frame Rate: 30 (with a scale from 1 to 30)A 'Stop Configuration Test' button is located below these metrics.
- Video Feed:** A large window showing a live video stream of a person in a workshop.
- Status Bar:** A blue bar at the bottom displaying the text 'hermes 3 2 0 147.27.1.125 5500 4096'.

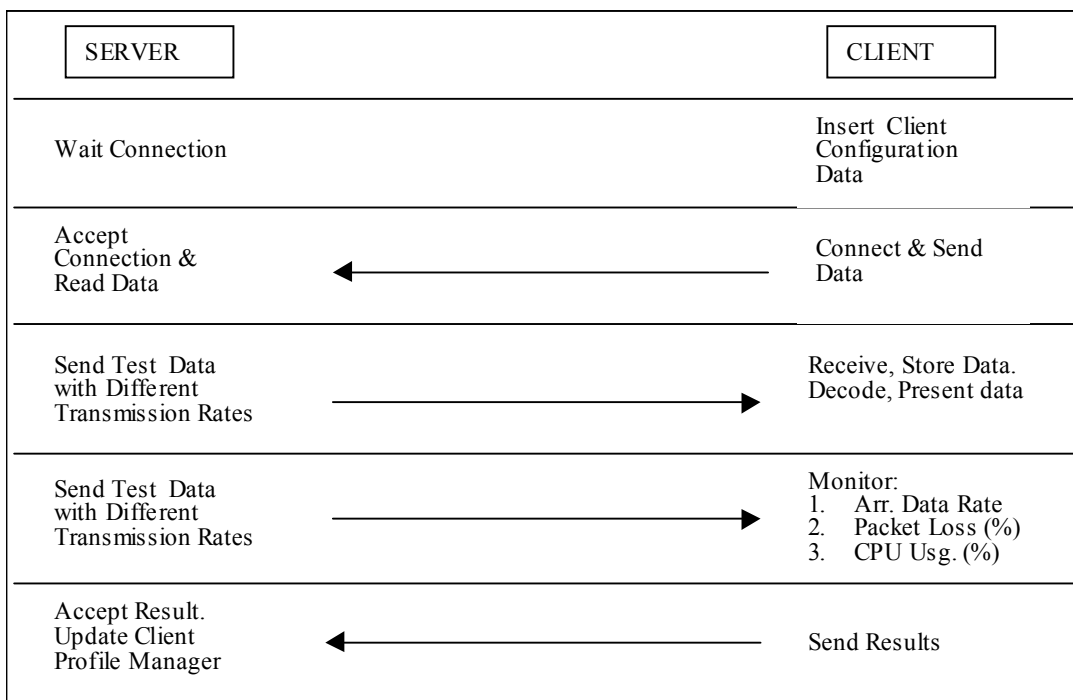
Κάθε πελάτης που επιθυμεί να τύχει εξυπηρέτησης, πρέπει να ακολουθήσει την απαραίτητη διαδικασία εγγραφής στο σύστημα. Έτσι ο πελάτης, όπως έχει ήδη αναφερθεί, τρέχει τοπικά μια ειδική εφαρμογή μέσω της οποίας ο δρομολογητής δέχεται τα τεχνικά χαρακτηριστικά του πελάτη (είδος επεξεργαστή, μέγεθος προσωρινής αποταμίευσης, μέγεθος δευτερεύουσας μνήμης, δικτυακή υποδομή, εξοπλισμό αναπαραγωγής δεδομένων ροής κ.α.), ενώ το πιο σημαντικό πλεονέκτημα είναι ότι με την βοήθεια της εφαρμογής διαπιστώνονται σε πραγματικό περιβάλλον οι δυνατές επιδόσεις του πελάτη με τις τρέχουσες ρυθμίσεις. Συνεπώς, όταν η διαδικασία εγγραφής για κάθε πελάτη ολοκληρώνεται, ο διαχειριστής πελατών (Client Profile Manager) ενημερώνεται με νέα στοιχεία, όπως κωδικός νέου πελάτη, μέγιστη δυνατότητα ρυθμού λήψης δεδομένων και διαθέσιμος χώρος αποταμίευσης δεδομένων στην πρωτεύουσα και δευτερεύουσα μνήμη.

Το παράθυρο της εφαρμογής χωρίζεται σε τέσσερα μέρη, στα τρία από τα οποία εισάγονται σε ειδικά πεδία πληροφορίες από τον χρήστη, ενώ στο τέταρτο απεικονίζονται όλες οι παράμετροι που συνιστούν την απόδοση του πελάτη και γίνεται η παρουσίαση του μέσου.

Ο χρήστης πρέπει να εισάγει, όπως μπορεί κανείς να παρατηρήσει στο παραπάνω σχήμα, όλα τα στοιχεία που είναι απαραίτητα για την επικοινωνία των πελατών με τον εξυπηρετητή ροής (IP, portno), το αναγνωριστικό όνομα του πελάτη, καθώς και το μέγιστο θεμιτό μέγεθος των πακέτων δεδομένων μετάδοσης. Επίσης απαραίτητη είναι η εισαγωγή των τρέχουσων ρυθμίσεων του πελάτη, δηλαδή το μέγεθος της πρωτεύουσας και δευτερεύουσας μνήμης, ο τύπος του επεξεργαστή, η δικτυακή του υποδομή και ο τύπος του αποκωδικοποιητή MPEG (λογισμικό ή ειδικό υλικό).

Ο πελάτης καθορίζει την έναρξη του πειραματικού ελέγχου, σύμφωνα με τον οποίο αρχίζει η αναπαραγωγή των δεδομένων, ενώ ταυτόχρονα ο πελάτης λαμβάνει πακέτα από τον εξυπηρετητή και τα αποταμιεύει στους κατάλληλους αποταμιευτικούς χώρους. Κατά την διάρκεια της παραπάνω διαδικασίας καταγράφονται και παρουσιάζονται γραφικά οι μετρήσεις σχετικά με το ποσοστό απώλειας πακέτων, τον τρέχων ρυθμό λήψης δεδομένων, το ποσοστό χρήσης του επεξεργαστή από την εφαρμογή του πελάτη και το ολικό ποσοστό απασχόλησης του επεξεργαστή.

Απώτερος στόχος της εφαρμογής είναι η εύρεση του ιδανικού ρυθμού μετάδοσης δεδομένων, ώστε να αποφεύγονται οι απώλειες δεδομένων και να μην επηρεάζεται ο ρυθμός παρουσίασης των μέσων ροής (βλ. πειράματα της ενότητας 5.4). Στην συνέχεια του παραρτήματος περιγράφεται ο τρόπος αλληλεπίδρασης της εφαρμογής με την υπηρεσία ροής και η διαδικασία εγγραφής του πελάτη στο σύστημα.



➤ Βιβλιογραφία

1 [Reum94]	“An Introduction to Disk Drive Modeling” Chris Reummler & John Wilkes. IEEE Computer March 1994
2 [Worth94]	“Scheduling for Modern Disk Drives and Non-Random Workloads” B.L. Worthington, G. Ganger, Y. Patt, U.Michigan Technical Report CSE-TR-194-94.March 1994
3 [Jacob91]	“Disk Scheduling algorithms based on rotational position” D. Jacobson, John Wilkes. HP technical report , March 1991
4 [Denn67]	“Effects of scheduling on file memory operations” P. Denning. AFIPS Spring Joint Computer Conference, April 1967
5 [Mert70]	“Some quantitive techniques for file organization” A. Mergen. PhD Thesis Univ. of Wisconsin,1970
6 [Geis87]	“Disk Scheduling in System V” R.Geist, R. Reynolds, E. Pittard. Performance evaluation review , May 1987
7 [Gem92]	“Principles of Delay Sensitive Myltimedia Data Storage Servers” D. Gemmel, S. Christodoulakis. ACM Transactions on Information Systems, Vol. 10, No. 1, 1992
8 [Gem94a]	“Multimedia Storage Servers: A Tutorial and Survey” D. Gemmel. IEEE Computer, March 1994
9 [Vin93]	“Designing a Multi-User HDTV Storage ServerL” H. Vin, P. Rangan. Journal on selected areas in communications, January 1993
10 [Yu93]	“Grouped Sweeping Scheduling for DASD-based multimedia storage management” P.Yu, M. Chen,D. Kandlur. Multimedia Systems Journal 1993
11 [Ghan95]	“On Disk Scheduling and Data Placement for Video Servers” S. Ghandeharizadeh, S. Kim, C. Sharabi. December 1995
12 [Gem94b]	“Multimedia Network File Servers: Multi-channel Delay Sensitive Data Retrieval” J. Gemmel, J. Han. Multimedia Systems Vol1 ,No6 1994
13 [Ozd94]	“On Storage and Retrieval of Continous Media Data” B. Ozden, R. Rastogi, A. Silberchatz. 1994
14 [Red94]	“ I/O Issues in a Multimedia System” A.L. Narasimha Reddy, J. Wyllie. IEEE Computer. March 1994

15 [Gan95]	“Techniques to Quantify SCSI-2 Disk Subsystem Specifications for Multimedia” S. Ghandeharizadeh, J. Stone, R. Zimmermann. March 1995
16 [Chun96]	“Multimedia Information Storage and Management”, S. Chung. ISBN 0-7923-9764-9. 1996
17 [Vin94]	“Analyzing the Performane of Asynchronous Disk Arrays for Multimedia Retrieval” H. Vin, P. Shenoy, S Rao. Proceedings of the ISMM International Conference on Distributed Multimedia Systems and Applications, August 1994
18 [Ber94]	“Staggered Striping in Multimedia Information System” S. Berson, R. Muntz, S. Ghandeharizadeh, X. Ju. SIGMOD May 1994
19 [Ozd95]	“Disk Striping in Video Server Environments” B. Ozden, R. Rastogi, A. Silberschatz. 1995
20 [Zim97]	“Continuous Display Using Heterogeneous Disk-Subsystems” R. Zimmermann, S. Ghandeharizadeh. ACM Multimedia November 1997
21 [Tob93]	“Streaming RAID: A Disk Storage System for Video and Audio Files” Proceedings Multimedia August 1993
22 [Tan87]	“Operating Systems : Design and Implementation” A. Tanenbaum. ISBN 0-13-637331-3. 1987
23 [Cor96]	“Real Time CORBA” Object Management Group’s Real-Time Special Interest Group. White paper November 1996
24 [Goy96]	“A Hierarchical CPU Scheduler for Multimedia Operating Systems” P. Gyal, X. Guo, H. Vin. Proceeding of the 2 nd USENIX Symposium on Operating Systems Design and Implementaion (OSDI), October 1996
25 [Jon97]	“CPU Reservations and Time Constraints: Efficient, Predictable Scheduling of Independent Activities” M. Jones, D. Rosu, M Rosu. SIGOPS’97 Saint-Malo France, November 1997
26 [Stil95]	“Latency-Rate Servers: A General Model for Analysis of traffic Scheduling Algorithms” D. Stiliadis, A. Varma. Technical Report University of California Santa Cruz, July 1995
27 [Kalm90]	“Rate Controlled Servers for Very High-Speed Networks” C. Kalmanek, H. Knakia. IEEE Global Telecommunications Conference, December 1990
28 [Zha]	“RCSP and Stop-and_Go: A Commparison of two Non-Work-Conserving Disciplines fro Supporting Multimedia Communication” H. Zhang, E. Knightly. To apper in ACCMSpringer Multimedia Systems Journal
29 [His94]	“Performance of a Mass Storage System for Video-On-Demand” J.

	Hsieh, M. Lin, C. Liu, D. Du. Submitted to Journal of Parallel and Distributed Computing: Special Issue on Multimedia Processing Technology
30 [Eff84]	"Principles of Database Buffer Management" W. Efflesberg T. Haerder. ACM Transactions on Database Systems, Vol. 9 December 1984
31 [And92]	"A File System for Continuous Media" D.P. Anderson, Y. Osawa, R. Govindan. ACM Transactions on Computer Systems Vol. 10 November 1992
32 [Chris91]	"An Object-Oriented Architecture for Multimedia Information Systems", S. Christodoulakis et al . The Quarterly Bulletin of the IEEE Computer Society Technical Committee on Data Engineering, Vol.14 September 1991
33 [Ma97]	"Efficient Real-Time Data Retrieval Through Scalable Multimedia Storage" G. Ma, C. Wu, M. Liu, B. Lin. ACM Multimedia '97 Proceedings, November 1997
34 [Lee95]	"Highly-Available, Scalable Network Storage" E. Lee. COMPCON '95, 1995
35 [Tez95]	"Elements of Scalable Video Servers" W. Tezlaff, R. Flynn. COMPCON'95, 1995
36 [Loug94]	"Scalable Storage Servers for Digital Audio & Video" P. Lougher, D. Pegler, D. Shepherd. IEEE International Conference on Storage and Recording Systems, 1994
37 [Ghan97]	"Mitra: A Scalable Continuous Media Server" S. Ghandharizadeh, R. Zimmermann, W. Shi, R. Rejaie, D. Ierardi, T. Li. To appear in Multimedia Tools and Applications Journal, January 1994
38 [Peg95]	"Scalability Issues for Networked Multimedia Storage Architecture" D. Pegler, D. Hutchison, P. Lougher, A. Scott, D. Shepherd. Technical Report Lancaster University, 1995
39 [Kor95]	"Scalability in Distributed Multimedia Systems" M. Korkea-aho. Master's Thesis Helsinki University of Technology, 1995
40 [Lal95]	"A protocol for Scalable Group and Public Annotation" D. Laliberte, A. Braverman. Proceedings of the Third International World-Wide Web Conference, 1995
41 [Bol97]	"Distributed Schedule Management in the Tiger Video Fileserver" W. Bolosky, R. Fitzgerald, J. Douceur. Proceedings of the 16 th ACM Symposium of Operating System Principles, October 1997

42 [Sand96]	“Design and Implementation of the Elvira Video Server” O. Sandsta, S. Langorgen, R. Midtstraum. 1996
43 [Tetz95]	“Elements of Scalable Video Servers”, W. Tetzlaff, R. Flynn
44 [Bern95]	“A Scalable Video Server: Architecture, Design, and Implementation” C. Bernhardt, E. Biersack. Realtime Systems Conference Paris, January 1995
45 [Gafs98]	“Design and Implementation of a Scalable, Reliable, and Distributed VOD-Server” J. Gafsi, U. Walther, E. Biersack. Proceedings of the 5 th joint IFIP and ICCS Conference on Computer Communications, AFRICOM-CCDC, Tunis 1998
46 [DAV95]	“Description Of DAVIC Functionalities” Digital Audio-Visual Council, DAVIC 1.0 Specification Part 1, 1995
47 [AKL96]	“The Design and Analysis of Parallel Algorithms” Selim G. AKL Prentice-Hall International Editionw ISBN:0-13-200073-3
48 [Ritc78]	“The UNIX Time-Sharing System: A Retrospective” Ritchie D. ,The Bell System Technical Journal, 1978
49 [SunM92]	“Multiprocessing Technology: A Primer”, Sun Microsystems, Software Technical Bulletin, March 1992
50 [MK85]	“Reseach On Parallel Machine Architecture for Fifth Generation Computer Systems”, Murakami K. , T. Kakuta, R. Onai, Computer v18 n6.
51 [Bart87]	“Can UNIX take the Plunge?”, J.M. Barton, UNIX Review, v5 n10.
52 [Gape]	“Essential Issues in Multiprocessor Systems”, Computer, v18 n6.
53 [ChMol97]	“Effective Memory Use in a Media Server”, E. Chang & G. Molina, VLDB Conference 1997
54 [Fren86]	“Evaluating Two Massively Parallel Machines”, K. Frenkel, Communications of the ACM, v29 n8, 1986