



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

**ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΕΠΙΚΟΙΝΩΝΙΩΝ ΗΛΕΚΤΡΟΝΙΚΗΣ ΚΑΙ ΣΥΣΤΗΜΑΤΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ**

**Ανάπτυξη μεθοδολογίας Εκτιμητή Φόρτου Εργασίας σε
περιβάλλον υπολογιστικού πλέγματος**

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

ΕΜΜΑΝΟΥΗΛ Χ. ΛΕΥΚΙΔΗ

Επιβλέπουσα: Θεοδώρα Βαρβαρίγου
Καθηγήτρια Ε.Μ.Π.

Αθήνα, Ιούνιος 2008



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΕΠΙΚΟΙΝΩΝΙΩΝ ΗΛΕΚΤΡΟΝΙΚΗΣ
ΚΑΙ ΣΥΣΤΗΜΑΤΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ

Ανάπτυξη μεθοδολογίας Εκτιμητή Φόρτου Εργασίας σε περιβάλλον υπολογιστικού πλέγματος

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

ΕΜΜΑΝΟΥΗΛ Χ. ΛΕΥΚΙΔΗ

Επιβλέπουσα: Θεοδώρα Βαρβαρίγου
Καθηγήτρια Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 29^η Μαΐου 2008.

.....
Θεοδώρα Βαρβαρίγου
Καθηγήτρια Ε.Μ.Π.

.....
Βασίλειος Λούμος
Καθηγητής Ε.Μ.Π.

.....
Ελευθέριος Καγιάφας
Καθηγητής Ε.Μ.Π.

Αθήνα, Ιούνιος 2008

.....
EMMANΟΥΗΛ Χ. ΛΕΥΚΙΔΗΣ

Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π.

Copyright © EMMANΟΥΗΛ Χ. ΛΕΥΚΙΔΗΣ, 2008
Με επιφύλαξη παντός δικαιώματος. All rights reserved

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τον συγγραφέα.

Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τον συγγραφέα και δεν πρέπει να ερμηνευθεί ότι αντιπροσωπεύουν τις επίσημες θέσεις του Εθνικού Μετσόβιου Πολυτεχνείου.

Περίληψη

Ο σκοπός της διπλωματικής εργασίας ήταν η μελέτη και η ανάπτυξη μεθοδολογίας για την εκτίμηση του φόρτου εργασίας σε περιβάλλον υπολογιστικού πλέγματος (Grid). Ειδικότερα η μεθοδολογία αυτή ενσωματώθηκε και υλοποιήθηκε σε ένα ολοκληρωμένο μηχανισμό εκτίμησης για την περίπτωση του 3D Rendering (επεξεργασία τρισδιάστατων γραφικών) μια εφαρμογή πολύ συνηθισμένη στα εμπορικά πλέγματα. Για το σκοπό αυτό πραγματοποιήθηκαν μετρήσεις σε εργασίες rendering για να αξιολογηθεί η μέθοδος.

Συγκεκριμένα η μεθοδολογία βασίστηκε σε τεχνικές δειγματοληψίας και την διεκπεραίωση δείγματος της εργασίας με σκοπό την άντληση πληροφοριών φόρτου για ολόκληρη την εργασία. Επίσης στα πλαίσια της εργασίας μελετήθηκαν τεχνικές βελτιστοποίησης της εκτίμησης φόρτου εργασίας. Οι τεχνικές βελτιστοποίησης βασίστηκαν στην αξιοποίηση της εμπειρίας του παρελθόντος από εκτιμήσεις προηγούμενων εργασιών. Για τον λόγο αυτό ενσωματώθηκε στον μηχανισμό ένα έμπειρο σύστημα ‘συλλογισμού βασισμένο σε περιπτώσεις’ (Case-Based Reasoning, CBR).

Γενικότερα διαπιστώθηκε ότι η εκτίμηση του φόρτου εργασίας μπορεί να βελτιώσει την κατανομή και την διαχείριση των υπολογιστικών πόρων καθώς και να εξασφαλίσει το επίπεδο της ποιότητας μιας υπηρεσίας ιστού (QoS) που προσφέρει το υπολογιστικό πλέγμα. Έτσι βοηθάει στην δημιουργία πιο αξιόπιστων συμφωνιών (SLA) μεταξύ του παρόχου της υπηρεσίας και του πελάτη που επιθυμεί να χρησιμοποιήσει μια υπηρεσία πλέγματος.

Λέξεις Κλειδιά: φόρτος εργασίας, εκτιμητής φόρτου εργασίας, rendering, Case-Based Reasoning, υπηρεσίες ιστού, υπολογιστικό πλέγμα, SLA

Abstract

The aim of our thesis was the research and the development of a methodology on the estimation of workload in Grid environment. Specifically this methodology was incorporated and was implemented in a completed mechanism of estimation for the case of the 3D Rendering, an application very common in the commercial Grids. For the evaluation of this method measurements were realized in rendering works.

Concretely the methodology was based on techniques of sampling and on the transaction of sample of work aiming at the collection of information of workload on the entire work. Also in the scope of this research studied techniques of optimizing the estimation of workload. The techniques of optimization were based on the exploitation of past experience from previous works. For this reason was incorporated in the mechanism a Case-Based Reasoning (CBR) expert system.

Generally it was realized that the estimation of workload can improve the allocation and the management of Grid resources and can ensure the level of quality of service (QoS) of the web-services that offers the Grid. Thus it helps in the creation of more reliable agreements (SLA) between the supplier of the service and the customer that wants to use a Grid service.

Keywords: workload, workload estimator, rendering, Case-Based Reasoning, web-service, Grid, SLA

Ευχαριστίες

Στο σημείο αυτό θα ήθελα να ευχαριστήσω την επιβλέπουσα καθηγήτρια κ. Θεοδώρα Βαρβαρίγου για την δυνατότητα που μου έδωσε να εκπονήσω μία τόσο ενδιαφέρουσα διπλωματική εργασία και τον υποψήφιο διδάκτορα Ανδρέα Μενύχτα για την πολύτιμη καθοδήγησή του και την άριστη συνεργασία μας. Επίσης θέλω να ευχαριστήσω τους γονείς μου για την στήριξη και την βοήθεια που μου παρείχαν κατά την διάρκεια των σπουδών μου.

Αφιερωμένη στη γιαγιά μου

Πίνακας περιεχομένων

1	Εισαγωγή.....	16
1.1	Επισκόπηση της Υπηρεσιοστραφούς Αρχιτεκτονικής, των Υπηρεσιών Ιστού και των Υπολογιστικών Πλεγμάτων.....	17
1.1.1	Υπηρεσιοστραφής Αρχιτεκτονική (Service Oriented Architecture).....	17
1.1.2	Υπηρεσίες δικτύου (Web services).....	19
1.1.3	Υπηρεσιοστραφής Αρχιτεκτονική και Υπηρεσίες Ιστού: Ύψος εναντίον της εφαρμογής.....	20
1.1.4	Υπολογιστικά Πλέγματα (Grids).....	20
1.2	Εκτίμηση φόρτου εργασίας.....	24
1.2.1	Γενικά.....	24
1.2.2	Φόρτος εργασίας υπολογιστικών συστημάτων.....	24
1.2.3	Εκτίμηση φόρτου εργασίας σε περιβάλλον υπολογιστικών πλεγμάτων.....	25
1.2.4	Ποιότητα Υπηρεσίας (Quality of Service).....	27
1.3	Αντικείμενο διπλωματικής.....	29
1.4	Οργάνωση κειμένου.....	29
2	Σχετικές εργασίες.....	31
3	Θεωρητικό υπόβαθρο.....	33
3.1	Στατιστική και δειγματοληψία.....	33
3.1.1	Δειγματοληψία.....	34
3.2	Αρχιτεκτονική των έμπειρων συστημάτων.....	36
3.2.1	Ιδιότητες ενός έμπειρου συστήματος.....	37
3.2.2	Δομή ενός έμπειρου συστήματος.....	37
3.2.3	Η βάση της γνώσης.....	37
3.2.4	Η μηχανή εξαγωγής συμπερασμάτων.....	38
3.3	Έμπειρο σύστημα ‘Συλλογισμού Βασισμένο σε Περιπτώσεις’ (Case-Based Reasoning expert system).....	38
3.3.1	Κύκλος ζωής.....	40
3.3.2	Πλεονεκτήματα.....	41

3.3.3	<i>Αναπαράσταση κατά περίπτωση</i>	42
3.3.4	<i>Ανάκτηση κατά περίπτωση</i>	42
3.3.5	<i>Προσαρμογή περίπτωσης</i>	44
3.3.6	<i>Εκμάθηση της μηχανής</i>	44
4	Εκτιμητής Φόρτου Εργασίας για εφαρμογές σε περιβάλλον πλέγματος	46
4.1	Αρχιτεκτονική των υπολογιστικών πλεγμάτων	46
4.2	Μοντέλο φόρτου εργασίας.....	48
4.2.1	<i>Παράμετροι πόρων πλατφόρμας</i>	49
4.2.2	<i>Παράμετροι φόρτου εργασίας</i>	50
4.2.3	<i>Εκτίμηση του συνολικού χρόνου εκτέλεσης</i>	50
4.3	Εκτιμητής Φόρτου Εργασίας	51
4.3.1	<i>Εύρεση του δείγματος διεκπεραίωσης</i>	51
4.3.2	<i>Εμπειρία παρελθόντος</i>	53
4.3.3	<i>Έμπειρο σύστημα ‘Συλλογισμού Βασισμένο σε Περιπτώσεις (Case-Based Reasoning)</i>	53
5	Εκτιμητής Φόρτου Εργασίας για την δημιουργία τρισδιάστατων γραφικών (3D Rendering)	55
5.1	Δημιουργία τρισδιάστατων γραφικών (3D Rendering)	56
5.1.1	<i>Χρήση</i>	56
5.1.2	<i>Χαρακτηριστικά</i>	56
5.1.3	<i>Τεχνικές</i>	57
5.1.4	<i>Η εξίσωση του Rendering</i>	58
5.1.5	<i>Η αμφίδρομη Reflectance Distribution Function</i>	58
5.1.6	<i>Οπτική αντίληψη</i>	59
5.1.7	<i>Renderman Interface</i>	59
5.1.8	<i>Αρχεία RIB</i>	59
5.1.9	<i>AIR</i>	61
5.2	Εκτίμηση φόρτου εργασίας στην δημιουργία τρισδιάστατων γραφικών.....	61
5.3	Εκτιμητής Φόρτου Εργασίας για τον AIR	62
5.3.1	<i>Δειγματοληψία</i>	64
5.3.2	<i>Μείωση της ανάλυσης της εικόνας</i>	68

5.3.3	Σχεδιασμός του ΕΦΕ.....	68
6	Πειράματα και μετρήσεις.....	74
6.1	Παράμετροι αξιολόγησης	74
6.2	Σύστημα αξιολόγησης.....	75
6.3	Οργάνωση πειραμάτων	75
6.4	Αποτελέσματα.....	76
6.4.1	Χρονικές μετρήσεις.....	76
6.4.2	Μετρήσεις μνήμης	86
6.4.3	Μετρήσεις αποθηκευτικού χώρου.....	92
6.4.4	Μετρήσεις στην ανάλυση της εικόνας.....	93
6.5	Σύνοψη συμπερασμάτων αξιολόγησης.....	96
7	Τεχνικές λεπτομέρειες	99
7.1	Λεπτομέρειες υλοποίησης.....	99
7.1.1	Εκτέλεση του Air στο παρασκήνιο	99
7.1.2	Δημιουργία στρωμάτων	100
7.1.3	Δημιουργία υποστρωμάτων	102
7.2	Πλατφόρμες και προγραμματιστικά εργαλεία	103
7.2.1	Πλατφόρμα ανάπτυξης	103
7.2.2	Γλώσσες προγραμματισμού	103
7.2.3	Προγραμματιστικά εργαλεία.....	104
8	Επίλογος	106
8.1	Σύνοψη και συμπεράσματα.....	106
8.2	Μελλοντικές επεκτάσεις	107
9	Βιβλιογραφία και αναφορές.....	108

Πίνακας περιεχομένων εικόνων

Εικόνα 1 Αλληλεπίδραση υπηρεσιών σε υπηρεσιοστραφές περιβάλλον.....	19
Εικόνα 2 Εκτιμητής Φόρτου Εργασίας.....	26
Εικόνα 3 Τυπική μορφή έμπειρου συστήματος	37
Εικόνα 4 Ένα έμπειρο σύστημα Case-Based Reasoning	40
Εικόνα 5 Κύκλος ζωής του CBR	40
Εικόνα 6 Ανάκτηση περίπτωσης του CBR.....	43
Εικόνα 7 Διαδικασία εκμάθησης της μηχανής του CBR	45
Εικόνα 8 Στιγμιότυπο αρχιτεκτονικής υπολογιστικού πλέγματος.....	47
Εικόνα 9 Αίτηση υπηρεσίας πλέγματος.....	48
Εικόνα 10 Εκτιμητής Φόρτου Εργασίας.....	51
Εικόνα 11 Αποτέλεσμα δημιουργίας τρισδιάστατης εικόνας από renderer	61
Εικόνα 12 Στιγμιότυπο από το Gui του Air	63
Εικόνα 13 Στιγμιότυπο του Air από το command prompt	63
Εικόνα 14 Αποτελέσματα από την εκτέλεση του Air	64
Εικόνα 15 Στρωματοποίηση εργασίας.....	67
Εικόνα 16 Στρωματοποίηση και υποστρωματοποίηση εργασίας	68
Εικόνα 17 Βασική λειτουργία του CBR	70

Πίνακας περιεχομένων πινάκων

Πίνακας 1 Παράμετροι Πόρων Πλατφόρμας	49
Πίνακας 2 Παράμετροι φόρτου εργασίας	50
Πίνακας 3 Πειραματικό υλικό από συλλογές με αρχεία rib	76
Πίνακας 4 Αντιστοιχία ανάλυσης εικόνας και μεγέθους αρχείου σε κωδικοποίηση tif.....	92
Πίνακας 5 Βάση δεδομένων- πίνακας stats του σχήματος dbstats.....	102

Πίνακας περιεχομένων μετρήσεων

Μέτρηση 1 Τυχαία δειγματοληψία στις συλλογές 1+9 (Χρόνος).....	77
Μέτρηση 2 Τυχαία δειγματοληψία στις συλλογές 2+8 (Χρόνος).....	77
Μέτρηση 3 Τυχαία δειγματοληψία στις συλλογές 8+9 (Χρόνος).....	77
Μέτρηση 4 Τυχαία δειγματοληψία στις συλλογές 1+2+9 (Χρόνος)	77
Μέτρηση 5 Στρωματοποιημένη δειγματοληψία στις συλλογές 1+9 (Χρόνος).....	79
Μέτρηση 6 Στρωματοποιημένη δειγματοληψία στις συλλογές 2+8 (Χρόνος).....	80
Μέτρηση 7 Στρωματοποιημένη δειγματοληψία στις συλλογές 8+9 (Χρόνος).....	80
Μέτρηση 8 Στρωματοποιημένη δειγματοληψία στις συλλογές 1+2+9 (Χρόνος).....	80
Μέτρηση 9 Δειγματοληψία ποσόστωσης στις συλλογές 1+9 (Χρόνος).....	82
Μέτρηση 10 Δειγματοληψία ποσόστωσης στις συλλογές 2+8 (Χρόνος).....	82
Μέτρηση 11 Δειγματοληψία ποσόστωσης στις συλλογές 8+9 (Χρόνος).....	82
Μέτρηση 12 Δειγματοληψία ποσόστωσης στις συλλογές 1+2+9 (Χρόνος)	83
Μέτρηση 13 Σύγκριση εκτιμώμενου και πραγματικού μεγέθους αρχείο σε κωδικοποίηση tif.....	93

Πίνακας περιεχομένων γραφικών παραστάσεων

Γραφική παράσταση 1 Τυχαία δειγματοληψία στις συλλογές 1+9 (Χρόνος).....	78
Γραφική παράσταση 2 Τυχαία δειγματοληψία στις συλλογές 2+8 (Χρόνος).....	78
Γραφική παράσταση 3 Τυχαία δειγματοληψία στις συλλογές 8+9 (Χρόνος).....	79
Γραφική παράσταση 4 Τυχαία δειγματοληψία στις συλλογές 1+2+9 (Χρόνος).....	79
Γραφική παράσταση 5 Στρωματοποιημένη δειγματοληψία στις συλλογές 1+9 (Χρόνος).....	81
Γραφική παράσταση 6 Στρωματοποιημένη δειγματοληψία στις συλλογές 2+8 (Χρόνος).....	81
Γραφική παράσταση 7 Στρωματοποιημένη δειγματοληψία στις συλλογές 8+9 (Χρόνος).....	81
Γραφική παράσταση 8 Στρωματοποιημένη δειγματοληψία στις συλλογές 1+2+9 (Χρόνος).....	82
Γραφική παράσταση 9 Δειγματοληψία ποσόστωσης στις συλλογές 1+9 (Χρόνος).....	83
Γραφική παράσταση 10 Δειγματοληψία ποσόστωσης στις συλλογές 2+8 (Χρόνος).....	83
Γραφική παράσταση 11 Δειγματοληψία ποσόστωσης στις συλλογές 8+9 (Χρόνος).....	84
Γραφική παράσταση 12 Δειγματοληψία ποσόστωσης στις συλλογές 1+2+9 (Χρόνος).....	84
Γραφική παράσταση 13 Ποσόστωσης εναντίον στρωματοποιημένης στις συλλογές 1+9.....	85
Γραφική παράσταση 14 Ποσόστωσης εναντίον στρωματοποιημένης στις συλλογές 2+8.....	85
Γραφική παράσταση 15 Ποσόστωσης εναντίον στρωματοποιημένης στις συλλογές 8+9.....	85
Γραφική παράσταση 16 Ποσόστωσης εναντίον στρωματοποιημένης στις συλλογές 1+2+9.....	86
Γραφική παράσταση 17 Τυχαία δειγματοληψία 1% στις συλλογές 1+2 (Μνήμη).....	87
Γραφική παράσταση 18 Τυχαία δειγματοληψία 5% στις συλλογές 1+2 (Μνήμη).....	87
Γραφική παράσταση 19 Τυχαία δειγματοληψία 10% στις συλλογές 1+2 (Μνήμη).....	87
Γραφική παράσταση 20 Τυχαία δειγματοληψία 1% στις συλλογές 3+4+9(Μνήμη).....	88
Γραφική παράσταση 21 Τυχαία δειγματοληψία 5% στις συλλογές 3+4+9(Μνήμη).....	88
Γραφική παράσταση 22 Τυχαία δειγματοληψία 10% στις συλλογές 3+4+9(Μνήμη).....	88
Γραφική παράσταση 23 Τυχαία δειγματοληψία 1% στις συλλογές 1+2+4+7(Μνήμη).....	89
Γραφική παράσταση 24 Τυχαία δειγματοληψία 5% στις συλλογές 1+2+4+7(Μνήμη).....	89
Γραφική παράσταση 25 Τυχαία δειγματοληψία 10% στις συλλογές 1+2+4+7(Μνήμη).....	89
Γραφική παράσταση 26 Δειγματοληψία ποσόστωσης 1% στις συλλογές 1+2+7(Μνήμη).....	90
Γραφική παράσταση 27 Δειγματοληψία ποσόστωσης 5% στις συλλογές 1+2+7(Μνήμη).....	90
Γραφική παράσταση 28 Δειγματοληψία ποσόστωσης 10% στις συλλογές 1+2+7(Μνήμη).....	91
Γραφική παράσταση 29 Δειγματοληψία ποσόστωσης 1% στις συλλογές 3+4+7(Μνήμη).....	91
Γραφική παράσταση 30 Δειγματοληψία ποσόστωσης 5% στις συλλογές 3+4+7(Μνήμη).....	91
Γραφική παράσταση 31 Δειγματοληψία ποσόστωσης 10% στις συλλογές 3+4+7(Μνήμη).....	92
Γραφική παράσταση 32 Σύγκριση εκτιμώμενου –πραγματικού χρόνου σε μικρές αναλύσεις.....	94
Γραφική παράσταση 33 Σύγκριση εκτιμώμενου –πραγματικού χρόνου σε μεγάλες αναλύσεις.....	95
Γραφική παράσταση 34 Σύγκριση εκτιμώμενου –πραγματικού χρόνου σε μικρές αναλύσεις.....	95
Γραφική παράσταση 35 Σύγκριση εκτιμώμενου –πραγματικού χρόνου σε μεγάλες αναλύσεις.....	95
Γραφική παράσταση 36 Σύγκριση εκτιμώμενου –πραγματικού χρόνου σε μικρές αναλύσεις.....	96
Γραφική παράσταση 37 Σύγκριση εκτιμώμενου –πραγματικού χρόνου σε μεγάλες αναλύσεις.....	96

1

Εισαγωγή

Η χρήση των κατανεμημένων συστημάτων από τις επιχειρήσεις και τις ακαδημαϊκές οργανώσεις έχει αυξηθεί εκθετικά τα τελευταία χρόνια ενδυναμωμένη από παράγοντες όπως η πρόσβαση στο διαδίκτυο και το World Wide Web, η ωριμότητα και η πανταχού παρουσία του πρωτοκόλλου HTTP και η βελτίωση στην ασφαλή τεχνολογία των επικοινωνιών. Στις πρώτες μέρες, οι κατανεμημένες εφαρμογές επικοινωνούσαν με τη χρησιμοποίηση ιδιόκτητων πρωτοκόλλων και οι διαχειριστές συστημάτων χρησιμοποιούσαν ειδικές μεθόδους για να διαχειριστούν τα συστήματα υπολογιστών που μπορεί να βρίσκονταν σε διαφορετικές πόλεις, χώρες ή ακόμη και σε διαφορετικές ηπείρους.

Πολυάριθμα πρότυπα έχουν αναπτυχθεί κατά τη διάρκεια των ετών για να διευκολύνουν τις δαπάνες της επέκτασης και της συντήρησης των κατανεμημένων συστημάτων με διαφορετικά ποσοστά επιτυχίας. Σήμερα, οι βασικές τεχνολογίες στα κατανεμημένα συστήματα είναι προσανατολισμένες προς την υπηρεσιοστραφή αρχιτεκτονική (Service Oriented Architecture), τις υπηρεσίες ιστού (Web Services) και τα υπολογιστικά πλέγματα (Grids), τα οποία αποσκοπούν στην τυποποίηση και την υιοθέτηση τους από τις οργανώσεις και τις εταιρείες [13].

Είναι προφανές ότι όταν οι υπηρεσίες προσφέρονται σε επαγγελματικό επίπεδο πρέπει να ανταποκρίνονται σε υψηλό επίπεδο ποιότητας της υπηρεσίας (QoS). Γι' αυτό και οποιαδήποτε τεχνολογία μπορεί να βελτιώσει την ποιότητα μιας υπηρεσίας είναι επιθυμητή.

Τα υπολογιστικά πλέγματα συνήθως προσφέρουν υπηρεσίες σε περιπτώσεις που απαιτείται μεγάλη υπολογιστική ισχύς. Μία τεχνολογία που μπορεί να βελτιώσει αυτές τις υπηρεσίες είναι ένας Εκτιμητής του Φόρτου Εργασίας (ΕΦΕ). Είναι σημαντικό για έναν διαχειριστή συστήματος να ξέρει εκ των προτέρων τι απαιτήσεις έχει μία εργασία σε υπολογιστική ισχύ και κυρίως τον χρόνο διεκπεραίωσης αυτής. Επίσης είναι σημαντικό και για έναν πελάτη να ξέρει τι θα του κοστίσει σε χρήματα η χρησιμοποίηση μιας υπηρεσίας πλέγματος. Το rendering είναι μία διαδομένη υπηρεσία που μπορεί να προσφέρει ένα πλέγμα. Παρακάτω κάνουμε μια επισκόπηση της υπηρεσιοστραφής αρχιτεκτονικής, των υπηρεσιών ιστού και των υπολογιστικών πλεγμάτων, για να γίνει πιο κατανοητή η ανάγκη που υπάρχει για την εκτίμηση του φόρτου εργασίας σε ένα υπολογιστικό πλέγμα.

1.1 Επισκόπηση της Υπηρεσιοστραφούς Αρχιτεκτονικής, των Υπηρεσιών Ιστού και των Υπολογιστικών Πλεγμάτων

1.1.1 Υπηρεσιοστραφής Αρχιτεκτονική (Service Oriented Architecture)

Για την καλύτερη κατανόηση του όρου πρέπει να αναλύσουμε ξεχωριστά τις λέξεις που τον αποτελούν.

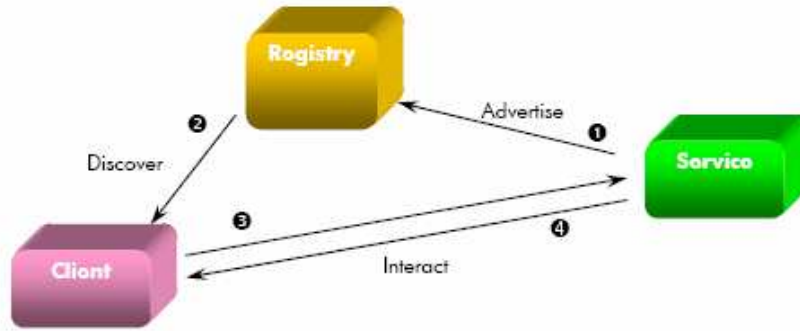
- Η αρχιτεκτονική είναι μια επίσημη περιγραφή ενός συστήματος η οποία καθορίζει το σκοπό, τις λειτουργίες, τις εξωτερικά ορατές ιδιότητες και τις διεπαφές του. Περιλαμβάνει επίσης την περιγραφή των εσωτερικών τμημάτων του συστήματος και των σχέσεών τους, μαζί με τις αρχές που κυβερνούν το σχέδιο, τη λειτουργία και την εξέλιξή του.
- Μια υπηρεσία είναι ένα τμήμα λογισμικού που μπορεί να προσεγγιστεί μέσω ενός δικτύου για να παράσχει τη λειτουργία στον αιτούντα υπηρεσιών.
- Ο όρος υπηρεσιοστραφής αρχιτεκτονική αναφέρεται σε ένα ύφος της οικοδόμησης των αξιόπιστων κατανεμημένων συστημάτων που παραδίδουν τη λειτουργία ως υπηρεσία, με έμφαση στη χαλαρή σύζευξη των αλληλεπιδρώσων υπηρεσιών.

Τεχνικά, κατ' επέκταση, ο όρος SOA αναφέρεται στο σχέδιο ενός συστήματος και όχι στην εφαρμογή του. Θεωρούμε τα SOA ως ένα αρχιτεκτονικό ύφος που εστιάζει στην εφαρμογή των συστατικών ως υπηρεσίες που μπορούν να ανακαλυφθούν και να χρησιμοποιηθούν από τους πελάτες. Οι υπηρεσίες έχουν γενικά τα ακόλουθα χαρακτηριστικά:

- Οι υπηρεσίες μπορούν να είναι ξεχωριστά χρήσιμες ή μπορούν να ενσωματωθούν και να συνθέσουν υψηλότερου επιπέδου υπηρεσίες. Μεταξύ άλλων ωφελειών, προωθείται η επαναχρησιμοποίηση της υπάρχουσας λειτουργίας.
- Οι υπηρεσίες επικοινωνούν με τους πελάτες τους με την ανταλλαγή μηνυμάτων. Καθορίζονται από τα μηνύματα που μπορούν να δεχτούν και τις απαντήσεις που μπορούν να δώσουν.
- Οι υπηρεσίες μπορούν να συμμετέχουν στη ροή μιας δουλειάς όπου η σειρά με την οποία τα μηνύματα στέλνονται και παραλαμβάνονται έχει επιπτώσεις στην έκβαση των διαδικασιών που εκτελούνται από μια υπηρεσία. Αυτή η έννοια ορίζεται ως “service choreography”.
- Οι υπηρεσίες μπορούν να είναι απολύτως ανεξάρτητες ή μπορούν να εξαρτηθούν από τη διαθεσιμότητα άλλων υπηρεσιών ή από την ύπαρξη ενός πόρου όπως μια βάση δεδομένων. Στην απλούστερη περίπτωση μια υπηρεσία μπορεί να εκτελέσει έναν υπολογισμό όπως ο υπολογισμός της ρίζας ενός αριθμού χωρίς να πρέπει να αναφερθεί σε οποιοδήποτε εξωτερικό πόρο ή μπορεί να έχει φορτώσει εκ των προτέρων όλα τα στοιχεία που χρειάζεται για τη διάρκεια ζωής της. Αντιθέτως, μια υπηρεσία που εκτελεί τη μετατροπή νομίσματος θα χρειαζόταν σε πραγματικό χρόνο πρόσβαση στις πληροφορίες τιμής συναλλάγματος προκειμένου να παραχθούν οι σωστές τιμές.
- Οι υπηρεσίες διαφημίζουν τις λεπτομέρειες τους όπως οι ικανότητες, οι διεπαφές, οι πολιτικές και τα υποστηριγμένα πρωτόκολλα επικοινωνιών τους. Λεπτομέρειες της εκτέλεσης όπως η γλώσσα προγραμματισμού και η πλατφόρμα λειτουργίας τους είναι αδιάφορες για τους πελάτες και δεν αποκαλύπτονται.

Η Εικόνα 1 επεξηγεί έναν απλό κύκλο αλληλεπίδρασης υπηρεσιών ο οποίος αρχίζει με μια υπηρεσία που διαφημίζεται μέσω μιας γνωστής υπηρεσίας καταγραφής.

1. Ένας πιθανός πελάτης, που μπορεί να είναι μια άλλη υπηρεσία, ρωτά το μητρώο για μια υπηρεσία που τον ενδιαφέρει.
2. το μητρώο επιστρέφει έναν (ενδεχομένως κενό) κατάλογο κατάλληλων υπηρεσιών και ο πελάτης επιλέγει μία υπηρεσία στέλνοντας ένα μήνυμα αίτησης χρησιμοποιώντας οποιοδήποτε αμοιβαία αναγνωρισμένο πρωτόκολλο.
3. Σε αυτό το παράδειγμα, η υπηρεσία αποκρίνεται
4. είτε με το αποτέλεσμα της ζητούμενης λειτουργίας είτε με ένα μήνυμα σφάλματος.



Εικόνα 1 Αλληλεπίδραση υπηρεσιών σε υπηρεσιοστραφές περιβάλλον.

1.1.2 Υπηρεσίες δικτύου (*Web services*)

Οι περισσότεροι άνθρωποι είναι εξοικειωμένοι με την πρόσβαση του Ιστού μέσω ενός web browser ο οποίος παρέχει μια διεπαφή στις πληροφορίες και τις υπηρεσίες που ενδιαφέρουν τον χρήστη. Όταν ένας χρήστης ζητά μια ιστοσελίδα το αίτημα του αντιμετωπίζεται από έναν μακρινό κεντρικό υπολογιστή δικτύου ο οποίος επιστρέφει τις πληροφορίες στη γλώσσα σήμανσης υπερκειμένων (HTML) που επιτρέπει στον browser να παρουσιάσει την σελίδα με κείμενο, εικόνες, χρώματα και οτιδήποτε άλλο που μπορεί να κάνει την σελίδα πιο φιλική προς τον χρήστη.

Οι υπηρεσίες Ιστού είναι καταναμημένα τμήματα λογισμικού που παρέχουν τις πληροφορίες στις εφαρμογές παρά στους ανθρώπους, μέσω μιας διεπαφής. Οι πληροφορίες είναι δομημένες χρησιμοποιώντας την εκτεταμένη γλώσσα σήμανσης (XML) έτσι ώστε να μπορούν να αναλυθούν και να υποβληθούν σε εύκολη επεξεργασία. Παραδείγματα τέτοιων υπηρεσιών είναι η αγορά μέσω πιστωτικών καρτών, το ηλεκτρονικό ταχυδρομείο κτλ.

Οι υπηρεσίες Ιστού δημοσιεύουν τις λεπτομέρειες των λειτουργιών και των διεπαφών τους αλλά κρατούν τις λεπτομέρειες της εκτέλεσής τους ιδιωτικές και κατά συνέπεια ένας πελάτης ή μια υπηρεσία που υποστηρίζουν τα κοινά πρωτόκολλα επικοινωνίας μπορούν να αλληλεπιδράσουν ανεξάρτητα από τις πλατφόρμες στις οποίες τρέχουν ή τις γλώσσες προγραμματισμού στις οποίες γράφονται. Αυτό καθιστά τις υπηρεσίες Ιστού ιδιαίτερα εφαρμόσιμες σε ένα καταναμημένο ετερογενές περιβάλλον.

Οι βασικές προδιαγραφές που χρησιμοποιούνται από τις υπηρεσίες Ιστού είναι:

- XML (εκτεταμένη γλώσσα σήμανσης) - μια γλώσσα σήμανσης για τη μορφοποίηση και την ανταλλαγή των δεδομένων.
- SOAP (αρχικά απλό πρωτόκολλο πρόσβασης αντικειμένου) - ένα xML-βασισμένο πρωτόκολλο για τη διευκρίνιση των πληροφοριών και των μηνυμάτων.
- WSDL (γλώσσα περιγραφής υπηρεσιών Ιστού) - μια xML-βασισμένη γλώσσα που χρησιμοποιείται για να περιγράψει τις ιδιότητες, τις διεπαφές και άλλες ιδιότητες μιας

υπηρεσίας Ιστού. Ένα έγγραφο WSDL μπορεί να διαβαστεί από έναν πιθανό πελάτη για να ενημερωθεί για την υπηρεσία.

Αν και μια υπηρεσία Ιστού μπορεί να υποστηρίξει οποιοδήποτε πρωτόκολλο επικοινωνίας και μπορεί να προσφέρει στους πελάτες της το περιθώριο της επιλογής, το πιο κοινό είναι το SOAP άνω του HTTP ή του HTTPS.

1.1.3 Υπηρεσιοστραφής Αρχιτεκτονική και Υπηρεσίες Ιστού: Ύψος εναντίον της εφαρμογής

Αρχικά περιγράψαμε τα SOA χωρίς να αναφέρουμε τις υπηρεσίες Ιστού και αντίστροφα. Αυτό επειδή είναι αντίθετα: οι υπηρεσίες αρχιτεκτονικής είναι ένα αρχιτεκτονικό ύψος ενώ οι υπηρεσίες Ιστού είναι μια τεχνολογία εφαρμογής. Και τα δύο μπορούν να χρησιμοποιηθούν μαζί, όπως γίνεται συχνά, αλλά δεν είναι αμοιβαία εξαρτώμενα. Παραδείγματος χάριν, αν και θεωρείται μια ευρέως αποδεκτή λύση στα κατακεκομμένα υπολογιστικά συστήματα, το SOA μπορεί να εφαρμοστεί σε ένα ενιαίο σύστημα όπου οι υπηρεσίες είναι μεμονωμένες διαδικασίες με τις σαφείς διεπαφές τους που επικοινωνούν με τη χρησιμοποίηση των τοπικών καναλιών ή σε μια ανεξάρτητη συστάδα όπου επικοινωνούν με μια μεγάλη ταχύτητα διασύνδεσης.

Παρομοίως ενώ οι υπηρεσίες Ιστού είναι συνηθισμένο να αποτελούν τη βάση για ένα προσανατολισμένο προς τις υπηρεσίες περιβάλλον, δεν υπάρχει τίποτα στον καθορισμό τους που να απαιτεί να ενσωματώσουν τις αρχές SOA.

1.1.4 Υπολογιστικά Πλέγματα (Grids)

Το υπολογιστικό πλέγμα είναι μία συλλογή γεωγραφικά κατακεκομμένων ετερογενών υπολογιστικών πόρων. Το υπολογιστικό πλέγμα είναι γενικά ένας πρόσθετος τύπος παράλληλης επεξεργασίας, το οποίο στηρίζεται σε πλήρεις υπολογιστές (με ΚΜΕ, αποθηκευτικά μέσα, την παροχή ηλεκτρικού ρεύματος, τη διεπαφή δικτύων κ.λπ.) που συνδέονται με ένα δίκτυο (ιδιωτικό, δημόσιο ή το διαδίκτυο) με μια συμβατική διεπαφή δικτύων, όπως το Ethernet. Έτσι το υπολογιστικό πλέγμα δίνει την αίσθηση ότι είναι ένας υπέρ-υπολογιστής. Αυτό είναι σε αντίθεση με την παραδοσιακή έννοια ενός υπέρ-υπολογιστή ο οποίος συνδέει πολλούς επεξεργαστές με ένα τοπικό διάδρομο υψηλής ταχύτητας. Έτσι ο χρήστης υποβάλλει εργασίες για εκτέλεση χωρίς να τον ενδιαφέρει που θα εκτελεστούν [16].

Το υπολογιστικό πλέγμα είναι μια αναπτυσσόμενη υποδομή που χρησιμοποιείται όπου υπάρχουν απαιτήσεις για μεγάλη υπολογιστική ισχύ ή για επεξεργασία πολύ μεγάλου όγκου δεδομένων. Τα τελευταία χρόνια το πλέγμα έχει ραγδαία ανάπτυξη. Σε αυτό έχουν βοηθήσει

οι νέες τεχνικές που χρησιμοποιούνται στο διαδίκτυο, οι οπτικές ίνες και η βελτίωση των ασύρματων ζεύξεων που συμβάλλουν στην αύξηση της ταχύτητας των δικτύων υπολογιστών. Δεν υπάρχει αμφιβολία πώς η πιο άμεση εφαρμογή του Grid αφορά τον επαγγελματικό χώρο. Οι προσωπικοί υπολογιστές, που λειτουργούν πάντα χαμηλότερα από τις δυνατότητές τους, είναι οι κυριότεροι υπολογιστικοί πόροι που μπορούν να χρησιμοποιηθούν σε ένα εταιρικό δίκτυο για την δημιουργία ενός Grid [1][9].

Υπάρχει όμως και ο προσανατολισμός της δημιουργίας εταιριών που θα παρέχουν υπηρεσίες υψηλών υπολογιστικών απαιτήσεων, που θα βασίζονται σε τεχνολογίες Grid. Οι εταιρίες αυτές θα καλύπτουν τις ανάγκες οποιονδήποτε οργανισμών οι οποίοι θέλουν να εκτελέσουν της εφαρμογές τους και να πάρουν τα αποτελέσματά του, με τρόπο οικονομικό, γρήγορο, ασφαλή και αποδοτικό. Διαπιστώνοντας λοιπόν τις ανάγκες των εταιριών σε ισχύ αλλά και των υπάρχοντων τεχνολογικών υποδομών, η δυνατότητα παροχής υπολογιστικής ισχύος καθίσταται αναγκαία αλλά και παράλληλα εφικτή.

Είναι προφανές ότι για να λειτουργήσει ένα ουσιαδές Grid είτε σε επίπεδο εταιρίας, που θα το χρησιμοποιεί για τους δικούς της σκοπούς είτε σαν ανεξάρτητος πάροχος υπολογιστικών υπηρεσιών χρειάζεται να πληροί κάποιες προϋποθέσεις.

Αρχικά πρέπει να υπάρχει η απαραίτητη προτυποποίηση. Για να είναι εφικτό το Grid να φτάσει σε κάθε πελάτη και να χρησιμοποιηθεί ευρέως είναι υποχρεωτικό να ακολουθήσει τα πρότυπα ενός ανώτερου οργανισμού. Ο οργανισμός που έχει αναλάβει την εποπτεία και την συνεργασία όλων των ερευνητικών εγχειρημάτων σχετικά με το Grid είναι το GGF(Global Grid Forum). Το εκάστοτε πλέγμα και ειδικά αυτά που προορίζονται για επαγγελματικές εφαρμογές πρέπει να υιοθετούν τα υπάρχοντα πρότυπα και να συνεισφέρουν στην δημιουργία νέων. Αυτός είναι ο μόνος τρόπος να εξασφαλιστεί συμβατότητα ανάμεσα στους οργανισμούς και μεταξύ των εφαρμογών και υλοποιήσεων Grid. Παράλληλα με αυτόν τον τρόπο είναι εύκολο να υιοθετηθούν και νέες τεχνολογικές εξελίξεις που σίγουρα θα πραγματοποιηθούν τα επόμενα χρόνια.

Ακόμη η εξέλιξη του Grid έχει προσανατολιστεί στην ενσωμάτωση των νέων τεχνολογιών για μέγιστη χρησιμοποίηση όλων των δυνατών υπολογιστικών πόρων. Η διαδικασία αυτή πρέπει να δημιουργήσει ένα νέο περιβάλλον που θα έχει παράλληλα δυνατότητες διαχείρισης από τους υπευθύνους της εταιρίας αλλά και δυνατότητες χρέωσης των πελατών. Στις μέρες μας οι συναλλαγές μεταξύ των εταιριών γίνονται αποκλειστικά με ηλεκτρονικά μέσα κάτι που φυσικά πρέπει να ενσωματώνει και το περιβάλλον του επαγγελματικού Grid.

Πολύ σημαντικός είναι και ο τομέας της ασφάλειας. Σήμερα έχουμε διαπιστώσει ότι κάθε σύστημα που χρησιμοποιεί το διαδίκτυο είναι ευάλωτο σε επιθέσεις που γίνονται για κερδοσκοπικούς ή όχι σκοπούς. Σε ένα ευαίσθητο σύστημα πλέγματος όπου διακινούνται σημαντικά δεδομένα, όπως για παράδειγμα ανταγωνιστικές εταιρικές εφαρμογές, η ασφάλεια

είναι πολύ σημαντικός παράγοντας. Αρχικά είναι υψίστης σημασίας η ταυτοποίηση των χρηστών. Κάθε εταιρία που παρέχει υπολογιστικές υπηρεσίες πρέπει να έχει μηχανισμούς μέσα από τους οποίους να επιτρέπει μόνο σε εξουσιοδοτημένους – πιστοποιημένους χρήστες να έχουν πρόσβαση και ταυτόχρονα να αποκλείει την είσοδο σε όλους τους άλλους. Ακόμη πρέπει να εξασφαλίζει το απόρρητο ανάμεσα στους χρήστες που καταχωρούν δουλειές και τον απόλυτο διαχωρισμό των δεδομένων. Σε κάθε διαφορετική περίπτωση θα αμφισβητηθεί η τεχνολογία και οι υπηρεσίες του πλέγματος που παρέχονται και τελικά θα απορριφθούν από τον επαγγελματικό χώρο.

Τέλος έχει πολύ μεγάλη σημασία και ο τομέας της χρηστικότητας. Από την μέχρι τώρα εμπειρία σε τεχνολογίες Grid έχουμε διαπιστώσει πως οι εφαρμογές που υπάρχουν μέχρι σήμερα είναι αρκετά δύσχρηστες και συνήθως παρουσιάζουν πολλά προβλήματα διότι βρίσκονται ακόμη σε φάση εξέλιξης. Για να μπορεί να λειτουργήσει ένα εταιρικό πλέγμα θα πρέπει να είναι σταθερό, χωρίς προβλήματα που να προϋποθέτουν την απασχόληση ενός ειδικού σε θέματα Grid. Ακόμα θα πρέπει να περιλαμβάνει ένα εύχρηστο γραφικό περιβάλλον. Ο εκάστοτε πελάτης δεν θα πρέπει να έχει καμία υποχρέωση γνώσεων περί πλέγματος. Γνωρίζει μόνο τις ανάγκες και τις απαιτήσεις που έχει η δουλειά του και μέσα από αυτό το περιβάλλον πρέπει να μπορεί να καταχωρεί τις δουλειές του με ευκολία ανάλογη των αγορών μέσω Internet.

1.1.4.1 Κατηγορίες υπολογιστικών πλεγμάτων

- Υπολογιστικά Grids (Computational Grids): Συλλογή κατανεμημένων υπολογιστικών υποδομών οι οποίες λειτουργούν ως ενιαίος επεξεργαστής και πραγματοποιούν επεξεργασία δεδομένων με μεγάλες υπολογιστικές απαιτήσεις. Τα αποτελέσματα είναι ταχύτερα, αποτελεσματικότερα και με μικρότερο κόστος.
- Grids Δεδομένων (Data Grids): Οι χρήστες και οι εφαρμογές διαχειρίζονται πληροφορίες από βάσεις δεδομένων που βρίσκονται σε κατανεμημένες πλατφόρμες ευκολότερα και αποτελεσματικότερα. Επίσης το κόστος είναι μικρότερο γιατί δεν υπάρχει ανάγκη για μεταφορά, αντιγραφή και συγκέντρωση δεδομένων σε ένα κεντρικό σημείο, καθώς και μεγαλύτερη αξιοπιστία κατά την πρόσβαση στα δεδομένα.
- Grids Υπηρεσιών (Service Grids): Πραγματοποίηση επεξεργασίας πραγματικού Χρόνου

1.1.4.2 Βασικές αρχές

- Διαμοιρασμός των υπολογιστικών πόρων : ένας χρήστης που αποκτάει πρόσβαση στο πλέγμα μπορεί να χρησιμοποιεί απομακρυσμένους πόρους που θα του επιτρέψουν να εκτελέσει εργασίες που δεν έχει τη δυνατότητα να εκτελέσει σε ένα μεμονωμένο υπολογιστή ή σε μία συστοιχία υπολογιστών ή ακόμα και να αποκτήσει πρόσβαση σε απομακρυσμένο λογισμικό, υπολογιστικούς πόρους, δεδομένα, απομακρυσμένους αισθητήρες, τηλεσκόπια ή επιστημονικά όργανα που ανήκουν σε άλλα ερευνητικά κέντρα.
- Ασφαλής πρόσβαση : αυτό επιτυγχάνεται με την πολιτική πρόσβασης (access policy), την ταυτοποίηση (authentication) και την εξουσιοδότηση (authorization).
- Αποτελεσματική χρήση των πόρων : Ανάπτυξη αλγορίθμων για την βέλτιστη ανάθεση των εργασιών στους πόρους που διατίθενται.
- Εξάλειψη της γεωγραφικής απόστασης : Απαιτούνται δίκτυα υψηλών ταχυτήτων για την διασύνδεση των πόρων.
- Δημιουργία κοινών προτύπων (Open Standards).

1.1.4.3 Δυνατότητες

- Καλύτερη εκμετάλλευση πόρων
- Παράλληλη υπολογιστική επεξεργασία
- Συνεργασία μέσω εικονικών οργανισμών
- Ισορροπία πόρων
- Αυξημένη αξιοπιστία

1.1.4.4 Χρήστες των υπολογιστικών πλεγμάτων

- Χρήστες εφαρμογών (End Users): αυτοί που θέλουν να χρησιμοποιήσουν μια εφαρμογή του πλέγματος.
- Σχεδιαστές εφαρμογών (Application Developers): αυτοί που σχεδιάζουν και αναπτύσσουν τις εφαρμογές που θα τρέξουν σε περιβάλλον υπολογιστικού πλέγματος.
- Διαχειριστές συστημάτων Grid (Grid Administrators): υπεύθυνοι για την διαχείριση των επιμέρους Grid υποδομών και την εξασφάλιση της σωστής λειτουργίας τους. (διαχειριστές δικτύου, διαχειριστές cluster, διαχειριστές cluster ευρύτερης

γεωγραφικής περιοχής, διαχειριστές των προγραμμάτων που χρησιμοποιούνται από τους εικονικούς οργανισμούς)

- Σχεδιαστές συστημάτων Grid (Grid Developers): υπεύθυνοι για την ανάπτυξη, τον σχεδιασμό και την εφαρμογή των νέων υπηρεσιών.
- Σχεδιαστές εργαλείων (Tool Developers): υπεύθυνοι για την ανάπτυξη εργαλείων, μεταγλωττιστών και βιβλιοθηκών.

1.2 Εκτίμηση φόρτου εργασίας

1.2.1 Γενικά

Ενώ ο ακριβής ορισμός του φόρτου εργασίας είναι αόριστος, ένας συνήθως αποδεκτός ορισμός είναι η υποθετική σχέση μεταξύ μιας ομάδας ή μεμονωμένων ανθρώπινων χειριστών, μηχανών ή ζώων και της διεκπεραίωσης μιας εργασίας.

Η αξιολόγηση του φόρτου εργασίας ασκεί ζωτικής σημασίας επίδραση στο σχεδιασμό νέων συστημάτων. Από την αξιολόγηση του φόρτου εργασίας κατά τη διάρκεια του σχεδίου ενός νέου συστήματος ή την επανάληψη ενός υπάρχοντος συστήματος, τα προβλήματα όπως οι δυσχέρειες και η υπερφόρτωση μπορούν να αποφευχθούν. Δεδομένου ότι ο ανθρώπινος χειριστής είναι ένα κεντρικό μέρος ενός συστήματος ανθρώπου-μηχανής, η διόρθωση αυτών των προβλημάτων είναι απαραίτητη για τη λειτουργία των ασφαλών και αποδοτικών συστημάτων [17].

1.2.2 Φόρτος εργασίας υπολογιστικών συστημάτων

Το Ferrari (1978) [2] καθορίζει τον φόρτο εργασίας ως την εργασία που εκτελείται από ένα συγκρότημα ηλεκτρονικών υπολογιστών σε μια χρονική περίοδο. Όλες οι είσοδοι στο συγκρότημα ηλεκτρονικών υπολογιστών π.χ, όλες οι απαιτήσεις που απαιτούν χρόνο εκτέλεσης, είναι μέρος του καθορισμού του φόρτου εργασίας. Η εκτέλεση ενός προγράμματος που κάνει μία εκτίμηση του φόρτου εργασίας αξιοποιεί όλους τους πόρους του συγκροτήματος ηλεκτρονικών υπολογιστών. Τέτοιοι πόροι είναι ο επεξεργαστής, ο διάδρομος συστήματος, η μνήμη, οι δίσκοι κ.λπ. Η χρησιμοποίηση των πόρων εξαρτάται από τον σχεδιασμό του συγκροτήματος ηλεκτρονικών υπολογιστών και του προγράμματος της εκτίμησης. Διαφορετικό πρόγραμμα θα δώσει διαφορετικά αποτελέσματα. Απώτερος σκοπός της εκτίμησης είναι ο καθορισμός των πόρων που απαιτούνται από ένα συγκρότημα ηλεκτρονικών υπολογιστών για να επιτύχουν ένα επιθυμητό επίπεδο απόδοσης. Οι περισσότεροι φόρτοι εργασίας μπορούν να περιγραφούν ανεξάρτητα από το συγκρότημα των ηλεκτρονικών υπολογιστών. Συγκεκριμένα, οι φόρτοι εργασίας που χρησιμοποιούνται για τη

σύγκριση των διαφορετικών συγκροτημάτων ηλεκτρονικών υπολογιστών πρέπει να είναι καθορισμένοι ανεξάρτητα της εφαρμογής (Gustafson and Snell, 1995).

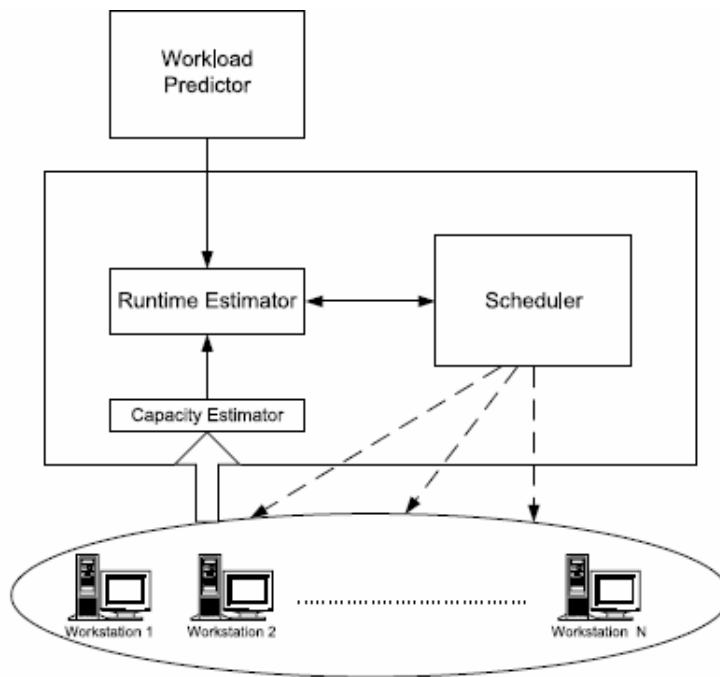
Η εκτίμηση του φόρτου εργασίας μπορεί να βοηθήσει σε δύο περιπτώσεις:

- Στην βελτίωση ενός υπάρχοντος συστήματος υπολογιστών ή κάποιας εφαρμογής.
- Στον καλύτερο σχεδιασμό ενός νέου συστήματος υπολογιστών ή εφαρμογής.

1.2.3 Εκτίμηση φόρτου εργασίας σε περιβάλλον υπολογιστικών πλεγμάτων

Η πρόβλεψη του συνολικού χρόνου εκτέλεσης που μια εργασία θα διαρκέσει όταν εκτελείται σε μια συγκεκριμένη πλατφόρμα είναι σημαντική επειδή οδηγεί σε μια καλύτερη χρησιμοποίηση των υπάρχοντων υπολογιστικών πόρων. Το γεγονός αυτό πολλαπλασιάζεται όταν η εργασία θα διεκπεραιωθεί σε ένα υπολογιστικό πλέγμα που μπορεί να αποτελείται από δεκάδες υπολογιστές. Σε ένα πλέγμα υπάρχουν αλγόριθμοι χρονοδρομολόγησης των εργασιών. Ένας χρονοδρομολογητής πλέγματος πρέπει να λάβει τις αποφάσεις επιλογής των πόρων σε ένα περιβάλλον όπου δεν έχει κανέναν έλεγχο των τοπικών πόρων αφού οι πόροι είναι διανεμημένοι [12]. Η αποτελεσματικότητα του πλέγματος είναι δυνατή μόνο εάν οι πόροι αξιοποιούνται καλά. Η χρονοδρομολόγηση πλέγματος ορίζεται ως η διαδικασία κατανομής των πόρων πέρα από τις πολλαπλές διοικητικές περιοχές. Αυτή η διαδικασία μπορεί να περιλάβει την έρευνα των πολλαπλών περιοχών για να χρησιμοποιήσει μια ενιαία μηχανή ή το σχεδιασμό μιας ενιαίας εργασίας για να χρησιμοποιήσει τους πόρους του πλέγματος. Μια από τις αρχικές διαφορές μεταξύ ενός χρονοδρομολογητή πλέγματος και ενός τοπικού χρονοδρομολογητή των πόρων είναι ότι ο χρονοδρομολογητής πλέγματος δεν είναι κύριος των τοπικών πόρων και επομένως δεν έχει τον έλεγχο τους. Ο χρονοδρομολογητής πλέγματος πρέπει να λάβει τις καλύτερες αποφάσεις και να υποβάλει έπειτα την εργασία στους πόρους που θα επιλεγθούν.

Αρα ένας Εκτιμητής του Φόρτου Εργασίας (ΕΦΕ) μπορεί να βοηθήσει στην καλύτερη απόδοση του πλέγματος. Ο ΕΦΕ μπορεί να διασπαστεί σε δύο μέρη. Στο πρώτο μέρος ο ΕΦΕ θα προβλέπει τον χρόνο διεκπεραίωσης μιας εργασίας χρησιμοποιώντας ιδιότητες και χαρακτηριστικά της εφαρμογής (workload prediction). Είναι φανερό ότι κάθε εφαρμογή θα έχει διαφορετικές ιδιαιτερότητες και απαιτήσεις. Οι παράμετροι φόρτου εργασίας καθορίζονται από κοινού με τις παραμέτρους των πόρων. Προκειμένου να υπολογιστεί ο φόρτος εργασίας ενός στόχου πρέπει να εξαγάγουμε τα χαρακτηριστικά γνωρίσματα που περιγράφουν τη συγκεκριμένη εφαρμογή. Στο δεύτερο μέρος η πρόβλεψη γίνεται χρησιμοποιώντας ιδιότητες και χαρακτηριστικά των πόρων που προσφέρει το υπολογιστικό πλέγμα (capacity estimator) [7] (Εικόνα 2).



Εικόνα 2 Εκτιμητής Φόρτου Εργασίας

Ενδιαφέρουσες προσεγγίσεις για την πρόβλεψη φόρτου εργασίας έχουν προταθεί. Μια κύρια διάκριση μπορεί να γίνει σύμφωνα με τις στατιστικές και τις αναλυτικές μεθόδους. Η στατιστική μέθοδος είναι συγκεκριμένη για την εφαρμογή που τρέχει σε διάφορες μεμονωμένες πλατφόρμες. Η εφαρμογή αναλύεται και καθορίζονται οι παράμετροι της εφαρμογής που έχουν επιπτώσεις στο χρόνο εκτέλεσης. Κατόπιν ένας αριθμός από διάφορα σενάρια, που χρησιμοποιούν διαφορετικές τιμές παραμέτρων, εκτελούνται από την εφαρμογή στις διάφορες πλατφόρμες και τα αποτελέσματα είναι ταξινομημένα σε ομάδες που εξαρτώνται από τις παραμέτρους εισαγωγής, την τεχνολογία και τη διαμόρφωση των πόρων. Το σύστημα ταξινομεί την υποβληθείσα εργασία σύμφωνα με τις τιμές των παραμέτρων εισαγωγής. Σύμφωνα με αυτήν την ταξινόμηση και την γνώση των πόρων του πλέγματος, το σύστημα επιστρέφει μια εκτίμηση του χρόνου εκτέλεσης της εργασίας.

Η αναλυτική μέθοδος για την εκτίμηση του χρόνου εκτέλεσης μπορεί να εφαρμοστεί με πολύ καλύτερα αποτελέσματα από την στατιστική μέθοδο, ειδικά στα ετερογενή συγκροτήματα ηλεκτρονικών υπολογιστών. Σε αυτήν την περίπτωση απαιτείται η ανάλυση του πηγαίου κώδικα της εφαρμογής. Βασικές διαδικασίες που έχουν επιπτώσεις στον χρόνο εκτέλεσης της εφαρμογής επιλέγονται και ο χρόνος εκτέλεσης μετριέται για κάθε διαφορετική πλατφόρμα. Δεδομένου ότι ο χρόνος εκτέλεσης των μεμονωμένων διαδικασιών είναι γνωστός σε μια δεδομένη πλατφόρμα, η εκτίμηση του συνολικού χρόνου εκτέλεσης μπορεί να υπολογιστεί άμεσα για το συγκεκριμένο σενάριο εφαρμογής.

Είδαμε ότι τα υπολογιστικά πλέγματα υποστηρίζουν τη διανομή, τη διασύνδεση και τη χρήση των διαφορετικών πόρων στα δυναμικά συστήματα υπολογισμού που μπορούν να

ενσωματωθούν για να παραδώσουν την υπολογιστική δύναμη στις εφαρμογές που την χρειάζονται. Εντούτοις η εμπορευματοποίηση και η εκβιομηχάνιση των πλεγμάτων υπονοούν την επέκταση των ενοτήτων και των υπηρεσιών πέρα από τις υπάρχουσες στα τρέχοντα πλέγματα, τα οποία θα ικανοποιούσαν τις απαιτήσεις της ποιότητας της υπηρεσίας (Quality of Service) που επιβάλλονται από τους χρήστες που είναι πρόθυμοι να χρησιμοποιήσουν την υποδομή πλέγματος για την πραγματοποίηση των εμπορικών αναγκών τους. Το QoS είναι ένα μέτρο σύγκρισης για το πόσο κοντά είναι η υπηρεσία που παρέχεται από την υποδομή δικτύων και συστημάτων στις προσδοκίες και τις απαιτήσεις του χρήστη. Σε ένα υπολογιστικό πλέγμα οι χρήστες υποβάλλουν τις εργασίες τους με ένα σύνολο απαιτήσεων όπως η χρονική προθεσμία για την εκτέλεση της εργασίας ή το κόστος που θα πληρώσουν. Η εφικτή λύση είναι η συμφωνία QoS όπου ο χρήστης θα υποβάλει την εργασία έχοντας μερικές συγκεκριμένες απαιτήσεις. Κατόπιν το σύστημα απαντά εάν μπορεί να ικανοποιήσει τις απαιτήσεις χρηστών ή όχι λαμβάνοντας υπόψη τους διαθέσιμους πόρους, την εκτέλεση άλλων εργασιών και τις αναμενόμενες μελλοντικές υποβαλλόμενες εργασίες. Προκειμένου να ληφθεί η συμφωνία QoS είναι απαραίτητο να είναι γνωστός εκ των προτέρων ο χρόνος εφαρμογής μιας υποβληθείσας εργασίας σε κάθε πόρο.

1.2.4 Ποιότητα Υπηρεσίας (Quality of Service)

Η ποιότητα εξυπηρέτησης (QoS) [8] είναι ένας συνδυασμός διάφορων ιδιοτήτων μιας υπηρεσίας, όπως:

- Διαθεσιμότητα : το ποσοστό του χρόνου που μια υπηρεσία λειτουργεί
- Ασφάλεια : η ύπαρξη και ο τύπος μηχανισμών επικύρωσης η εμπιστευτικότητα και η ακεραιότητα των μηνυμάτων που ανταλλάσσονται.
- Ο χρόνος απόκρισης : ο χρόνος που μια υπηρεσία παίρνει για να αποκριθεί στους διάφορους τύπους αιτημάτων.
- Η ρυθμοαπόδοση: το ποσοστό στο οποίο μια υπηρεσία μπορεί να επεξεργαστεί τα αιτήματα. Τα μέτρα QoS μπορούν να περιλάβουν τη μέγιστη ρυθμοαπόδοση ή μια λειτουργία που περιγράφει πώς η ρυθμοαπόδοση ποικίλλει με την ένταση φορτίων.

Είναι φανερό πώς η πρόβλεψη του φόρτου εργασίας μπορεί να ικανοποιήσει τις παραπάνω ιδιότητες εκτός από την ασφάλεια. Ειδικότερα, για να εξασφαλιστεί η ποιότητα της υπηρεσίας από τον πάροχο στον χρήστη χρησιμοποιούνται τα SLA(Service Level Agreement) [5][10]. Ένα SLA είναι μία διμερής συμφωνία μεταξύ ενός φορέα παροχής υπηρεσιών και μιας υπηρεσίας. Τα SLAs διαμορφώνουν μια φόρμα για την αντιπροσώπευση των συμφωνηθέντων περιορισμών για τις εργασίες. Σε οποιαδήποτε περίπτωση, το SLA έχει ως

σκοπό να περιλάβει τα αποδεκτά χρονικά όρια παράδοσης της. Ένα SLA πρέπει να πληροί τις παρακάτω προϋποθέσεις.

- Σκοπός: περιγράφει τους λόγους πίσω από τη δημιουργία του SLA
- Συμβαλλόμενα μέρη: περιγράφει τα συμβαλλόμενα μέρη που συμμετέχουν στο SLA και τους αντίστοιχους ρόλους τους (προμηθευτής και καταναλωτής)
- Περίοδος ισχύος: καθορίζει τη χρονική περίοδο που το SLA θα καλύψει. Αυτό οριοθετείται από το χρόνο έναρξης και το χρόνο λήξης του όρου.
- Πεδίο: καθορίζει τις υπηρεσίες που καλύπτονται στη συμφωνία.
- Περιορισμοί: καθορίζει τα απαραίτητα μέτρα που λαμβάνονται για τα ζητούμενα επίπεδα εξυπηρέτησης που παρέχονται.
- Ποινικές ρήτρες: εξηγεί τι συμβαίνει σε περίπτωση που ο φορέας παροχής υπηρεσιών είναι ανίκανος να επιτύχει τους στόχους του SLA.
- Προαιρετικές υπηρεσίες: επιτρέπει οποιεσδήποτε υπηρεσίες που δεν απαιτούνται κανονικά από το χρήστη, να απαιτηθούν ως εξαίρεση.
- Αποκλεισμοί: διευκρινίζει τι δεν καλύπτεται στο SLA.
- Διοίκηση: περιγράφει τις διαδικασίες που δημιουργούνται στο SLA για να επιτύχουν και να μετρήσουν τους στόχους του και καθορίζει την οργανωτική ευθύνη για κάθε μια από εκείνες τις διαδικασίες.

Οι αλγόριθμοι χρονοδρομολόγησης θα πρέπει να βελτιστοποιήσουν με τα τυποποιημένα κριτήρια την χρησιμοποίηση των πόρων αλλά πρέπει επίσης να είναι σε θέση να αξιολογήσουν ένα προτεινόμενο SLA, όσον αφορά το τρέχον σύνολο αποδεκτών SLAs τους, και να είναι σε θέση να δείξουν εάν το SLA είναι ή όχι αποδεκτό, επιστρέφοντας ένα προτεινόμενο SLA με τις πρόσθετες πληροφορίες στην καταφατική περίπτωση ή με τις προτεινόμενες αλλαγμένες παραμέτρους στην αρνητική περίπτωση.

Έτσι σε μία εκ των προτέρων εκτίμηση του φόρτου εργασίας δεν μας ενδιαφέρει απαραίτητα η ακρίβεια των αποτελεσμάτων αλλά η τάξη μεγέθους του φόρτου εργασίας και σε ποιο SLA θα ανήκει. Για παράδειγμα έστω ότι έχουμε 3 SLA. Το πρώτο αφορά εργασίες που διεκπεραιώνονται σε λιγότερο από 10 ώρες. Το δεύτερο από 10 ως 30 ώρες και το τρίτο από 30 και πάνω. Σε κάθε SLA αλλάζουν οι τιμές χρέωσης καθώς και οι ρήτρες αποζημίωσης. Έτσι ο πάροχος δεν ενδιαφέρεται άμεσα για το αν μια εργασία θα κάνει 5 ή 7 ώρες αλλά το ότι ανήκει στην πρώτη κατηγορία των SLAs. Σαφώς η μεγαλύτερη ακρίβεια του εκτιμητή είναι επιθυμητή αλλά και μία εκτίμηση κατά προσέγγιση είναι αποδεκτή.

1.3 Αντικείμενο διπλωματικής

Σκοπός της διπλωματικής είναι η μελέτη και η ανάπτυξη μιας μεθοδολογίας για την εκτίμηση του φόρτου εργασίας σε περιβάλλον υπολογιστικού πλέγματος. Η ιδιαιτερότητα της εφαρμογής μας έγκειται στο γεγονός ότι είναι εφαρμόσιμη σε περιπτώσεις εργασιών που δεν υπάρχει η δυνατότητα ή είναι πολύ δύσκολη η εκ των προτέρων εκτίμηση του φόρτου πριν την εκτέλεση της εργασίας. Μία τέτοια εφαρμογή και στην οποία θα εστιάσει η διπλωματική μας είναι η περίπτωση του 3D Rendering. Η πρόβλεψη του φόρτου εργασίας βοηθάει αφενός στην καλύτερη αξιοποίηση των πόρων ενός πλέγματος και στην χρονοδρομολόγηση των εργασιών και αφετέρου σε πιο αξιόπιστες συμφωνίες μεταξύ του παρόχου της υπηρεσίας και του πελάτη. Ειδικότερα η πρόβλεψη του χρόνου διεκπεραίωσης μιας εργασίας είναι η πιο σημαντική παράμετρος του φόρτου εργασίας καθώς ο χρόνος παράδοσης αποτελεί έναν από τους βασικότερους όρους των SLAs. Τα SLAs είναι συμφωνητικά που εξασφαλίζουν την ποιότητα της υπηρεσίας από τον πάροχο στον χρήστη, κάτι απαραίτητο στα εμπορικά πλέγματα που πρέπει να ικανοποιούν τα επαγγελματικά μοντέλα.

Η μεθοδολογία μας βασίζεται στην εκτέλεση ενός μέρους της εργασίας με σκοπό να παραχθούν πληροφορίες που θα είναι ικανές να δώσουν μία εκτίμηση για όλη την εργασία. Για τον σκοπό αυτό χρησιμοποιήσαμε μεθόδους δειγματοληψίας και πραγματοποιήσαμε μετρήσεις για να αξιολογήσουμε τις μεθόδους. Ωστόσο για την βελτίωση του μηχανισμού μας ενσωματώσαμε ένα έμπειρο σύστημα ‘Συλλογισμού Βασισμένο σε Περιπτώσεις’ (Case-Based Reasoning Expert System). Το έμπειρο αυτό σύστημα μπορεί να βελτιώσει τις μεθόδους δειγματοληψίας αξιοποιώντας την εμπειρία του παρελθόντος από προηγούμενες επιτυχημένες εκτιμήσεις.

1.4 Οργάνωση κειμένου

Στο πρώτο κεφάλαιο γίνεται μια επισκόπηση των υπηρεσιοστραφών υπηρεσιών, των υπηρεσιών ιστού και των υπολογιστικών πλεγμάτων μιας και αυτός είναι ο χώρος εφαρμογής της μεθοδολογίας μας. Στο κεφάλαιο 2 αναφέρονται εργασίες σχετικές με την διπλωματική. Στο τρίτο κεφάλαιο γίνεται μια παρουσίαση θεωρίας που θα χρειαστεί στην μελέτη μας. Το μοντέλο του φόρτου εργασίας για υπολογιστικά πλέγματα και η μεθοδολογία για την εκτίμηση του φόρτου για εφαρμογές που εκτελούνται σε πλέγματα παρουσιάζονται στο τέταρτο κεφάλαιο. Στο κεφάλαιο 5 αναπτύσσουμε την μεθοδολογία μας για την περίπτωση της εφαρμογής του 3D-rendering. Οι μετρήσεις και τα πειράματα για την αξιολόγηση των μεθοδολογιών και των τεχνικών καταγράφονται στο έκτο κεφάλαιο. Στο κεφάλαιο 7 παρουσιάζονται κάποιες τεχνικές λεπτομέρειες που αφορούν το προγραμματιστικό μέρος

καθώς και τα προγραμματιστικά εργαλεία που χρησιμοποιήθηκαν. Τα συμπεράσματα και η σύνοψη φαίνονται στο όγδοο κεφάλαιο ενώ το ένατο περιέχει τις απαραίτητες αναφορές.

2

Σχετικές εργασίες

Η εκτίμηση του φόρτου εργασίας σε περιβάλλον υπολογιστικού πλέγματος είναι ένα θέμα που απασχολεί την επιστημονική κοινότητα καθώς τα πλέγματα είναι ένας αναπτυσσόμενος κλάδος που χρησιμοποιείται και για εμπορικούς σκοπούς. Ειδικότερα η περίπτωση του 3D Rendering είναι από τις πιο διαδεδομένες υπηρεσίες υπολογιστικών πλεγμάτων και έχουν πραγματοποιηθεί διάφορες μελέτες για την επίλυση αυτού του προβλήματος. Προτού προχωρήσουμε στην ανάπτυξη της δικιάς μας μεθοδολογίας θα περιγράψουμε σύντομα μεθόδους που ασχολήθηκαν με αυτό το θέμα.

Η κεντρική ιδέα των μεθόδων αυτών βασίζεται στην ασαφή κατάταξη (fuzzy classification) σε συνδυασμό με τα νευρωνικά δίκτυα (neural networks) και στην λεκτική ανάλυση του rib αρχείου. Ένα rib αρχείο περιγράφει έναν συνθετικό κόσμο (βλ 5.1.8). Με την λεκτική ανάλυση του αρχείου μπορούν να εξαχθούν οι περιγραφητές οι οποίοι παρέχουν μια γενικότερη δομή που περιγράφει την εικόνα που θα παραχθεί από τον υπολογιστή. Η ασαφής ταξινόμηση χρησιμοποιείται για την οργάνωση των περιγραφητών έτσι ώστε να επιτυγχάνεται μια αξιόπιστη αντιπροσώπευση που αυξάνει την ακρίβεια της πρόβλεψης. Το νευρικό δίκτυο εκτελεί την πρόβλεψη του φόρτου εργασίας με τη μοντελοποίηση της μη γραμμικής σχέσης εισόδου-εξόδου μεταξύ των περιγραφητών και της αντίστοιχης υπολογιστικής πολυπλοκότητας. Ο λόγος που χρησιμοποιούνται νευρωνικά δίκτυα οφείλεται στο γεγονός ότι έχουν τη δυνατότητα να υπολογίσουν οποιαδήποτε συνεχή μη γραμμική συνάρτηση με οποιοδήποτε βαθμό ακρίβειας.

Η μεθοδολογία που θα αναπτύξουμε θα προσεγγίσει το πρόβλημα της εκτίμησης με έναν τελείως διαφορετικό τρόπο αλλά έχει ενδιαφέρον να έχουμε γνώση των μεθόδων που έχουν ασχοληθεί με το ίδιο αντικείμενο.

3

Θεωρητικό υπόβαθρο

Στην παρούσα μελέτη θα ήταν καλό ο αναγνώστης να έχει βασικές γνώσεις στατιστικής και τεχνικών δειγματοληψίας καθώς και γνώση των βασικών εννοιών της τεχνητής νοημοσύνης και των έμπειρων συστημάτων.

3.1 Στατιστική και δειγματοληψία

Η στατιστική [6] είναι ένας κλάδος των εφαρμοσμένων μαθηματικών που ασχολείται με τη συλλογή, τη συνοπτική παρουσίαση και ανάλυση παρατηρήσεων που υπόκεινται σε τυχαίες μεταβολές με τελικό στόχο την εξαγωγή βάσιμων συμπερασμάτων και τη λήψη βέλτιστων αποφάσεων σε συνθήκες αβεβαιότητας. Οι παρατηρήσεις αφορούν ένα ή περισσότερα χαρακτηριστικά των μονάδων ενός συνόλου που ονομάζεται πληθυσμός. Το σύνολο των μονάδων του πληθυσμού από τις οποίες λαμβάνουμε τις παρατηρήσεις ονομάζεται δείγμα. Είναι προφανές ότι τα οποιαδήποτε συμπεράσματα στα οποία θα μας οδηγήσει η ανάλυση του δείγματος θα πρέπει να μπορούν να επεκταθούν σε ολόκληρο τον πληθυσμό. Για να επιτευχθεί αυτό θα πρέπει το δείγμα να είναι αντιπροσωπευτικό του πληθυσμού. Η αντιπροσωπευτικότητα του δείγματος εξαρτάται από τον τρόπο συλλογής του δείγματος δηλαδή από τις τεχνικές δειγματοληψίας που μπορούν να εφαρμοστούν.

3.1.1 Δειγματοληψία

Η δειγματοληψία [18] είναι μέρος της πρακτικής στατιστικής, που με την κατάλληλη επιλογή παρατηρήσεων, προορίζεται να αποφέρει κάποια γνώση για τον ενδιαφερόμενο πληθυσμό. Κάθε παρατήρηση μετρά μία ή περισσότερες ιδιότητες μίας οντότητας του πληθυσμού και κάνοντας χρήση της θεωρίας πιθανοτήτων και της στατιστικής εξάγουμε αποτελέσματα. Η διαδικασία της δειγματοληψίας αποτελείται από επτά φάσεις.

1. καθορισμός του πληθυσμού
2. καθορισμός ενός πλαισίου από γεγονότα που θα μετρηθούν
3. καθορισμός της δειγματοληπτικής μεθόδου
4. καθορισμός του μεγέθους του δείγματος
5. εφαρμογή του σχεδίου δειγματοληψίας
6. δειγματοληψία και συλλογή στοιχείων
7. επανεξέταση της διαδικασίας

Για να υπάρξουν σωστά αποτελέσματα είναι αναγκαίο να καθοριστεί σωστά ο πληθυσμός. Δεν υπάρχει κανένας ακριβής κανόνας που να ορίζει τον πληθυσμό και ο ερευνητής πρέπει να στηριχθεί στη λογική και την κρίση. Ο πληθυσμός καθορίζεται σύμφωνα με τους στόχους της μελέτης. Μερικές φορές ολόκληρος ο πληθυσμός είναι αρκετά μικρός και ο ερευνητής μπορεί να περιλάβει ολόκληρο τον πληθυσμό στη μελέτη. Αυτός ο τύπος έρευνας καλείται μελέτη απογραφής επειδή τα δεδομένα συγκεντρώνονται σε κάθε μέλος του πληθυσμού. Συνήθως, ο πληθυσμός είναι πάρα πολύ μεγάλος για τον ερευνητή για να προσπαθήσει να ερευνήσει όλα τα μέλη του. Ένα μικρό αλλά προσεκτικά επιλεγμένο δείγμα μπορεί να χρησιμοποιηθεί για να αντιπροσωπεύσει τον πληθυσμό. Το δείγμα απεικονίζει τα χαρακτηριστικά του πληθυσμού από τον οποίο προέρχεται. Επίσης μεγάλη σημασία έχει ο καθορισμός του πλαισίου των ιδιοτήτων που θα μετρηθούν.

Η δειγματοληψία μπορεί να κατηγοριοποιηθεί στην πιθανοτική και στην μη πιθανοτική. Στην πρώτη περίπτωση κάθε μέλος του πληθυσμού έχει μία μη μηδενική και εκ των προτέρων γνωστή πιθανότητα να επιλεγεί. Στις πιθανοτικές ανήκουν η τυχαία (random), η συστηματική (systematic) και η στρωματοποιημένη (stratified) δειγματοληψία. Στις μη πιθανοτικές κάθε μέλος επιλέγεται από τον πληθυσμό με κάποια μη τυχαία διαδικασία.

Οι μέθοδοι δειγματοληψίας [19] είναι οι εξής:

- Η τυχαία δειγματοληψία είναι η καθαρότερη μορφή πιθανοτικής δειγματοληψίας. Κάθε μέλος του πληθυσμού έχει μια ίση και γνωστή πιθανότητα επιλογής.
- Η συστηματική δειγματοληψία χρησιμοποιείται συχνά αντί της τυχαίας δειγματοληψίας. Καλείται επίσης νιοστή τεχνική επιλογής δείγματος. Αφότου έχει

υπολογιστεί το απαραίτητο μέγεθος δειγμάτων, κάθε νιοστό δείγμα επιλέγεται από τον πληθυσμό. Εφ' όσον δεν περιέχει ο πληθυσμός οποιαδήποτε κρυμμένη πληροφορία, αυτή η δειγματοληπτική μέθοδος είναι τόσο καλή όσο η τυχαία δειγματοληπτική μέθοδος. Το μόνο πλεονέκτημά της τυχαίας δειγματοληψίας είναι η απλότητα. Η συστηματική δειγματοληψία χρησιμοποιείται συχνά για να επιλέξει έναν διευκρινισμένο αριθμό αρχείων από υπολογιστή.

- Η στρωματοποιημένη δειγματοληψία είναι η πιο συνηθισμένη πιθανοτική μέθοδος δειγματοληψίας που χρησιμοποιείται και είναι ανώτερη από την τυχαία δειγματοληψία επειδή μειώνει το λάθος δειγματοληψίας. Ένα στρώμα είναι ένα υποσύνολο του πληθυσμού που έχει τουλάχιστον ένα κοινό χαρακτηριστικό. Ο ερευνητής προσδιορίζει αρχικά τα σχετικά στρώματα και την πραγματική αντιπροσώπευσή τους στον πληθυσμό. Η τυχαία δειγματοληψία χρησιμοποιείται έπειτα για να επιλέξει έναν ικανοποιητικό αριθμό θεμάτων από κάθε στρώμα.
- Η άνετη δειγματοληψία χρησιμοποιείται στη διερευνητική έρευνα όπου ο ερευνητής ενδιαφέρεται να πάρει μια ανέξοδη προσέγγιση της αλήθειας. Όπως το όνομα υπονοεί, το δείγμα επιλέγεται επειδή είναι εύκαιρο. Αυτή η μη πιθανοτική μέθοδος χρησιμοποιείται συχνά κατά τη διάρκεια των προκαταρκτικών ερευνητικών προσπαθειών ώστε να αποκτηθεί μια ακαθόριστη εκτίμηση των αποτελεσμάτων χωρίς την ανάληψη του κόστους ή του χρόνου που απαιτείται για την επιλογή ενός τυχαίου δείγματος.
- Η δειγματοληψία κρίσης είναι μια κοινή μη πιθανοτική μέθοδος. Ο ερευνητής επιλέγει το δείγμα βασισμένο στην κρίση. Αυτή είναι συνήθως και η επέκταση της άνετης δειγματοληψίας. Κατά τη χρησιμοποίηση αυτής της μεθόδου, ο ερευνητής πρέπει να είναι βέβαιος ότι το επιλεγμένο δείγμα είναι αληθινά αντιπροσωπευτικό ολόκληρου του πληθυσμού.
- Η δειγματοληψία ποσόστωσης είναι η μη πιθανοτική ισοδύναμη της στρωματοποιημένης δειγματοληψίας. Όπως στη στρωματοποιημένη δειγματοληψία, ο ερευνητής προσδιορίζει αρχικά τα στρώματα και τις αναλογίες τους όπως αντιπροσωπεύονται στον πληθυσμό. Κατόπιν η άνετη δειγματοληψία ή η κρίσης χρησιμοποιείται για να επιλέξει τον απαραίτητο αριθμό δειγμάτων από κάθε στρώμα. Αυτό διαφέρει από τη στρωματοποιημένη δειγματοληψία, όπου η τυχαία δειγματοληψία επιλέγει τα δείγματα από τα στρώματα.
- Η δειγματοληψία χιονοστιβάδας είναι μια πρόσθετη μη πιθανοτική μέθοδος χρησιμοποιούμενη όταν το επιθυμητό χαρακτηριστικό δειγμάτων είναι σπάνιο. Μπορεί να είναι εξαιρετικά δύσκολο ή το κόστος απαγορευτικό να βρεθούν τα δείγματα σε αυτές τις καταστάσεις. Η δειγματοληψία χιονοστιβάδας στηρίζεται στις

αρχικές παραπομπές για να παραγάγει τα πρόσθετα δεδομένα. Ενώ αυτή η τεχνική μπορεί εντυπωσιακά να χαμηλώσει τις δαπάνες αναζήτησης, έρχεται εις βάρος της εισαγωγής της προκατάληψης επειδή η ίδια η τεχνική μειώνει την πιθανότητα ότι το δείγμα θα είναι αντιπροσωπευτικό.

3.2 Αρχιτεκτονική των έμπειρων συστημάτων

Ένα έμπειρο σύστημα είναι ένα πρόγραμμα υπολογιστών που μιμείται τη διαδικασία σκέψης ενός ανθρώπινου εμπειρογνώμονα για να λύσει τα σύνθετα προβλήματα απόφασης σε μια συγκεκριμένη περιοχή [15]. Ένα έμπειρο σύστημα λειτουργεί ως ένα διαλογικό σύστημα που αποκρίνεται στις ερωτήσεις, υποβάλλει διευκρινίσεις, υποβάλλει συστάσεις και βοηθά γενικά τη διαδικασία λήψης απόφασης. Τα έμπειρα συστήματα παρέχουν στον εμπειρογνώμονα συμβουλές και καθοδήγηση σε μια ευρεία ποικιλία δραστηριοτήτων, από τη διάγνωση υπολογιστών ως στη λεπτή ιατρική χειρουργική επέμβαση. Διάφοροι ορισμοί των έμπειρων συστημάτων έχουν προταθεί. Ένας γενικός καθορισμός που είναι αντιπροσωπευτικός των προοριζόμενων λειτουργιών των έμπειρων συστημάτων είναι ο εξής:

An expert system is an interactive computer-based decision tool that uses both facts and heuristics to solve difficult decision problems based on knowledge acquired from an expert.

Οι περισσότερες εφαρμογές των έμπειρων συστημάτων ανήκουν σε μια από τις ακόλουθες κατηγορίες:

- Ερμηνεία και προσδιορισμός
- Πρόβλεψη
- Διάγνωση
- Σχεδίαση
- Εποπτεία
- Διόρθωση και δοκιμή
- Καθοδήγηση και κατάρτιση
- Έλεγχος

Τα έμπειρα συστήματα είναι κατάλληλα σε προβλήματα διαγνωστικής όταν δεν υπάρχει καθιερωμένη θεωρία, η ανθρώπινη εμπειρία είναι σπάνια ή τα δεδομένα περιέχουν αβεβαιότητα. Αντίθετα είναι ακατάλληλα σε καθαρά υπολογιστικά προβλήματα όταν υπάρχει μαθηματική λύση ή όταν τα δεδομένα είναι γνωστά επακριβώς.

3.2.1 Ιδιότητες ενός έμπειρου συστήματος

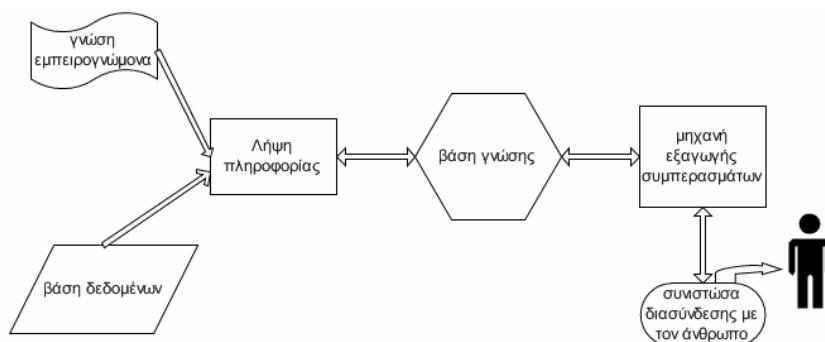
- Κάθε έμπειρο σύστημα είναι εφαρμόσιμο σε μία συγκεκριμένη περιοχή της ανθρώπινης εμπειρίας.
- Μπορεί να βγάλει συμπεράσματα από αβέβαια δεδομένα.
- Μπορεί να εξηγήσει τον δρόμο συλλογισμού του κατά νοήμονα τρόπο.
- Τα γεγονότα και ο μηχανισμός εξαγωγής συμπερασμάτων διαχωρίζονται σαφώς.
- Σχεδιάζεται έτσι ώστε να αναπτύσσεται βαθμιαία.
- Ένα έμπειρο σύστημα βασίζεται σε κανόνες.
- Η έξοδος του έχει τη μορφή συμβουλών και όχι τη μορφή πινάκων ή οπτικών εικόνων.

3.2.2 Δομή ενός έμπειρου συστήματος

Κάθε πλήρες έμπειρο σύστημα έχει τέσσερις βασικές συνιστώσες:

1. τη βάση γνώσης
2. τη μηχανή εξαγωγής συμπερασμάτων
3. τη συνιστώσα λήψης της γνώσης
4. τη συνιστώσα ερμηνείας / διασύνδεσης.

Για να είναι πλήρες ένα έμπειρο σύστημα πρέπει να περιέχει και τις τέσσερις συνιστώσες. Στο παρακάτω σχήμα φαίνεται η οργάνωση και η σύνδεση των τεσσάρων αυτών συνιστωσών.



Εικόνα 3 Τυπική μορφή έμπειρου συστήματος

3.2.3 Η βάση της γνώσης

Μία βάση γνώσης περιέχει γεγονότα και κανόνες. Τα γεγονότα είναι βραχυπρόθεσμη πληροφορία που μπορεί να αλλάζει γρήγορα. Οι κανόνες είναι μακροπρόθεσμη πληροφορία

που περιγράφει πως παράγονται νέα γεγονότα ή νέες υποθέσεις από αυτά που είναι διαθέσιμα μέχρι τώρα.

Η βάση γνώσης δεν είναι μία απλή βάση δεδομένων. Τα γεγονότα σε μία βάση δεδομένων είναι συνήθως παθητικά δηλαδή είτε υπάρχουν είτε δεν υπάρχουν. Αντίθετα η βάση γνώσης είναι πιο δημιουργική και προσπαθεί ενεργά να συμπληρώσει την πληροφορία που λείπει.

Οι κανόνες παραγωγής είναι ένας κατάλληλος τρόπος ενσωμάτωσης ανθρωπίνων κανόνων σκέψης και εμπειρίας. Συνήθως έχουν τη μορφή IF – THEN (EAN – TOTE) ή είναι δένδρα αποφάσεων.

3.2.4 Η μηχανή εξαγωγής συμπερασμάτων

Όλα τα έμπειρα συστήματα χρησιμοποιούν αλγόριθμους αναζήτησης. Πολλές τεχνικές έχουν υιοθετηθεί για να καταστήσουν αυτές τις αναζητήσεις αποδοτικότερες. Οι αλγόριθμοι Branch and bound, pruning, depth-first search, and breadth-first search είναι μερικές από τις τεχνικές αναζήτησης που έχουν εξερευνηθεί. Εξαιτίας της εντατικής αναζήτησης είναι καλό να υπάρχουν καλές στρατηγικές ελέγχου κατά την διάρκεια της διαδικασίας.

Οι δύο πιο γενικές στρατηγικές εξαγωγής συμπερασμάτων είναι:

- ορθή αλυσίδα (Forward Chaining): Αυτή η μέθοδος περιλαμβάνει τον έλεγχο της υπόθεσης ενός κανόνα για να καθοριστεί εάν είναι αληθινή ή ψεύτικη. Αν η υπόθεση είναι αληθής τότε και το αποτέλεσμα του κανόνα είναι αληθές. Δηλαδή πηγαίνουμε από τα δεδομένα προς τις υποθέσεις. Η διαδικασία επαναλαμβάνεται μέχρι να βρεθεί η λύση ή να φτάσει στο τέλος η αναζήτηση.
- Ανάστροφη αλυσίδα (Backward Chaining): η ανάστροφη αλυσίδα είναι η αντιστροφή της ορθής αλυσίδας. Χρησιμοποιείται για να οπισθοδρομήσει από έναν στόχο στα μονοπάτια που οδηγούν στο στόχο. Η ανάστροφη αλυσίδα είναι πολύ καλή όταν είναι γνωστές όλες οι εκβάσεις και ο αριθμός πιθανών εκβάσεων δεν είναι μεγάλος. Σε αυτήν την περίπτωση, ένας στόχος διευκρινίζεται και το έμπειρο σύστημα προσπαθεί να καθορίσει ποιοι όροι απαιτούνται για να φθάσουν στο διευκρινισμένο στόχο. Στην περίπτωση αυτή πηγαίνουμε από την υπόθεση στα δεδομένα και προσπαθούμε να βρούμε δεδομένα προς απόδειξη ή απόρριψη μιας υπόθεσης.

3.3 Έμπειρο σύστημα ‘Συλλογισμού Βασισμένο σε

Περιπτώσεις’ (Case-Based Reasoning expert system)

Το CBR έμπειρο σύστημα είναι ένα παράδειγμα επίλυσης προβλήματος που είναι από πολλές απόψεις πλήρως διαφορετικό από άλλες σημαντικές προσεγγίσεις της τεχνητής νοημοσύνης.

Αντί να στηριχθεί απλώς στη γενική γνώση μιας περιοχής προβλήματος το CBR είναι σε θέση να χρησιμοποιήσει τη γνώση των προηγούμενων πεπειραμένων συγκεκριμένων καταστάσεων προβλήματος (περιπτώσεις). Ένα νέο πρόβλημα λύνεται με την εύρεση μιας παρόμοιας προηγούμενης περίπτωσης και την επαναχρησιμοποίηση της στη νέα κατάσταση προβλήματος. Μια δεύτερη σημαντική διαφορά είναι ότι το CBR έχει μια προσέγγιση στην επαυξητική, συνεχή εκμάθηση, δεδομένου ότι μια νέα εμπειρία διατηρείται κάθε φορά που έχει λυθεί ένα πρόβλημα, καθιστώντας το αμέσως διαθέσιμο για τα μελλοντικά προβλήματα.

Η επαναχρησιμοποίηση των προηγούμενων περιπτώσεων είναι ένας ισχυρός και συχνά εφαρμοσμένος τρόπος για να λυθούν διάφορα προβλήματα. Για παράδειγμα ένας γιατρός μπορεί να κάνει πιο εύκολα διάγνωση μίας ασθένειας βασιζόμενος στην εμπειρία του και να προτείνει και την καλύτερη θεραπεία.

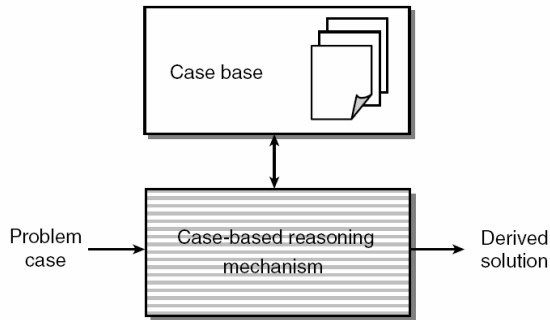
Στην ορολογία CBR [11] μια περίπτωση δείχνει συνήθως μια κατάσταση προβλήματος. Μια προηγούμενος πεπειραμένη κατάσταση, που έχει καταγραφεί με τέτοιο τρόπο ώστε να μπορεί να επαναχρησιμοποιηθεί στην επίλυση των μελλοντικών προβλημάτων, αναφέρεται ως προηγούμενη περίπτωση. Αντίστοιχα μια καινούργια περίπτωση ή μια άλυτη περίπτωση είναι η περιγραφή ενός νέου προβλήματος που επιθυμούμε να λύσουμε.

Ένα πολύ σημαντικό χαρακτηριστικό γνώρισμα του CBR είναι η ικανότητά του στην εκμάθηση. Η εκμάθηση σε CBR εμφανίζεται ως φυσικό υποπροϊόν της επίλυσης προβλήματος. Όταν ένα πρόβλημα λύνεται επιτυχώς, η εμπειρία διατηρείται προκειμένου να λυθούν τα παρόμοια προβλήματα στο μέλλον. Όταν μια προσπάθεια να λυθεί ένα πρόβλημα αποτυγχάνει, ο λόγος για την αποτυχία προσδιορίζεται και αναφέρεται, προκειμένου να αποφευχθεί το ίδιο λάθος στο μέλλον.

Οι βασικότερες διαδικασίες που πρέπει να περιέχει ένα CBR είναι ο προσδιορισμός μίας τρέχουσας κατάστασης προβλήματος, η εύρεση μιας προηγούμενης περίπτωσης παρόμοιας με τη νέα, η πρόταση μιας λύσης στο τρέχον πρόβλημα, η αξιολόγηση της προτεινόμενης λύσης και η ενημέρωση του συστήματος με την εκμάθηση από αυτήν την εμπειρία.

Το CBR καλύπτει μια σειρά διαφορετικών μεθόδων για τη γνώση που διατηρείται σε προηγούμενες περιπτώσεις. Οι περιπτώσεις μπορούν να κρατηθούν ως συγκεκριμένη εμπειρία ή ένα σύνολο παρόμοιων περιπτώσεων μπορεί να διαμορφώσει μια γενικευμένη περίπτωση. Οι περιπτώσεις μπορούν να αποθηκευτούν ως χωριστές μονάδες γνώσης ή ως υπομονάδες και να παραμείνουν μέσα στη δομή γνώσης. Οι περιπτώσεις μπορούν να συνταχθούν από ένα προταγμένο ή ανοικτό σύνολο και μέσα σε μια επίπεδη ή ιεραρχική δομή δεικτών. Η λύση από μια προηγούμενη περίπτωση μπορεί να εφαρμοστεί άμεσα στο παρόν πρόβλημα ή να τροποποιηθεί σύμφωνα με τις διαφορές μεταξύ των δύο περιπτώσεων. Το ταίριασμα των περιπτώσεων, η προσαρμογή των λύσεων και η εκμάθηση από μια εμπειρία μπορούν να καθοδηγηθούν και να υποστηριχθούν από ένα βαθύ πρότυπο της

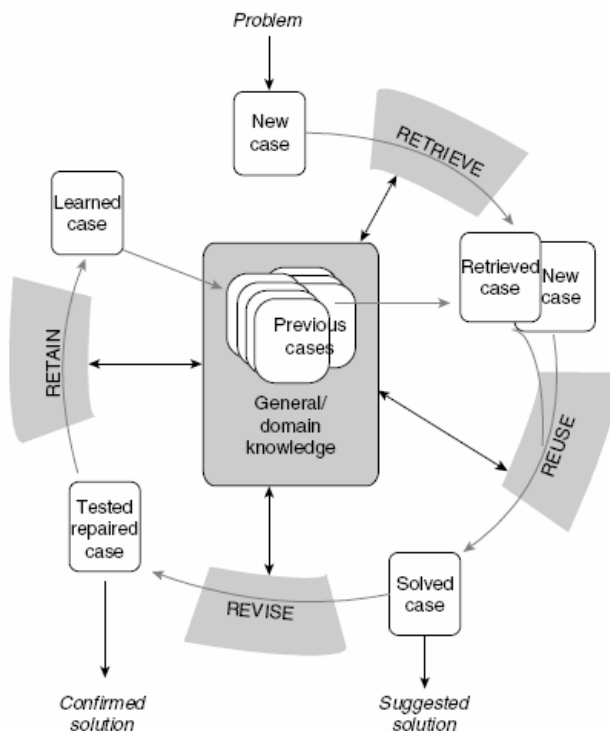
γενικής γνώσης περιοχών, από την πιο ρηχή και συνταγμένη γνώση ή να βασιστούν σε μια προφανή, συντακτική ομοιότητα μόνο. Οι μέθοδοι CBR μπορούν να είναι καθαρώς ανεξάρτητες και αυτόματες ή μπορούν να αλληλεπιδράσουν με το χρήστη για την υποστήριξη και την καθοδήγηση των επιλογών τους. Στην Εικόνα 4 φαίνεται η βασική λειτουργία ενός CBR.



Εικόνα 4 Ένα έμπειρο σύστημα Case-Based Reasoning

3.3.1 Κύκλος ζωής

Ο κύκλος ζωής επίλυσης προβλήματος σε ένα σύστημα CBR αποτελείται ουσιαστικά από τα ακόλουθα τέσσερα μέρη



Εικόνα 5 Κύκλος ζωής του CBR

1. Ανάκτηση των παρόμοιων προηγούμενων πεπειραμένων περιπτώσεων των οποίων πρόβλημα θεωρείται παρόμοιο.

2. Επαναχρησιμοποίηση των περιπτώσεων με την αντιγραφή ή την ενσωμάτωση των λύσεων από τις περιπτώσεις που ανακτήθηκαν.
3. Αναθεώρηση ή προσαρμογή της λύσης που ανακτήθηκε σε μία προσπάθεια να λυθεί το νέο πρόβλημα.
4. Διατήρηση της νέας λύσης μόλις επιβεβαιωθεί ή επικυρωθεί.

Σε πολλές πρακτικές εφαρμογές, τα στάδια της επαναχρησιμοποίησης και της αναθεώρησης είναι δύσκολο να διακριθούν και διάφοροι ερευνητές χρησιμοποιούν ένα ενιαίο στάδιο προσαρμογής που τα αντικαθιστά και τα συνδυάζει.

3.3.2 Πλεονεκτήματα

- Μείωση του στόχου απόκτησης γνώσης. Με την εξάλειψη της ανάγκης ενός προτύπου ή ενός συνόλου κανόνων, όπως είναι απαραίτητο στα βασισμένα στους κανόνες συστήματα, οι στόχοι απόκτησεων γνώσης CBR αποτελούνται πρώτιστα από τη συλλογή σχετικών εμπειριών /περιπτώσεων, την αντιπροσώπευση και την αποθήκευσή τους.
- Αποφυγή των λαθών που έχουν γίνει στο παρελθόν. Στα συστήματα που καταγράφουν τις αποτυχίες όπως και τις επιτυχίες και πληροφορίες για τον λόγο που προκάλεσε τις αποτυχίες στο παρελθόν, μπορεί να χρησιμοποιηθούν για να προβλέψουν τις πιθανές αποτυχίες στο μέλλον.
- Πρόβλεψη της πιθανής επιτυχίας μιας προσφερόμενης λύσης. Όταν οι πληροφορίες αποθηκεύονται σχετικά με το επίπεδο επιτυχίας των προηγούμενων λύσεων, το CBR είναι σε θέση να προβλέψει την επιτυχία της λύσης που προτείνεται για ένα τρέχον πρόβλημα.
- Εκμάθηση με την πάροδο του χρόνου. Εάν οι περιπτώσεις λύσης εξετάζονται στη συνέχεια στο πραγματικό κόσμο και ένα επίπεδο επιτυχίας καθορίζεται για εκείνες τις λύσεις, αυτές οι περιπτώσεις μπορούν να προστεθούν στη βάση περίπτωσης και να χρησιμοποιηθούν για να βοηθήσουν στην επίλυση των μελλοντικών προβλημάτων.

Υπάρχουν τρία γενικά ζητήματα που πρέπει να εξεταστούν κατά τη δημιουργία ενός CBR

- Η δομή και η αναπαράσταση των περιπτώσεων
- Το πρότυπο μνήμης που χρησιμοποιείται για την οργάνωση
- Η επιλογή των δεικτών που χρησιμοποιούνται για να προσδιορίσουν κάθε περίπτωση.

3.3.3 Αναπαράσταση κατά περίπτωση

Οι περιπτώσεις σε ένα CBR μπορούν να αντιπροσωπεύσουν πολλούς διαφορετικούς τύπους γνώσεων που μπορούν να αποθηκευτούν με πολλά διαφορετικά αντιπροσωπευτικά σχήματα. Ο προοριζόμενος σκοπός ενός συστήματος CBR θα επηρεάσει πολύ τι θα αποθηκεύεται. Επομένως σε κάθε τύπο συστήματος CBR, μια περίπτωση μπορεί να αντιπροσωπεύσει κάτι διαφορετικό. Παραδείγματος χάριν οι περιπτώσεις θα μπορούσαν να αντιπροσωπεύουν ανθρώπους, αντικείμενα, καταστάσεις, διαγνώσεις, σχέδια ή αποφάσεις.

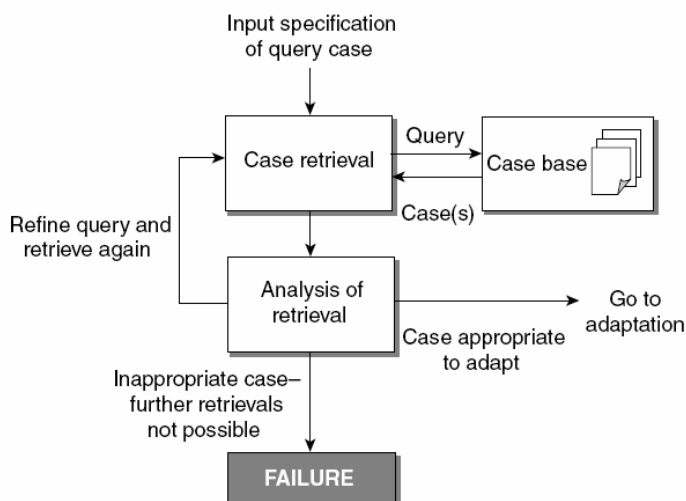
Πολύ σημαντικό είναι και η ανάθεση δεικτών στις περιπτώσεις για τη μελλοντική ανάκτηση και σύγκριση. Η επιλογή των δεικτών είναι σημαντική για την ανάκτηση της σωστής περίπτωσης στη σωστή στιγμή. Αυτό επειδή οι δείκτες μιας περίπτωσης θα καθορίσουν ποια θα ανακτηθεί στο μέλλον.

3.3.4 Ανάκτηση κατά περίπτωση

Η ανάκτηση περίπτωσης είναι η διαδικασία εύρεσης της πιο ταιριαστής προηγούμενης περίπτωσης με την παρούσα περίπτωση. Για να πραγματοποιηθεί αποτελεσματική ανάκτηση περίπτωσης πρέπει να υπάρξουν κριτήρια επιλογής που να καθορίζουν πως μια περίπτωση θεωρείται κατάλληλη για την ανάκτηση και έναν μηχανισμό αναζήτησης. Τα κριτήρια επιλογής είναι απαραίτητα για τον καθορισμό της καλύτερης ανάκτησης περίπτωσης. Τα κριτήρια επιλογής περίπτωσης εξαρτώνται εν μέρει από την διαδικασία ανάκτησης περίπτωσης και τι θεωρούμε ως σωστή ανάκτηση. Συνήθως η διαδικασία ανάκτησης ψάχνει για μια ολόκληρη περίπτωση, τα χαρακτηριστικά γνωρίσματα της οποίας συγκρίνονται με εκείνα της παρούσας περίπτωσης (Εικόνα 6). Εντούτοις, υπάρχουν φορές που επιδιώκεται ένα μέρος μιας προηγούμενης περίπτωσης. Η ανάκτηση είναι ένας σημαντικός ερευνητικός τομέας σε CBR. Οι συνηθέστερα ερευνημένες τεχνικές ανάκτησης είναι οι K-κοντινότεροι γείτονες (K-NN), τα δέντρα απόφασης και τα παράγωγά τους. Αυτές οι τεχνικές περιλαμβάνουν την ανάπτυξη μιας μετρήσιμης ομοιότητας των περιπτώσεων προς ανάκτηση.

- Ανάκτηση πλησίον-γειτόνων (Nearest-neighbor retrieval). Στην ανάκτηση πλησίον-γειτόνων η περίπτωση που ανακτάται επιλέγεται όταν το σταθμισμένο ποσό των χαρακτηριστικών γνωρισμάτων του που ταιριάζουν με την παρούσα περίπτωση είναι μεγαλύτερο από άλλες περιπτώσεις στη βάση περίπτωσης. Πιο απλά, με όλα τα χαρακτηριστικά γνωρίσματα που σταθμίζονται εξίσου, μια περίπτωση που ταιριάζει με την παρούσα περίπτωση στα χαρακτηριστικά γνωρίσματα n θα ανακτηθεί, παρά μια περίπτωση που ταιριάζει στα χαρακτηριστικά γνωρίσματα k , όπου $k < n$. Τα χαρακτηριστικά γνωρίσματα που θεωρούνται σημαντικότερα σε μια κατάσταση επίλυσης προβλήματος μπορούν να έχουν μεγαλύτερο βάρος.

- Επαγωγικές προσεγγίσεις(Inductive approaches). Όταν οι επαγωγικές προσεγγίσεις χρησιμοποιούνται για να καθορίσουν τη δομή των CBR, η οποία καθορίζει την σημασία των χαρακτηριστικών γνωρισμάτων για την διάκριση των παρόμοιων περιπτώσεων, η προκύπτουσα ιεραρχική δομή της περίπτωσης παρέχει ένα μειωμένο διάστημα αναζήτησης για ανάκτηση. Αυτό μπορεί, στη συνέχεια, να μειώσει το χρόνο αναζήτησης.
- Προσεγγίσεις καθοδηγούμενης γνώσης(Knowledge-guided approaches). Οι προσεγγίσεις καθοδηγούμενης γνώσης στην ανάκτηση χρησιμοποιούν μία περιοχή γνώσης για να καθορίσουν τα χαρακτηριστικά γνωρίσματα μιας περίπτωσης που είναι σημαντικά για να ανακτηθεί εκείνη η περίπτωση στο μέλλον. Σε μερικές καταστάσεις τα διαφορετικά χαρακτηριστικά γνωρίσματα μιας περίπτωσης έχουν διαφορετικά επίπεδα σπουδαιότητας ή συμβολής στα επίπεδα επιτυχίας που συνδέονται με εκείνη την περίπτωση.
- Επικυρωμένη ανάκτηση(Validate retrieval). Έχουν υπάρξει πολυάριθμες προσπάθειες στη βελτίωση της ανάκτησης. Μία από αυτές είναι η επικυρωμένη ανάκτηση η οποία λαμβάνει μέρος σε δύο φάσεις. Η φάση 1 περιλαμβάνει την ανάκτηση όλων των περιπτώσεων που εμφανίζονται να είναι σχετικές με ένα πρόβλημα βασισμένο στα κύρια χαρακτηριστικά γνωρίσματα της παρούσας περίπτωσης. Η φάση 2 περιλαμβάνει την άντληση περισσότερων διακριτικών χαρακτηριστικών γνωρισμάτων από την αρχική ομάδα ανακτημένων περιπτώσεων για να καθορίσει εάν αυτές οι περιπτώσεις ισχύουν στη παρούσα κατάσταση.



Εικόνα 6 Ανάκτηση περίπτωσης του CBR

3.3.5 Προσαρμογή περίπτωσης

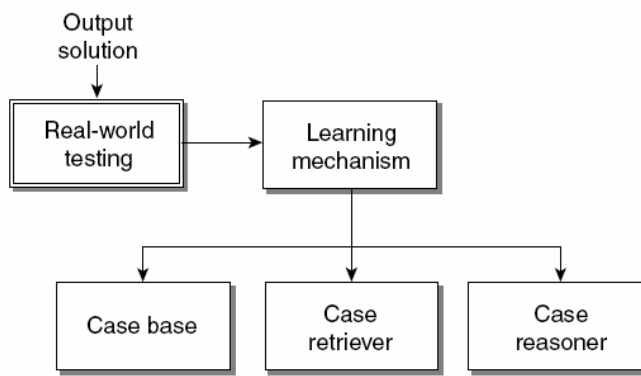
Η προσαρμογή περίπτωσης είναι η διαδικασία μετατροπής μιας λύσης που ανακτάται σε μια λύση κατάλληλη για το τρέχον πρόβλημα. Έχει υποστηριχτεί ότι η προσαρμογή μπορεί να είναι το σημαντικότερο βήμα CBR δεδομένου ότι προσθέτει τη νοημοσύνη σε αυτό που ειδάλλως θα ήταν απλά ένα ταίριασμα. Διάφορες μέθοδοι μπορούν να υιοθετηθούν για να πραγματοποιήσουν την προσαρμογή περίπτωσης:

- Η ανακτημένη λύση θα μπορούσε να χρησιμοποιηθεί ως λύση στο τρέχον πρόβλημα χωρίς τροποποίηση ή με τις τροποποιήσεις όπου η λύση δεν είναι εξ ολοκλήρου κατάλληλη για τη παρούσα κατάσταση.
- Τα βήματα ή οι διαδικασίες που ακολουθήθηκαν για να λάβουν την προηγούμενη λύση, μπορούν να επαναληφθούν χωρίς τροποποίηση ή με τις τροποποιήσεις όπου δεν υπήρχαν ικανοποιητικά αποτελέσματα στην προηγούμενη περίπτωση.
- Όπου περισσότερες από μια περιπτώσεις έχουν ανακτηθεί, μια λύση θα μπορούσε να προέλθει από τις πολλαπλάσιες περιπτώσεις ή διάφορες εναλλακτικές λύσεις θα μπορούσαν να παρουσιαστούν.

3.3.6 Εκμάθηση της μηχανής

Μόλις παραχθεί μια κατάλληλη λύση, υπάρχει κάποια προσδοκία ότι η λύση θα εξεταστεί στην πραγματικότητα. Για να εξεταστεί μια λύση θα πρέπει να ορίσουμε τον τρόπο που μπορεί να εξεταστεί και πώς η έκβαση της δοκιμής θα ταξινομηθεί ως επιτυχία ή αποτυχία. Με άλλα λόγια, μερικά κριτήρια πρέπει να είναι καθορισμένα για την εκτίμηση απόδοσης της προσφερόμενης λύσης. Χρησιμοποιώντας την αξιολόγηση ένα σύστημα CBR μπορεί να ενημερωθεί για να λάβει υπόψη οποιεσδήποτε νέες πληροφορίες που αποκαλύπτονται στην επεξεργασία της νέας λύσης. Αυτές οι πληροφορίες μπορούν να προστεθούν στο σύστημα κι έτσι όσο περισσότερες πληροφορίες αποθηκεύονται σε ένα CBR τόσο πιο πιθανό είναι να βρεθεί μία λύση. Η εκμάθηση μπορεί να εμφανιστεί με διάφορους τρόπους. Η προσθήκη ενός νέου προβλήματος, η λύση του και η έκβαση του στο CBR είναι μια κοινή μέθοδος. Η προσθήκη των περιπτώσεων θα αυξήσει τη σειρά των καταστάσεων που καλύπτονται από τις αποθηκευμένες περιπτώσεις και θα μειώσει τη μέση απόσταση μεταξύ ενός διανύσματος εισαγωγής και του πιο στενού αποθηκευμένου διανύσματος.

Μια δεύτερη μέθοδος σε ένα σύστημα CBR είναι να αξιολογήσει τη λύση και να τροποποιήσει τους δείκτες των αποθηκευμένων περιπτώσεων ή να τροποποιήσει τα κριτήρια για την ανάκτηση περίπτωσης.



Εικόνα 7 Διαδικασία εκμάθησης της μηχανής του CBR

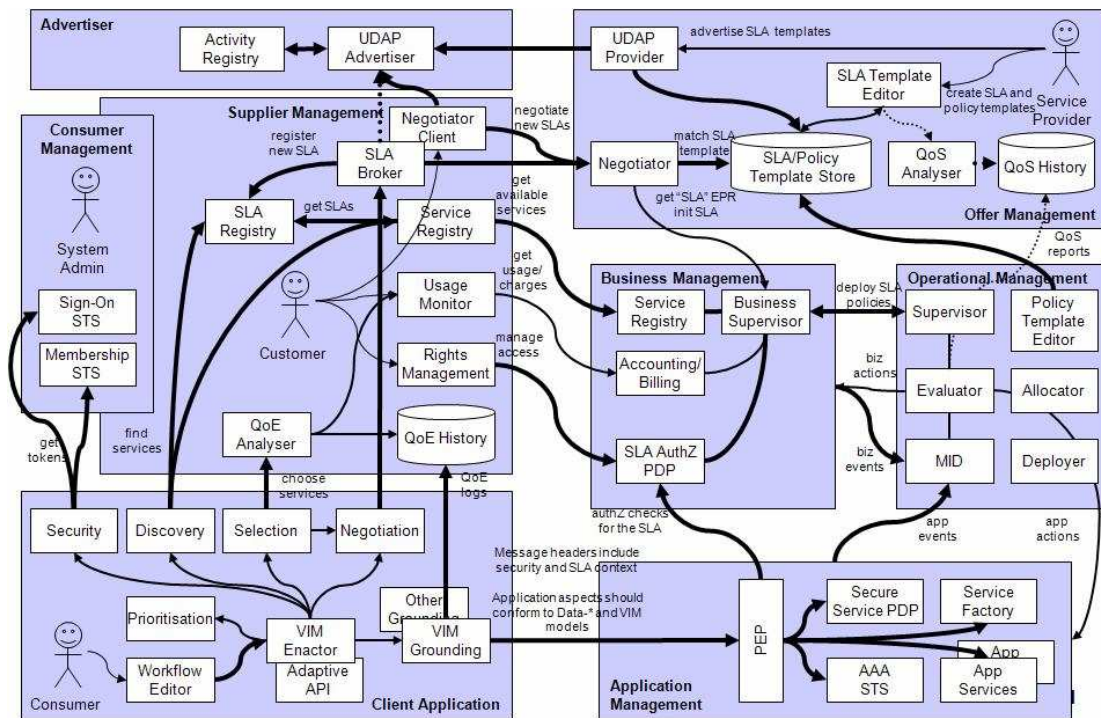
4

Εκτιμητής Φόρτου Εργασίας για εφαρμογές σε περιβάλλον πλέγματος

Στο κεφάλαιο αυτό θα αναπτύξουμε την μεθοδολογία μας για το γενικότερο μοντέλο πάνω στο οποίο θα βασιστεί η σχεδίαση ενός Εκτιμητή Φόρτου Εργασίας εφαρμογών που εκτελούνται σε περιβάλλον πλέγματος. Προτού προχωρήσουμε στην ανάπτυξη θα ήταν καλό να κατανοήσουμε ποια είναι η λειτουργία ενός εκτιμητή μέσα στην γενικότερη λειτουργία του πλέγματος καθώς και να ορίσουμε το μοντέλο του φόρτου εργασίας πάνω στο οποίο θα βασιστεί η εκτίμηση.

4.1 Αρχιτεκτονική των υπολογιστικών πλεγμάτων

Το παρακάτω διάγραμμα απεικονίζει τις διάφορες αρχιτεκτονικές οντότητες ενός στιγμιότυπου της γενικότερης αρχιτεκτονικής των πλεγμάτων. Παρουσιάζει επίσης μια επισκόπηση για το πως αυτές οι οντότητες επικοινωνούν η μια με την άλλη βασισμένη στις αλληλεπιδράσεις τους. Οι αρχιτεκτονικές οντότητες είναι συστατικά που εφαρμόζουν τις διάφορες διαδικασίες για την λειτουργία του πλέγματος [25]. Στο κεφάλαιο αυτό σκοπός μας δεν είναι να μπούμε σε λεπτομερή ανάλυση της λειτουργίας κάθε οντότητας ξεχωριστά αλλά να κατανοήσουμε την γενικότερη λειτουργία και την αρχιτεκτονική ενός πλέγματος. Είναι προφανές ότι η δομή και η αρχιτεκτονική, στις οποίες θα βασιστεί το στήσιμο ενός πλέγματος, θα καθορίσουν τον τρόπο λειτουργίας του και την απόδοσή του.

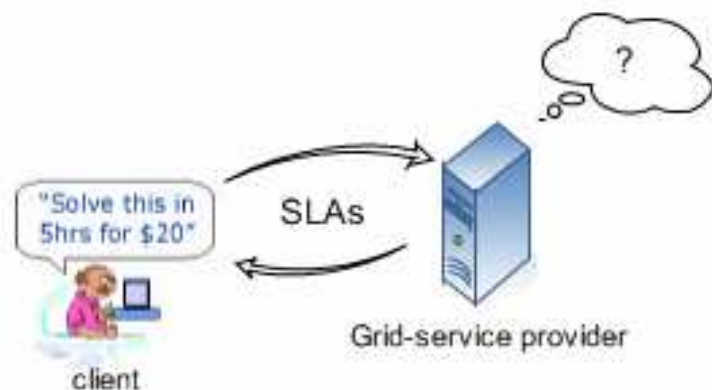


Εικόνα 8 Στιγμιότυπο αρχιτεκτονικής υπολογιστικού πλέγματος

Οι αρχιτεκτονικές αρχές προκύπτουν άμεσα από το όραμα των πλεγμάτων να είναι οικονομικά βιώσιμα, να είναι δυνατή η ανάπτυξή τους στα νέα και στα υπάρχοντα επιχειρησιακά πρότυπα, να είναι εύκολη η επέκταση και η συντήρηση των εφαρμογών και να παρέχουν ασφάλεια στις επιχειρήσεις και στους πελάτες. Η γενική άποψη είναι η δημιουργία μιας υποδομής που ικανοποιεί τα νέα επαγγελματικά μοντέλα τα οποία θέτουν νέες απαιτήσεις όπως την αποδοτικότερη χρήση της υποδομής, την ευελιξία των πόρων και την γρηγορότερη πρόσβαση στη σημαντική υπολογιστική δύναμη. Για την ικανοποίηση αυτών των μοντέλων οι βασικές αρχιτεκτονικές αρχές που χρησιμοποιούνται πρέπει να περιλαμβάνουν:

- Δυναμικά καθοδηγούμενα SLA (SLA-Driven Dynamics): Το SLA είναι κρίσιμη δομική μονάδα στην υποδομή και η δυναμική συμπεριφορά τους είναι κεντρική στη λειτουργία οποιουδήποτε πλέγματος.
- Δυναμική συνομοσπονδία (Dynamic Federation): Η δυναμική συνομοσπονδία των πόρων είναι ένας παράγοντας κλειδί στην καθιέρωση των πλεγμάτων στον επιχειρηματικό χώρο.
- Ελάχιστη υποδομή πλέγματος: Οποιοδήποτε πλέγμα πρέπει να είναι απλό ώστε να εξασφαλίζει την εύκολη συντήρηση και την επεκτασιμότητα των εφαρμογών.

Η ενσωμάτωση ενός εκτιμητή φόρτου εργασίας ως μία ξεχωριστή λειτουργία στην υποδομή του πλέγματος μπορεί να βοηθήσει στην ικανοποίηση των παραπάνω απαιτήσεων και ειδικότερα στην δυναμική συμπεριφορά των SLAs. Όταν ένας πελάτης αιτεί μία υπηρεσία πλέγματος ταυτόχρονα αποστέλλει και το προτεινόμενο του SLA. Το πλέγμα πρέπει να είναι σε θέση να αξιολογήσει το προτεινόμενο SLA και να απαντήσει αρνητικά ή να προτείνει ένα δικό του (Εικόνα 9).



Εικόνα 9 Αίτηση υπηρεσίας πλέγματος

Η εκ των προτέρων γνώση του φόρτου της εργασίας θα ωφελούσε τόσο το πλέγμα όσο και τον πελάτη. Τα οφέλη που θα αποκόμιζε το πλέγμα είναι ότι θα βελτιωνόταν η ικανότητά του στην λήψη σωστών αποφάσεων όσον αφορά την αποδοχή, άρνηση ή διαπραγμάτευση ενός SLA καθώς και στην καλύτερη κατανομή και διαχείριση των πόρων. Από την μεριά του πελάτη η γνώση του φόρτου θα τον βοηθούσε στην πρόταση ενός λογικού SLA που θα αντιπροσωπεύει τις ανάγκες του.

Κατανοούμε λοιπόν ότι ο εκτιμητής φόρτου δεν είναι ένας απλός μηχανισμός αλλά μία νέα οντότητα-λειτουργία μέσα στην αρχιτεκτονική του πλέγματος. Η λειτουργία αυτή θα ήταν καλύτερο να παρέχεται ως υπηρεσία ιστού και να εκτελείται στην μεριά του πελάτη για να μην απασχολεί πόρους από το πλέγμα. Τεχνικά και σύμφωνα με το παραπάνω σχήμα της αρχιτεκτονικής η οντότητα του εκτιμητή θα επικοινωνεί και θα δίνει τις πληροφορίες φόρτου στην οντότητα 'negotiation' η οποία αλληλεπιδρά με την 'SLA Broker' για την διαπραγμάτευση του SLA.

4.2 Μοντέλο φόρτου εργασίας

Η πρόβλεψη του συνολικού χρόνου εκτέλεσης που μια εργασία θα διαρκέσει όταν εκτελείται σε μια συγκεκριμένη πλατφόρμα είναι σημαντική μέσα σε εφαρμογές επειδή οδηγεί σε μια καλύτερη χρησιμοποίηση των υπάρχοντων υπολογιστικών πόρων. Θα καθορίσουμε ένα σύνολο από τις παραμέτρους φόρτου εργασίας που συσχετίζονται άμεσα με τις παραμέτρους

των πόρων που έχουν επιπτώσεις στο χρόνο εκτέλεσης. Το πρότυπο ενός φόρτου εργασίας παράγει μια αποδοτική πρόβλεψη για μια εφαρμογή βασισμένη στους διευκρινισμένους πόρους, τις δυνατότητες και τις παραμέτρους του προβλήματος.

4.2.1 Παράμετροι πόρων πλατφόρμας

Ο προσδιορισμός των παραμέτρων των πόρων πλατφόρμας είναι πραγματικά το μέρος που αφορά την πλευρά του παρόχου της υπηρεσίας και έχει αναφερθεί ήδη ως Capacity Estimator [7]. Ο προσδιορισμός των παραμέτρων γίνεται σε κάθε έναν από τους πόρους του συστήματος και παρέχει τις μετρημένες ποσότητες που έχουν επιπτώσεις στην απόδοση εκτέλεσης ενός συγκεκριμένου πόρου. Είναι απαραίτητο να μετρηθεί η απόδοση μιας μηχανής, η οποία θεωρείται ως πόρος, έτσι ώστε να μπορούμε να κάνουμε τις εκτιμήσεις για το χρόνο εφαρμογής μιας εργασίας. Σε ένα εμπορικά προσανατολισμένο υπολογιστικό πλέγμα η εκτίμηση ικανότητας των πόρων πρέπει να είναι ανεξάρτητη από τις εφαρμογές ώστε να υπάρξει ένας ενιαίος και ενοποιημένος τρόπος υπολογισμού του χρόνου εκτέλεσης για κάθε διαφορετική εφαρμογή. Συνεπώς οι παράμετροι πρέπει να είναι καθορισμένοι με τέτοιο τρόπο ώστε να απεικονίζουν τις δυνατότητες των υπολογιστικών πόρων και να μπορούν να μετρηθούν σε κάθε διαφορετική πλατφόρμα. Η διαδικασία του Capacity Estimator πρέπει να είναι διαχωρισμένη από την εκτέλεση της εφαρμογής ώστε να επιτρέπει την ανάπτυξη ενός προτύπου που θα είναι σε θέση να υπολογίσει το χρόνο εκτέλεσης ξεχωριστά για κάθε εφαρμογή. Οι παράμετροι απόδοσης που πρέπει να καθοριστούν για ένα Capacity Estimator που ενσωματώνεται σε ένα διανεμημένο ετερογενές σύστημα συνοψίζονται στον παρακάτω πίνακα.

Πίνακας 1 Παράμετροι Πόρων Πλατφόρμας

Παράμετροι	Μονάδα μέτρησης
Million Float Operations / sec	MFLOP/sec
Memory average I/O bandwidth	MB/sec
Disk average I/O bandwidth	KB/sec
Send communication bandwidth	KB/sec
Send communication latency	msec
Receive communication bandwidth	KB/sec
Receive communication latency	msec
Minimum Free Memory	MB
Minimum Free Disk Space	KB
Queue time interval	sec

4.2.2 Παράμετροι φόρτου εργασίας

Οι παράμετροι φόρτου εργασίας καθορίζονται από κοινού με τις παραμέτρους των πόρων. Προκειμένου να υπολογιστεί ο φόρτος εργασίας από μια εφαρμογή πρέπει να εξαγάγουμε τα χαρακτηριστικά γνωρίσματα που περιγράφουν τη συγκεκριμένη εφαρμογή. Έτσι, για κάθε εφαρμογή που θα ενσωματωθεί στο πλέγμα, ένας διαφορετικός εκτιμητής φόρτου εργασίας πρέπει να σχεδιαστεί. Οι παράμετροι φαίνονται στον παρακάτω πίνακα.

Πίνακας 2 Παράμετροι φόρτου εργασίας

Παράμετροι	Μονάδα μέτρησης
Million Float Operations / task	MFLOP
MB of memory I/O messages / task	MB
KB of disk I/O messages / task	KB
Maximum Memory Used	MB
Maximum Disk Space Used	KB
Amount of Bytes Sent	KB
Amount of Bytes Received	KB

4.2.3 Εκτίμηση του συνολικού χρόνου εκτέλεσης

Η εκτίμηση του συνολικού χρόνου εκτέλεσης για μια εργασία που ορίζεται σε έναν συγκεκριμένο πόρο του πλέγματος, είναι μέρος του γενικότερου προβλήματος εκτίμησης του συνολικού χρόνου που απαιτείται από το σύστημα για να ολοκληρώσει μια εργασία αρχίζοντας από τη στιγμή που η εργασία καταχωρείται στο σύστημα από τον πελάτη μέχρι τη στιγμή όπου παραδίδεται επιτυχώς από τον προμηθευτή πίσω στον πελάτη. Αυτός ο συνολικός χρόνος εκτέλεσης περιλαμβάνει τα εξής:

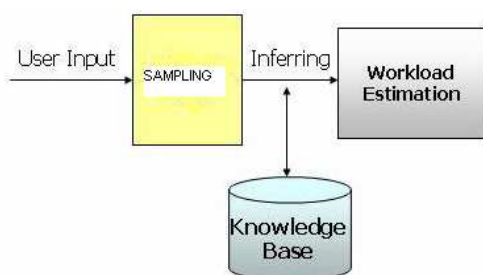
1. Χρόνος επικοινωνίας (XE): ο χρόνος που απαιτείται για τη διαβίβαση των δεδομένων εισόδου στον προμηθευτή (αποστολή) και ο χρόνος που απαιτείται για τη διαβίβαση των δεδομένων εξόδου (αποτελέσματα) από τον προμηθευτή στον πελάτη (λήψη).
2. Χρόνος αναμονής (XA): είναι το χρονικό διάστημα που μια εργασία περιμένοντας τη σειρά της για εκτέλεση βρίσκεται στην ουρά αναμονής.
3. Χρόνος εκτέλεσης (XEK): είναι ο υπολογιστικός χρόνος για την ολοκλήρωση μιας εργασίας από έναν δεδομένο πόρο. Η εκτίμηση XEK καθορίζεται μέσω των παραμέτρων φόρτου εργασίας εφαρμογής.

Ο συνολικός χρόνος εκτέλεσης είναι το άθροισμα των τριών χρονικών τιμών. Στην παρούσα μελέτη μας ενδιαφέρει ο ΕΦΕ να βρίσκει τον χρόνο εκτέλεσης (XEK) καθώς οι άλλοι δύο χρόνοι εξαρτώνται από άλλους παράγοντες που αφορούν την υποδομή του πλέγματος.

4.3 Εκτιμητής Φόρτου Εργασίας

Είδαμε λοιπόν πόσο σημαντικός είναι για ένα πλέγμα και για τους χρήστες του ο εκτιμητής φόρτου εργασίας (EFE). Προκειμένου να οριστεί κατάλληλα ο φόρτος εργασίας σε κάθε κόμβο ο χρονοδρομολογητής πρέπει να ξέρει τη διανομή φόρτου εργασίας του προγράμματος εφαρμογής. Για διάφορες εφαρμογές, όπως ο πολλαπλασιαστής πινάκων που λειτουργούν γραμμικά είναι εύκολο να υπολογιστεί ο φόρτος εργασίας. Ωστόσο υπάρχουν εφαρμογές που η πρόβλεψη του φόρτου είναι δύσκολη ως αδύνατη πριν την εκτέλεση της εργασίας [14]. Οπότε το πρόβλημα που ζητάμε να επιλύσουμε είναι ότι δεδομένης μίας εργασίας, για την οποία πολύ δύσκολα μπορούμε να έχουμε γνώση του φόρτου πριν την εκτέλεση και που πρέπει να διεκπεραιωθεί σε ένα υπολογιστικό πλέγμα, να υπάρχει μία εκ των προτέρων εκτίμηση του φόρτου που να μας δίνει ικανοποιητικά αποτελέσματα με βασικότερα τον χρόνο που θα χρειαστεί καθώς και τι πόρους θα δεσμεύσει από το πλέγμα. Ο εκτιμητής φόρτου εργασίας είναι αρμόδιος για την πρόβλεψη των τιμών που απεικονίζει ο Πίνακας 2 μέσω ενός μη γραμμικού προτύπου που χρησιμοποιεί ως εισαγωγή τα συγκεκριμένα χαρακτηριστικά γνωρίσματα μιας δεδομένης εργασίας.

Η βασική ιδέα της λύσης μας συνοψίζεται στην εν μέρει διεκπεραίωση της εργασίας που πρέπει να γίνει στο πλέγμα. Δηλαδή θα γίνεται υλοποίηση ενός ποσοστού της εργασίας με σκοπό να πάρουμε τα καλύτερα αποτελέσματα. Είναι φανερό ότι το ποσοστό της εργασίας που θα εκτελεστεί πρέπει να είναι όσο το δυνατόν μικρότερο και αποδοτικότερο. Αυτό θα το επιτύχουμε χρησιμοποιώντας μεθόδους δειγματοληψίας, αξιοποιώντας την εμπειρία του παρελθόντος και εισάγοντας ένα έμπειρο σύστημα CBR το οποίο θα αυτοματοποιεί όλη την διαδικασία όπως φαίνεται στο παρακάτω σχήμα.



Εικόνα 10 Εκτιμητής Φόρτου Εργασίας

4.3.1 Εύρεση του δείγματος διεκπεραίωσης

Το ποσοστό της εργασίας που θα διεκπεραιωθεί πρέπει να είναι όσο το δυνατόν μικρότερο. Αυτό γιατί η εκτίμηση του φόρτου πρέπει να γίνεται πολύ γρήγορα και σε χρόνο αμελητέο ως προς τον χρόνο που χρειάζεται για να γίνει η δουλειά στο πλέγμα. Είναι βασικό να υπάρχει μία εκ των προτέρων εκτίμηση αλλά αν για παράδειγμα μία εργασία χρειάζεται 5 ώρες για να

γίνει και ο εκτιμητής φόρτου εργασίας κάνει 3 ώρες για να εκτιμήσει ότι θέλει 5 ώρες, τότε η τάξη του χρόνου εκτέλεσης του ΕΦΕ είναι της ίδιας τάξεως με τον χρόνο εκτέλεσης της εργασίας.

Ένα ικανοποιητικό ποσοστό θα ήταν 1-5% αλλά σε καμία περίπτωση δεν θα πρέπει να ξεπερνάει το 10%. Σε αυτή την περίπτωση θα ήταν καλύτερο ο εκτιμητής να απαντήσει αρνητικά ότι δεν μπορεί να εκτιμήσει την εργασία. Για την εύρεση του δείγματος διεκπεραίωσης ο σχεδιαστής του ΕΦΕ θα πρέπει λάβει υπόψη του τα χαρακτηριστικά της εκάστοτε εφαρμογής και το πως μπορεί να εφαρμόσει την διαδικασία δειγματοληψίας. Στο τρίτο κεφάλαιο είδαμε ότι η διαδικασία δειγματοληψίας αποτελείται από επτά βήματα.

1. καθορισμός του πληθυσμού: ο συνολικός πληθυσμός είναι όλο το σύνολο της εργασίας που ένας πελάτης θέλει να εκτελεστεί στο πλέγμα. Είναι δεδομένος από την αρχή. Το πιο πιθανό είναι να έχει την μορφή αρχείων κειμένου, ήχου, εικόνας κτλ.
2. καθορισμός ενός πλαισίου από γεγονότα που θα μετρηθούν: τα γεγονότα προς μέτρηση είναι αυτά που θα μπορέσουν να δώσουν πληροφορίες για τις παραμέτρους που απεικονίζει ο Πίνακας 2. Ενδεχομένως κάποια παράμετρος να μην μπορεί να μετρηθεί άμεσα αλλά έμμεσα. Για παράδειγμα αντί να μετρηθούν τα MFLOP μπορεί να μετρηθεί ο χρόνος εκτέλεσης μιας διεργασίας ο οποίος είναι ισοδύναμος με τα MFLOP.
3. καθορισμός της δειγματοληπτικής μεθόδου: αυτό είναι από τα δυσκολότερα και σημαντικότερα βήματα. Ο σχεδιαστής θα πρέπει να βρει ποιες μέθοδοι δειγματοληψίας μπορούν να εφαρμοστούν σε κάθε εφαρμογή που θα ενσωματωθεί στο πλέγμα ώστε τα αποτελέσματα να είναι ικανά να δώσουν τις απαιτούμενες πληροφορίες για την εκτίμηση του συνολικού χρόνου. Είναι πολύ πιθανό κάποιες μέθοδοι να μην μπορούν να εφαρμοστούν ή η εφαρμογή τους να μην δίνει έγκυρα αποτελέσματα.
4. καθορισμός του μεγέθους του δείγματος: πολύ σημαντικό βήμα και αυτό γιατί από το μέγεθος του δείγματος εξαρτάται ο χρόνος που απαιτείται για να γίνει η εκτίμηση. Ο χρόνος αυτός θέλουμε να είναι όσο το δυνατόν μικρότερος οπότε και το μέγεθος του δείγματος θέλουμε να είναι εξίσου μικρό και αποδοτικό.
5. εφαρμογή του σχεδίου δειγματοληψίας: εύρεση των δειγμάτων που θα εκτελεστούν
6. δειγματοληψία και συλλογή στοιχείων: εκτέλεση των δειγμάτων και καταγραφή των παραμέτρων που χαρακτηρίζουν τον φόρτο εργασίας.
7. επανεξέταση της διαδικασίας: σε περίπτωση αποτυχίας της εκτίμησης καλό είναι να εξετάζονται οι λόγοι για τους οποίους η μέθοδος δειγματοληψίας απέτυχε στην

εύρεση του δείγματος που θα ήταν αντιπροσωπευτικό για όλη την εργασία. Η επανεξέταση μπορεί να γίνει αφού εκτελεστεί η εργασία και συγκριθούν ο εκτιμώμενος φόρτος με τον πραγματικό.

4.3.2 Εμπειρία παρελθόντος

Η εμπειρία παρελθόντος είναι πληροφορία που μπορεί να ανακτηθεί από προηγούμενες καταστάσεις. Για παράδειγμα ένα σούπερ-μάρκετ μπορεί από την καταγραφή της αποθήκης να γνωρίζει ποια αγαθά έχουν περισσότερη ζήτηση στις διάφορες εποχές του χρόνου. Έτσι θα μπορεί να προβλέπει τις ποσότητες των παραγγελιών για να μπορεί να καλύψει τις ανάγκες των πελατών. Αυτός ο μηχανισμός αυξάνει την ποιότητα της υπηρεσίας του σούπερ-μάρκετ προς τους πελάτες.

Η καταγραφή παλιότερων προβλημάτων και η επίλυση τους μπορεί να βοηθήσει στην επίλυση νέων προβλημάτων. Όταν παρουσιαστεί ένα νέο πρόβλημα ο αρμόδιος που είναι υπεύθυνος για την επίλυση του μπορεί να το συγκρίνει με παλιότερες περιπτώσεις και να βρει κάποιες που μοιάζουν. Στην συνέχεια κοιτάει με ποιο τρόπο είχε επιλυθεί τότε το πρόβλημα και να εφαρμόσει τις ίδιες ή παρόμοιες τεχνικές και στο υπάρχον πρόβλημα. Η εμπειρία παρελθόντος βελτιώνει πάρα πολύ την λήψη αποφάσεων του εμπειρογνώμονα για την επίλυση προβλημάτων.

Στην περίπτωση του ΕΦΕ το πρόβλημα που έχουμε να αντιμετωπίσουμε είναι η κατάλληλη εύρεση του δείγματος που θα εκτελεστεί για να δώσει ικανοποιητικά αποτελέσματα. Για να επιτευχθεί αυτό πρέπει να εφαρμοστεί η πιο αποδοτική μέθοδος δειγματοληψίας με το πιο μικρό ποσοστό. Αν έχει κρατηθεί ένα ιστορικό προηγούμενων εργασιών καθώς και η διαδικασία δειγματοληψίας που είχε επιτυχημένη εκτίμηση, ο ΕΦΕ μπορεί να βρει παλιότερες εργασίες που μοιάζουν με την καινούργια και να εφαρμόσει την ίδια διαδικασία δειγματοληψίας. Όμως επειδή ο ΕΦΕ είναι ένα πρόγραμμα και όχι ένας άνθρωπος θα πρέπει να του ενσωματωθεί ένα έμπειρο σύστημα που θα είναι υπεύθυνο για την λήψη των αποφάσεων. Το κατάλληλο έμπειρο σύστημα που μπορεί να αξιοποιήσει την εμπειρία του παρελθόντος είναι το Case-Based Reasoning (CBR) όπως είδαμε στο τρίτο κεφάλαιο.

4.3.3 Έμπειρο σύστημα ‘Συλλογισμού Βασισμένο σε Περιπτώσεις (Case-Based Reasoning)

Αφού το CBR είναι μέρος του ΕΦΕ ο σχεδιαστής θα πρέπει να μελετήσει καλά τον τρόπο που σχεδιάζεται ένα τέτοιο έμπειρο σύστημα και το πως μπορεί να το προσαρμόσει στην εκάστοτε εφαρμογή που θα ενσωματωθεί στο πλέγμα. Στο κεφάλαιο τρία είδαμε τις βασικότερες λειτουργίες του CBR.

Κατ' αρχάς θα πρέπει να μελετήσει την εφαρμογή και να βρει πως μπορεί να αναπαραστήσει την εκάστοτε περίπτωση ώστε να αποθηκεύεται ως παλιότερη εμπειρία. Το σχήμα πρέπει να είναι όσο το δυνατόν πιο αντιπροσωπευτικό της περίπτωσης. Για παράδειγμα ο όγκος μιας εργασίας μπορεί να αναπαρασταθεί είτε σε συνολικό μέγεθος σε bytes είτε στο πλήθος των αρχείων που την αποτελούν. Η επιλογή μεταξύ αυτών των δύο αναπαραστάσεων έχει να κάνει με το τι θεωρείται πιο σημαντικό στην εφαρμογή. Θα μπορούσαν να καταγράφονται και οι δύο αν ο σχεδιαστής κρίνει ότι η γνώση τους χρειάζεται. Η αναπαράσταση μπορεί να γίνεται περιγραφικά ή ποσοτικά. Περιγραφική αναπαράσταση είναι ότι ο καιρός σήμερα είναι ζεστός ή κρύος ενώ ποσοτική είναι ότι σήμερα έχει 38 βαθμούς θερμοκρασία. Σε γενικές γραμμές ένα σχήμα που θα αναπαριστά την γνώση θα περιέχει τα χαρακτηριστικά της εφαρμογής με τα οποία θα συγκρίνονται και θα ταιριάζουν οι εργασίες, τον εκτιμητέο χρόνο διεκπεραίωσης της εργασίας, τον πραγματικό χρόνο διεκπεραίωσης της εργασίας, τον χρόνο εκτέλεσης του ΕΦΕ και διάφορες συμπληρωματικές πληροφορίες όπως το SLA που συμφωνήθηκε σε αυτήν την εργασία.

Στην συνέχεια θα ορίσει ποιες περιπτώσεις θεωρούνται παρόμοιες και να βρει ποιες τεχνικές ανάκτησης περίπτωσης (Ανάκτηση πλησίον-γειτόνων, Επικυρωμένη ανάκτηση κτλ) μπορούν να χρησιμοποιηθούν. Όπως αναφέραμε και στο προηγούμενο παράδειγμα δυο εργασίες μπορεί να έχουν τον ίδιο φόρτο εργασίας αν μοιάζουν στον συνολικό όγκο σε bytes, αν έχουν περίπου τα ίδια αρχεία ή να ταιριάζουν και στα δύο. Στην περίπτωση που μια εργασία δεν ταιριάζει με καμία προηγούμενη περίπτωση είναι στην κρίση του σχεδιαστή να φτιάξει τον ΕΦΕ έτσι ώστε να κάνει μία προσαρμογή περίπτωσης, να κάνει μία εκτίμηση που βασίζεται σε γενικούς κανόνες και να εφαρμόσει την δειγματοληψία που θεωρεί καλύτερη ή να διακόψει την εκτίμηση και να ενημερώσει τον διαχειριστή του πλέγματος.

Τέλος πολύ σημαντικό είναι η εκμάθηση της μηχανής. Αφού διεκπεραιωθεί μία εργασία γνωρίζουμε πλέον τον πραγματικό χρόνο εκτέλεσης. Αν ο χρόνος εκτίμησης έχει απόκλιση μικρότερη ενός ποσοστού επιτυχίας από τον πραγματικό χρόνο εκτέλεσης τότε η εκτίμηση θεωρείται επιτυχής. Το ποσοστό απόκλισης ορίζεται από τις ανάγκες για το πόση ακρίβεια απαιτείται. Επίσης επιτυχημένη εκτίμηση μπορεί να θεωρηθεί όταν η εκτέλεση της εργασίας ικανοποίησε το SLA του παρόχου με τον πελάτη. Στην περίπτωση αποτυχίας ο διαχειριστής του πλέγματος θα αναλύσει τους λόγους της αποτυχίας και ενδεχομένως να βρει πως θα έπρεπε να είχε γίνει η εκτίμηση και με ποιο τρόπο δειγματοληψίας. Έτσι θα μπορεί να καταγραφεί η σωστή εκτίμηση της περίπτωσης αυτής και να αναβαθμιστεί η γνώση της μηχανής ακόμα και στις λανθασμένες εκτιμήσεις. Εννοείται ότι οι επιτυχημένες εκτιμήσεις καταγράφονται στην μηχανή. Ακόμη η διαδικασία αναβάθμισης θα μπορούσε να είναι και αυτή μια αυτοματοποιημένη διαδικασία.

5

Εκτιμητής Φόρτου Εργασίας για την δημιουργία τρισδιάστατων γραφικών (3D Rendering)

Η υλοποίηση μιας εμπορικής εφαρμογής σε ένα υπολογιστικό πλέγμα εισάγει νέες προκλήσεις για την διατήρηση του QoS από το γεγονός ότι η διαπραγμάτευση για QoS μεταξύ του χρήστη και του φορέα παροχής υπηρεσιών πρέπει αυστηρά να ικανοποιηθεί. Μια ενδιαφέρουσα εμπορική εφαρμογή με έναν ευρύ αντίκτυπο σε ποικίλους τομείς που μπορεί να ωφεληθεί από τις υπολογιστικές τεχνολογίες πλέγματος είναι το 3D rendering [3]. Ο λόγος που γίνεται αυτό είναι γιατί το rendering έχει πολύ μεγάλες απαιτήσεις σε υπολογιστική ισχύ, μπορεί να είναι εξαιρετικά χρονοβόρο ή η αγορά ενός πακέτου renderer από έναν χρήστη δεν είναι εφικτή ή αναγκαία. Προκειμένου να εφαρμοστεί το rendering στο πλέγμα πρέπει να αναπτυχθούν οι κατάλληλοι μηχανισμοί κατανομής και σχεδιασμού των πόρων έτσι ώστε να καλύπτονται οι απαιτήσεις του QoS. Ο αποδοτικός σχεδιασμός απαιτεί διαμόρφωση και πρόβλεψη του φόρτου εργασίας. Για να μπορέσουμε να μελετήσουμε τη σχεδίαση ενός ΕΦΕ για το rendering θα πρέπει να γνωρίζουμε πως λειτουργεί το rendering καθώς και ιδιαίτερα χαρακτηριστικά του renderer που θα χρησιμοποιήσουμε. Στην περίπτωση μας η μελέτη έγινε με τον air της sittedgraphics.

5.1 Δημιουργία τρισδιάστατων γραφικών (3D Rendering)

Το rendering είναι η διαδικασία της δημιουργίας μιας εικόνας από ένα μοντέλο με τη βοήθεια των προγραμμάτων ηλεκτρονικών υπολογιστών [23]. Το μοντέλο αυτό είναι μια περιγραφή των τρισδιάστατων αντικειμένων σε μια αυστηρά ορισμένη γλώσσα ή δομή δεδομένων. Η περιγραφή περιέχει γεωμετρία, οπτική γωνία, υφή, φωτισμό, σκίαση και διάφορες πληροφορίες. Το Rendering έχει χρήσεις στην αρχιτεκτονική, στα video games, στους προσομοιωτές, στις ταινίες ή στην τηλεόραση, στα ειδικά εφέ και στην οπτικοποίηση σχεδιασμού. Σε κάθε μία περίπτωση υπάρχουν διαφορετικές τεχνικές και διαφορετικά χαρακτηριστικά. Ως προϊόν οι renderers είναι διαθέσιμοι σε μια ευρεία ποικιλία. Μερικοί είναι ενταγμένοι σε μεγαλύτερα πακέτα σχεδιασμού, άλλοι ως αυτόνομα προγράμματα και άλλοι ως προγράμματα ανοιχτού κώδικα. Ο renderer είναι ένα μελετημένο πρόγραμμα σχετικό με φως, φυσική, οπτική αντίληψη, μαθηματικά και ανάπτυξης λογισμικού. Στην περίπτωση των τρισδιάστατων γραφικών το rendering μπορεί να γίνει αργά είτε σε πραγματικό χρόνο. Για παράδειγμα στα video games θέλουμε επεξεργασία σε πραγματικό χρόνο ενώ σε ταινίες δεν είναι απαραίτητη αυτή η απαίτηση.

5.1.1 Χρήση

Το rendering χρησιμοποιείται είτε για την δημιουργία εικόνων που θα δει ένας θεατής είτε για την δημιουργία ταινίας. Στην δεύτερη περίπτωση κάθε εικόνα είναι ένα καρέ και για αυτό χρειάζεται κάποιο πρόγραμμα που θα δημιουργήσει την κίνηση των καρέ.

5.1.2 Χαρακτηριστικά

Μια εικόνα μπορεί να γίνει αντιληπτή όταν υπάρχουν κάποια ορατά χαρακτηριστικά. Η έρευνα και η ανάπτυξη του rendering έχει βρει διάφορους τρόπους για την αποτελεσματική προσομοίωση αυτών των χαρακτηριστικών. Ορισμένοι τρόποι συνδέονται άμεσα με αλγόριθμους και τεχνικές ενώ άλλοι έχουν παραχθεί από κοινού.

- shading (σκίαση) – η μεταβολή του χρώματος και της φωτεινότητας μιας επιφάνειας ανάλογα με το φωτισμό
- texture-mapping — μέθοδος εφαρμογής λεπτομερειών σε επιφάνειες
- bump-mapping — μέθοδος προσομοίωσης προεξοχών μικρής κλίμακας στις επιφάνειες
- fogging/participating medium - ο τρόπος που το φως σκοτεινιάζει όταν διέρχεται από ασαφή ατμόσφαιρα ή αέρα
- shadows (σκιές) - Η επίδραση του παρεμποδιζόμενου φωτός

- soft shadows – μεταβολή σκοτεινότητας από μερικώς παρεμποδιζόμενη πηγή φωτός
- reflection — αντανάκλαση
- transparency or opacity (διαφάνεια ή αδιαφάνεια) - απότομη μετάδοση του φωτός μέσα από στερεά αντικείμενα
- translucency (διαφάνεια) - άκρως διάσπαρτη μετάδοση του φωτός μέσα από στερεά αντικείμενα
- refraction (διάθλαση)- κάμψη του φωτός που σχετίζεται με την διαφάνεια
- diffraction - η εξάπλωση και η παρέμβαση του φωτός που διέρχεται από ένα αντικείμενο
- indirect illumination (έμμεσος φωτισμός) – φωτισμός επιφανειών από το φως που αντανακλάται από άλλες επιφάνειες παρά από μια πηγή φωτός απευθείας
- caustics (μορφή έμμεσου φωτισμού) - αντανάκλαση του φωτός από ένα γυαλιστερό αντικείμενο ή εστίαση του φωτός μέσα από ένα διαφανές αντικείμενο
- depth of field (βάθος πεδίου) – αντικείμενα που εμφανίζονται θολά ή εκτός εστίασης κατά πολύ μπροστά ή πίσω από το αντικείμενο της εστίασης
- motion blur – αντικείμενα που εμφανίζονται θολά λόγω της υψηλής ταχύτητας κίνησης ή της κίνησης της κάμερας
- photorealistic morphing (φωτορεαλιστική μορφοποίηση) – απόδοση αληθοφάνειας
- non-photorealistic rendering (μη φωτορεαλιστική μορφοποίηση) δημιουργία των σκηνών σε ένα καλλιτεχνικό ύφος, το οποίο προορίζεται να μοιάζει με τη ζωγραφική.

5.1.3 Τεχνικές

Πολλοί αλγόριθμοι έχουν διερευνηθεί και το λογισμικό που χρησιμοποιείται για την απόδοση μπορεί να χρησιμοποιήσει μια σειρά από διαφορετικές τεχνικές για να δημιουργηθεί μια τελική εικόνα.

Ο εντοπισμός κάθε ακτίνας φωτός σε μια σκηνή θα ήταν ανεφάρμοστος και χρονοβόρος, ακόμα και αν γινόταν μερικώς, αν δεν εφαρμοστεί μία έξυπνη δειγματοληψία. Ωστόσο τέσσερις οικογένειες τεχνικών μοντελοποίησης του φωτός χρησιμοποιούνται.

- Rasterization: θεωρεί τα αντικείμενα της σκηνής και τα σχεδιάζει για να σχηματίσουν μια εικόνα χωρίς καμιά διευκόλυνση για την παραγωγή μιας προοπτικής.
- Ray casting: θεωρεί τη σκηνή όπως παρατηρείται από μια συγκεκριμένη θέα, δημιουργώντας την εικόνα, βασιζόμενη σε απλή γεωμετρία και σε πολύ βασικούς νόμους της οπτικής καθώς και χρησιμοποιώντας τεχνικές Monte Carlo.

- Radiosity : χρησιμοποιεί μαθηματικά πεπερασμένων στοιχείων για την προσομοίωση της διάχυτης διάδοσης του φωτός στις επιφάνειες.
- Ray tracing : είναι παρόμοια με την ray casting αλλά απασχολεί περισσότερο προηγμένες προσομοιώσεις οπτικής και συνήθως χρησιμοποιεί τεχνικές Monte Carlo για πιο ρεαλιστικά αποτελέσματα.

Η εφαρμογή ενός ρεαλιστικού renderer πάντα έχει βασικά στοιχεία της φυσικής προσομοίωσης ή εξομοίωσης, δηλαδή υπολογισμούς που μοιάζουν με μία πραγματική φυσική διαδικασία. Ο όρος " physically-based " υποδηλώνει τη χρήση των φυσικών μοντέλων και προσεγγίσεων που είναι πιο γενικά και ευρέως αποδεκτά εκτός της κοινότητας του rendering. Ένα ιδιαίτερο σύνολο από σχετικές τεχνικές έγιναν σταδιακά αποδεκτές στην επιστήμη του rendering. Τα μαθηματικά που χρησιμοποιούνται είναι: γραμμική άλγεβρα, λογισμός, αριθμητικά μαθηματικά, η επεξεργασία σήματος και monte carlo.

5.1.4 Η εξίσωση του Rendering

Αυτό είναι το κλειδί της θεωρητικής έννοιας του rendering. Χρησιμοποιεί ως η πιο τυπική έκφραση και όλοι οι αλγόριθμοι μπορούν να θεωρηθούν λύσεις των διαφόρων μορφών αυτής της εξίσωσης.

Εξίσωση 1 Γενική εξίσωση του rendering

$$L_o(x, \vec{w}) = L_e(x, \vec{w}) + \int_{\Omega} f_r(x, \vec{w}', \vec{w}) L_i(x, \vec{w}') (\vec{w}' \cdot \vec{n}) d\vec{w}'$$

Σε μια συγκεκριμένη θέση και κατεύθυνση το εξερχόμενο φως (L_{ie}) είναι το άθροισμα του εκπεμπόμενου φωτός (L_e) και του ανακλώμενου. Το φως που αντανακλάται είναι το άθροισμα του εισερχόμενου φωτός (L_i) από όλες τις κατευθύνσεις πολλαπλασιαζόμενο από την ανακλώμενη επιφάνεια και της εισερχόμενης γωνίας.

5.1.5 Η αμφίδρομη Reflectance Distribution Function

Η αμφίδρομη Reflectance Distribution Function (BRDF) εκφράζει ένα απλό μοντέλο αλληλεπίδρασης του φωτός με την επιφάνεια ως εξής.

Εξίσωση 2 Η αμφίδρομη Reflectance Distribution Function (BRDF)

$$f_r(x, \vec{w}', \vec{w}) = \frac{dL_r(x, \vec{w})}{L_i(x, \vec{w}') (\vec{w}' \cdot \vec{n}) d\vec{w}'}$$

5.1.6 Οπτική αντίληψη

Παρόλο που δέχεται λιγότερη προσοχή η κατανόηση της ανθρώπινης οπτικής αντίληψης είναι πολύτιμη για το rendering. Αυτό οφείλεται κυρίως στο γεγονός ότι η απεικόνιση της εικόνας και η ανθρώπινη αντίληψη έχουν περιορισμένο εύρος. Ένας renderer μπορεί να εξομοιώσει ένα σχεδόν άπειρο φάσμα του φωτός και του χρώματος αλλά οι υπάρχουσες οθόνες δεν μπορούν. Για αυτό κάποιο φάσμα απορρίπτεται ή συμπιέζεται. Επίσης η ανθρώπινη αντίληψη έχει και αυτή τα όριά της και δεν χρειάζεται να δοθεί υψηλού επιπέδου εικόνα για να δημιουργηθεί ρεαλισμός. Αυτό μπορεί να βοηθήσει στην επίλυση του προβλήματος της τοποθέτησης εικόνων σε οθόνες και μικρές περικοπές θα μπορούσαν να χρησιμοποιηθούν για το rendering δεδομένου ότι ορισμένες λεπτομέρειες δεν θα είναι αισθητές.

5.1.7 Renderman Interface

Το renderman interface είναι μία πρότυπη διεπαφή μεταξύ των modeling programs και των rendering προγραμμάτων (renderer) που έχουν την ικανότητα να δημιουργούν ποιοτικές εικόνες. Ένα rendering πρόγραμμα που περιέχει την renderman διεπαφή διαφέρει από οποιοδήποτε άλλο πρότυπο γραφικής αναπαράστασης στα εξής:

- Πρέπει να εξομοιώνει πολλές από τις ιδιότητες μίας αληθινής κάμερας και όχι μόνο την θέση της ή την κατεύθυνση της θέας. Η υψηλή ποιότητα συνεπάγεται ότι η εξομοίωση δεν εισάγει τεχνητά αντικείμενα από την υπολογιστική διαδικασία.
- Πρέπει να δέχεται κυρτά γεωμετρικά αντικείμενα έτσι ώστε όχι μόνο η γεωμετρία να απεικονιστεί ακριβώς αλλά και τα βασικά σχήματα να είναι αρκετά πλούσια και να περιλαμβάνουν την ποικιλομορφία των φυσικών αντικειμένων.
- Πρέπει να είναι ικανό να εξομοιώνει τις οπτικές ιδιότητες των διαφορετικών υλικών και των πηγών φωτός. Αυτό περιλαμβάνει μοντέλα σκίασης επιφανειών που περιγράφουν πώς το φως αλληλεπιδρά με μια επιφάνεια φτιαγμένη από δεδομένο υλικό, μοντέλα σκίασης του όγκου που περιγράφουν πώς το φως είναι διεσπαρμένο δεδομένου ότι διαβαίνει μια περιοχή στο χώρο και τα μοντέλα πηγών φωτός που περιγράφουν το χρώμα και την ένταση του φωτός που εκπέμπονται στις διαφορετικές κατευθύνσεις. Η επίτευξη μεγαλύτερου ρεαλισμού συχνά απαιτεί ότι οι ιδιότητες επιφάνειας ενός αντικειμένου ποικίλλουν.

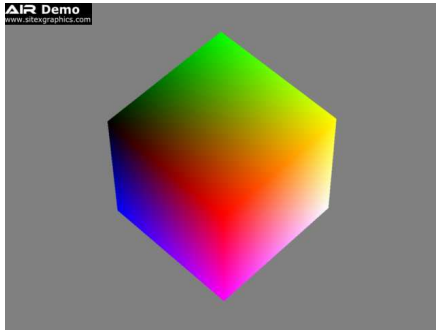
5.1.8 Αρχεία RIB

Ένα συνθετικός κόσμος μπορεί να κωδικοποιηθεί χρησιμοποιώντας το RenderMan Interface Bytestream (RIB) αρχείο [4]. Ο σκοπός της rib διάταξης είναι να προβάλλει μία γενική δομή

που περιγράφει ένα συνθετικό κόσμο. Η δομή αυτή έχει την δυνατότητα να περιγράφει γεωμετρικά σχήματα, όπως η σφαίρα, ο κύκλος, ο κώνος και άλλα και να επιτρέπει τον μετασχηματισμό τους στο χώρο. Έτσι ένα τρισδιάστατο αντικείμενο κατασκευάζεται από έναν συνδυασμό γεωμετρικών σχημάτων και μετασχηματισμούς αυτών. Ακόμα η διάταξη rib περιέχει επιπρόσθετες πληροφορίες για την επιφάνεια του αντικειμένου, τους φωτισμούς κτλ. Μία σκηνή περιγράφεται από την ανάλυση της εικόνας, την ποσότητα και το είδος των φωτισμών και από τους αλγόριθμους με τους οποίους γίνεται το rendering. Αυτοί είναι οι γενικοί περιγραφητές. Ο δεύτερος τύπος αφορά τους περιγραφητές σχετικούς με την συνθετική γεωμετρία και τα πρωτόγονα χαρακτηριστικά όπως η πολυπλοκότητα του αντικειμένου, η σύσταση της επιφάνειάς του και το υλικό του. Αυτοί είναι οι περιγραφητές αντικειμένου. Ένα αρχείο rib έχει την μορφή:

```
##RenderMan RIB-Structure 1.1
##Description Rotating color cube
##CreationDate 12-99
##Author Scott Iverson
Projection "perspective" "fov" 12
PixelSamples 1 1
PixelFilter "box" 1 1
Imager "background" "bgcolor" [.5 .5 .5]
Translate 0 0 10
MotionBegin [0 1]
  Rotate 15 0 1 0
  Rotate 375 0 1 0
MotionEnd
WorldBegin
  Surface "constant"
  ShadingInterpolation "smooth"
  ShadingRate 1000000
  Rotate 54 1 0 -1
  Translate -.5 -.5 -.5
  PointsPolygons [4 4 4 4 4 4]
    [2 1 5 6 1 0 4 5 0 3 7 4 7 3 2 6 1 2 3 0 6 5 4 7]
    "P" [0 1 1 1 1 1 1 1 0 0 1 0 0 0 1 1 0 1 1 0 0 0 0 0]
    "Cs" [0 1 1 1 1 1 1 1 0 0 1 0 0 0 1 1 0 1 1 0 0 0 0 0]
WorldEnd
```

Το αρχείο αυτό επεξεργασμένο από τον κατάλληλο renderer δίνει την παρακάτω εικόνα



Εικόνα 11 Αποτέλεσμα δημιουργίας τρισδιάστατης εικόνας από renderer

5.1.9 AIR

Το AIR είναι ένας προηγμένος renderer τρισδιάστατης γραφικής αναπαράστασης με μια μοναδική αρχιτεκτονική και εκτενή χαρακτηριστικά γνωρίσματα για τη γρήγορη παραγωγή υψηλής ποιότητας εικόνων. Το AIR είναι ένας υβριδικός renderer που συνδυάζει τα πλεονεκτήματα του scanline – rendering, του γρήγορου rendering των σύνθετων σκηνών, την θαμπάδα της κίνησης, την ευελιξία της ανίχνευσης ακτινών για τις ακριβείς αντανακλάσεις, τις μαλακές σκιές και το σφαιρικό φωτισμό.

5.2 Εκτίμηση φόρτου εργασίας στην δημιουργία

τρισδιάστατων γραφικών

Το rendering είναι μία πολύπλοκη διαδικασία. Ο χρόνος δημιουργίας μίας εικόνας από έναν renderer εξαρτάται από την πολυπλοκότητά της. Η πολυπλοκότητα της εικόνας βασίζεται στα αντικείμενα που περιέχει, στο υλικό τους, στην υφή τους, στους φωτισμούς και στις σκιάσεις, στην ανάλυση κτλ. Με ένα parsing στο rib αρχείο μπορούμε να βρούμε αυτά τα χαρακτηριστικά [3] αλλά το rendering είναι μία μη γραμμική διαδικασία με αποτέλεσμα να μην έχουμε ακριβή γνώση πριν από την εκτέλεση. Μια οποιαδήποτε μικρή αλλαγή σε ένα χαρακτηριστικό μπορεί να αλλάξει πάρα πολύ τον φόρτο εργασίας. Για παράδειγμα δεν υπάρχει κάποια γραμμική σχέση που να εγγυάται ότι ο διπλασιασμός των αντικειμένων θα διπλασιάσει και τον χρόνο εκτέλεσης. Για αυτό και καταφεύγουμε στην λύση που προτείναμε. Δηλαδή να εκτελέσουμε ένα μέρος της συνολικής εργασίας για να πάρουμε τα επιθυμητά αποτελέσματα. Σκοπός μας είναι λοιπόν να σχεδιάσουμε έναν εκτιμητή φόρτου εργασίας για την εφαρμογή του rendering σε περιβάλλον υπολογιστικού πλέγματος. Οι προϋποθέσεις που θέλουμε να πληροί ο εκτιμητής του φόρτου εργασίας είναι οι εξής:

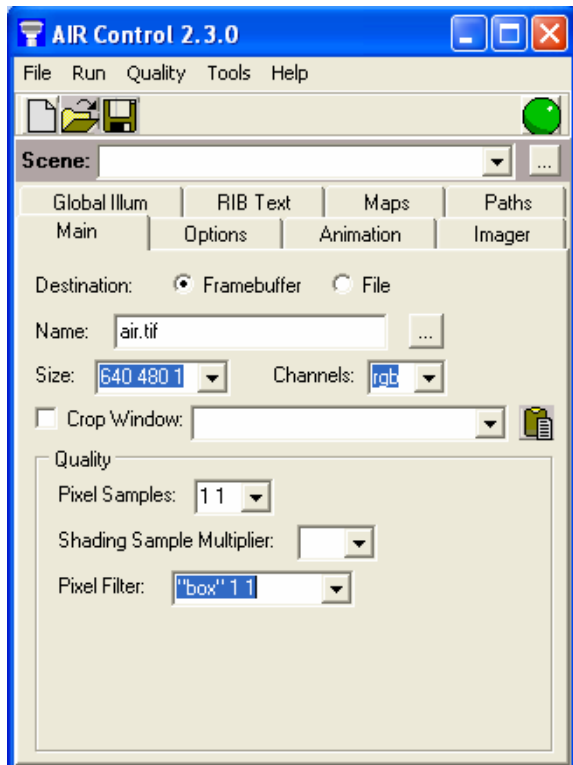
- Εύρεση του χρόνου διεκπεραίωσης μιας εργασίας rendering σε δεδομένη υπολογιστική ισχύ: αυτή είναι και η σημαντικότερη πληροφορία για την διατήρηση του QoS και την τήρηση της συμφωνίας του SLA που έχει συμφωνηθεί μεταξύ του

παρόχου της υπηρεσίας και του πελάτη. Επίσης η γνώση αυτή μπορεί να βοηθήσει στις συμφωνίες μελλοντικών εργασιών αφού ο διαχειριστής του πλέγματος θα ξέρει τότε θα ελευθερωθεί το πλέγμα από την συγκεκριμένη εργασία.

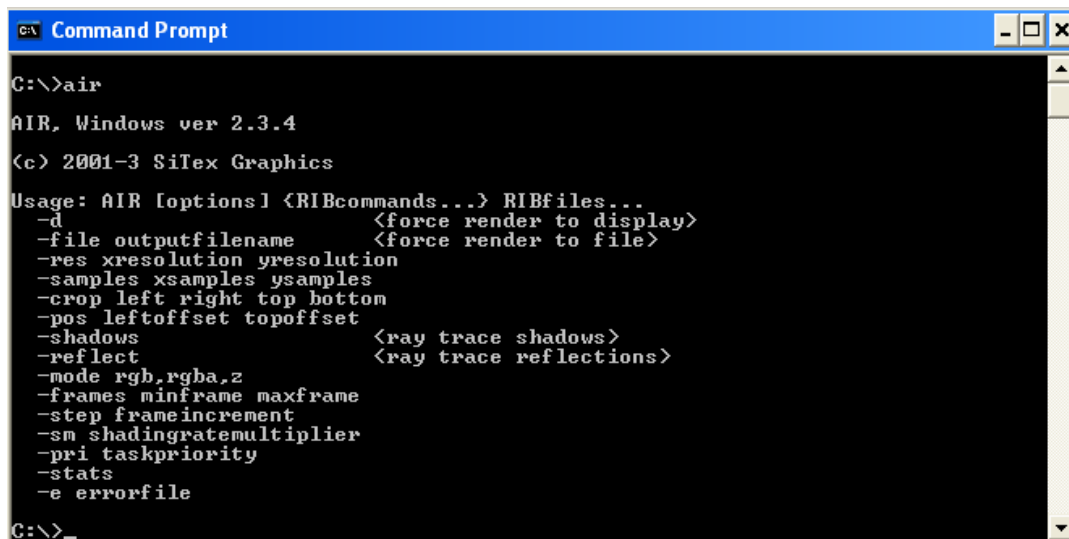
- Απαιτήσεις υπολογιστικής ισχύος για διεκπεραίωση εργασίας σε περιορισμένο χρόνο: αυτή η πληροφορία είναι χρήσιμη στην περίπτωση που έχουμε έναν ‘βιαστικό’ πελάτη και απαιτεί την διεκπεραίωση της εργασίας σε χρόνο που το πλέγμα δεν μπορεί να αντεπεξέλθει. Έτσι ο πάροχος της υπηρεσίας μπορεί να ξέρει πόση επιπλέον υπολογιστική ισχύ θα χρειαζόταν για να καλύψει αυτή την απαίτηση και να την βρει σε κάποιον εξωτερικό συνεργάτη.
- Απαιτήσεις σε αποθηκευτικό χώρο του πλέγματος: οι εικόνες γενικά καταλαμβάνουν αρκετό χώρο για την αποθήκευσή τους ειδικά αν ο πάροχος θέλει να κρατάει και κάποιο αντίγραφο για κάποιο χρονικό διάστημα. Επίσης η γνώση αυτή βοηθάει και στις απαιτήσεις του δικτύου (ταχύτητα γραμμής) αφού όλες οι υπηρεσίες ιστού γίνονται μέσω του Internet.
- Απαιτήσεις για την χρησιμοποίηση μνήμης κατά την διάρκεια της επεξεργασίας. Χρήσιμη πληροφορία για την καλύτερη κατανομή των πόρων.
- Ο χρόνος που θα χρειαστεί για την εκτίμηση θέλουμε να είναι όσο το δυνατόν μικρότερος: αυτό είναι λογικό αφού θέλουμε να έχουμε μια εκτίμηση της κατάστασης γρήγορα και σε χρόνο αμελητέο ως προς τον συνολικό χρόνο εκτέλεσης της εργασίας.
- Διαχωρισμό των αρχείων με βάση τον φόρτο εργασίας τους: και αυτή είναι μια χρήσιμη πληροφορία που μπορεί να βοηθήσει στην καλύτερη κατανομή των πόρων και τα χρονοβόρα αρχεία να εκτελούνται στους μεγαλύτερους υπολογιστές.

5.3 Εκτιμητής Φόρτου Εργασίας για τον AIR

Για να μπορέσουμε να σχεδιάσουμε ένα πρόγραμμα που θα κάνει εκτίμηση του φόρτου εργασίας για τον air θα πρέπει να δούμε πως λειτουργεί το πρόγραμμα, τι εισόδους δέχεται και τι έξοδο δίνει. Ο air μπορεί να λειτουργεί ως αυτόνομο πρόγραμμα με ολοκληρωμένο περιβάλλον (Gui) ή σε command prompt όπως φαίνεται παρακάτω στις παρακάτω εικόνες.



Εικόνα 12 Στιγμιότυπο από το Gui του Air

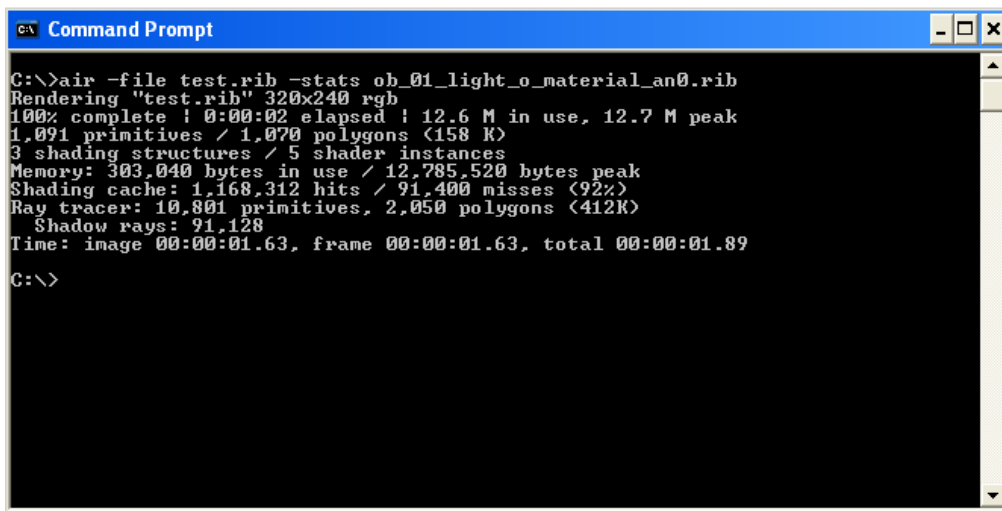


Εικόνα 13 Στιγμιότυπο του Air από το command prompt

Η δυνατότητα του air να τρέχει σε command prompt μας δίνει την ευελιξία να τον τρέχουμε ως διεργασία στο παρασκήνιο μέσα από δικά μας προγράμματα. Η Java έχει την δυνατότητα να εκτελεί εξωτερικά προγράμματα ως διεργασίες υποστηρίζοντας όλα τα ρεύματα εισόδου και εξόδου του προγράμματος.

Ο air δέχεται ως είσοδο αρχεία σε rib format καθώς και command line switches. Είναι προφανές ότι η είσοδος θα είναι τα αρχεία προς επεξεργασία που θα θέλει κάποιος πελάτης. Τα πιο σημαντικά command line switches από αυτά που φαίνονται στην εικόνα και που χρησιμοποιήσαμε είναι τα εξής:

- -file outputFile: με την παράμετρο αυτή καθορίζεται το όνομα του παραγόμενου αρχείου εικόνας.
- -res Xresolution Yresolution: με την παράμετρο αυτή καθορίζεται το μέγεθος της ανάλυσης της εικόνας.
- -stats: η παράμετρος αυτή δίνει στο τέλος στο ρεύμα εξόδου διάφορα στατιστικά στοιχεία που αφορούν την επεξεργασία. (χρόνο εκτέλεσης, primitives, polygons, shaders κτλ.) όπως φαίνεται στην παρακάτω εικόνα. Τα στατιστικά αυτά στοιχεία είναι πολύ χρήσιμα στον διαχωρισμό των δειγμάτων ανά κατηγορίες πολυπλοκότητας. Γενικά τα αρχεία που είναι της ίδιας πολυπλοκότητας και έχουν την ίδια συμπεριφορά κατά την εκτέλεσή τους εμφανίζουν τα ίδια στατιστικά στο τέλος της εργασίας.



```

C:\>air -file test.rib -stats ob_01_light_o_material_an0.rib
Rendering "test.rib" 320x240 rgb
100% complete | 0:00:02 elapsed | 12.6 M in use, 12.7 M peak
1,091 primitives / 1,070 polygons <158 K>
3 shading structures / 5 shader instances
Memory: 303,040 bytes in use / 12,785,520 bytes peak
Shading cache: 1,168,312 hits / 91,400 misses <92%>
Ray tracer: 10,801 primitives, 2,050 polygons <412K>
Shadow rays: 91,128
Time: image 00:00:01.63, frame 00:00:01.63, total 00:00:01.89
C:\>

```

Εικόνα 14 Αποτελέσματα από την εκτέλεση του Air

Η έξοδος του air είναι η παραγόμενη εικόνα είτε στο AirShow είτε σε αρχείο εικόνας (tif, jpg κτλ.). Επίσης ένα χαρακτηριστικό του είναι ότι δίνει πληροφορίες στο ρεύμα εξόδου για την εξέλιξη της εργασίας (ποσοστό ολοκλήρωσης) καθώς και τις πληροφορίες στο τέλος της εργασίας που παίρνουμε με την παράμετρο -stats όπως είδαμε παραπάνω. Ακόμη στο ρεύμα εξόδου λαθών δίνει πληροφορίες για οποιοδήποτε λάθος συνέβη κατά την επεξεργασία. Τέτοια λάθη μπορεί να είναι μία λάθος εντολή, ένα εξωτερικό αρχείο που λείπει ή προειδοποιήσεις για αποδοκιμασμένες μεθόδους(deprecated).

5.3.1 Δειγματοληψία

Όπως είπαμε στο τέταρτο κεφάλαιο η βασική ιδέα του ΕΦΕ βασίζεται στην εν μέρει διεκπεραίωση της εργασίας. Για αυτό θα αναλύσουμε πως μπορούμε να εφαρμόσουμε τη διαδικασία της δειγματοληψίας και ποιες μέθοδοι δειγματοληψίας μπορούν να εφαρμοστούν στην περίπτωση μας. Σε γενικές γραμμές και πρακτικά έχουμε δεχτεί ότι σε μια εργασία

rendering υπάρχουν πολλά όμοια gib αρχεία που έχουν την ίδια συμπεριφορά κατά την επεξεργασία τους. Σκοπός μας λοιπόν είναι να βρούμε τα λιγότερα αρχεία που θα προεπεξεργαστούμε για να πάρουμε ικανοποιητικά αποτελέσματα.

5.3.1.1 Οι φάσεις της δειγματοληψίας

Είδαμε ότι οι φάσεις της δειγματοληψίας είναι εφτά. Θα δούμε πως μπορούν να εφαρμοστούν στην περίπτωσή μας.

1. καθορισμός του πληθυσμού: ο πληθυσμός είναι όλα τα αρχεία gib της εργασίας που κάποιος χρήστης θέλει να επεξεργαστούν στο πλέγμα. Είναι πλήρως καθορισμένος από την αρχή.
2. καθορισμός του πλαισίου γεγονότων: τα ενδιαφερόμενα γεγονότα προς μέτρηση είναι ο χρόνος διεκπεραίωσης του δείγματος, η απαιτούμενη μνήμη που χρειάστηκε κατά την επεξεργασία και ο απαιτούμενος αποθηκευτικός χώρος. Δηλαδή μετράμε την πρώτη, την τρίτη και την τέταρτη παράμετρο από αυτές που απεικονίζει ο Πίνακας 2 του τέταρτου κεφαλαίου.
3. καθορισμός της δειγματοληπτικής μεθόδου: αυτό είναι ένα από τα σημαντικότερα και δυσκολότερα βήματα της δειγματοληψίας γιατί από την επιλογή της μεθόδου εξαρτάται το τελικό αποτέλεσμα. Θέλουμε σε κάθε περίπτωση να διαλέγουμε την μέθοδο που δίνει το πιο ακριβές αποτέλεσμα. Σε περίπτωση που δύο ή παραπάνω μέθοδοι δίνουν το ίδιο ικανοποιητικό αποτέλεσμα προτιμάμε την γρηγορότερη. Στο έκτο κεφάλαιο θα μελετήσουμε την συμπεριφορά των μεθόδων δειγματοληψίας και τα αποτελέσματά τους εκτελώντας πειραματικές μετρήσεις.
4. καθορισμός του μεγέθους του δείγματος: και αυτό είναι σημαντικό βήμα γιατί θέλουμε το δείγμα να είναι όσο το δυνατόν μικρότερο αφού θέλουμε όσο το δυνατόν γρηγορότερη εκτίμηση. Η εμπειρία παρελθόντος μπορεί να βελτιώσει πάρα πολύ τον καθορισμό του μεγέθους του δείγματος.
5. εφαρμογή του σχεδίου δειγματοληψίας: εύρεση των δειγμάτων-αρχείων που θα εκτελεστούν
6. δειγματοληψία και συλλογή στοιχείων: εκτέλεση του δείγματος και καταγραφή των παραμέτρων που θα δώσουν τις απαραίτητες πληροφορίες για τον φόρτο εργασίας του δείγματος και κατ' επέκταση για ολόκληρη την εργασία.
7. επανεξέταση της διαδικασίας: η φάση αυτή δεν μας ενδιαφέρει άμεσα αλλά αφού εκτελεστεί η εργασία στο πλέγμα και συγκρίνουμε τον πραγματικό φόρτο εργασίας με τον εκτιμώμενο. Ειδικότερα σε περίπτωση αποτυχίας θα θέλαμε να ξέρουμε σε ποιο βήμα έγινε το λάθος.

5.3.1.2 Τυχαία δειγματοληψία

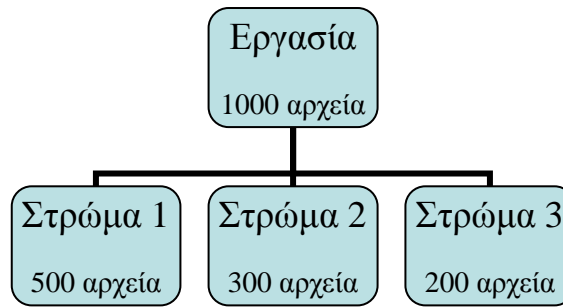
Η τυχαία δειγματοληψία είναι η απλούστερη περίπτωση δειγματοληψίας που μπορεί να εφαρμοστεί. Από τον συνολικό πληθυσμό επιλέγουμε τυχαία ένα ποσοστό αρχείων και τα εκτελούμε. Μετά την εκτέλεση των αρχείων τα διαχωρίζουμε σε ομάδες έτσι ώστε σε κάθε ομάδα τα αρχεία να έχουν τον ίδιο χρόνο εκτέλεσης, την ίδια χρησιμοποίηση μνήμης και τον ίδιο αποθηκευτικό χώρο. Έπειτα υπολογίζουμε τις σωστές αναλογίες της κάθε ομάδας στον συνολικό πληθυσμό και γενικεύουμε τα αποτελέσματα για όλη την εργασία. Για παράδειγμα υποθέτουμε ότι έχουμε μια εργασία με 100 αρχεία, εκτελούμε ένα δείγμα 10% και παίρνουμε τα εξής αποτελέσματα: 5 αρχεία είχαν χρόνο εκτέλεσης 1sec, 4MB χρησιμοποίηση μνήμης και 10KB χώρο, 3 αρχεία είχαν αντίστοιχα 2sec, 6MB, 8KB, και 2 αρχεία είχαν 5sec, 10MB, 10KB. Άρα στον συνολικό πληθυσμό έχουμε 50 αρχεία από την πρώτη κατηγορία, 30 από την δεύτερη και 20 από την τρίτη. Έτσι υπολογίζουμε τον συνολικό χρόνο εκτέλεσης και όποια άλλη παράμετρο μας ενδιαφέρει.

5.3.1.3 Συστηματική δειγματοληψία

Η συστηματική δειγματοληψία μπορεί και αυτή να εφαρμοστεί το ίδιο εύκολα με την τυχαία. Αν τοποθετήσουμε τα αρχεία σε έναν πίνακα παίρνουμε το νιοστό στοιχείο-αρχείο του πίνακα και το εκτελούμε. Η μέθοδος αυτή έχει στην ουσία την ίδια συμπεριφορά με την τυχαία και για αυτό δεν θα μελετηθεί ξεχωριστά.

5.3.1.4 Στρωματοποιημένη δειγματοληψία

Το επόμενο βήμα ήταν να προσπαθήσουμε να χωρίσουμε το σύνολο του πληθυσμού σε υποσύνολα που έχουν κάποιο κοινό χαρακτηριστικό. Πειραματικά είδαμε ότι η δημιουργία των εικόνων που ήταν frames ενός animation, για παράδειγμα η κίνηση μίας σφαίρας), είχαν τα ίδια χαρακτηριστικά (χρόνο διεκπεραίωσης, μνήμη κτλ.). Αυτό είναι αρκετά λογικό αφού δημιουργεί την ίδια εικόνα με μια μικρή μετατόπιση της σφαίρας. Το κοινό χαρακτηριστικό αυτών των αρχείων ήταν ότι είχαν περίπου το ίδιο μέγεθος σε bytes. Αυτό μας κάνει να πιστεύουμε ότι τα αρχεία, που βρίσκονται στο ίδιο project, με διαφορετικό μέγεθος μάλλον θα είναι διαφορετικά ενώ τα αρχεία που έχουν το ίδιο μέγεθος έχουν αρκετές πιθανότητες να μοιάζουν. Έτσι αφού κάνουμε το πρώτο στάδιο στρωματοποίησης εκτελούμε ένα μικρό ποσοστό από κάθε στρώμα και μετράμε τις παραμέτρους που μας ενδιαφέρουν. Έπειτα συγχωνεύουμε τα αποτελέσματα του κάθε στρώματος και εξάγουμε το τελικό αποτέλεσμα για ολόκληρη την εργασία. Στο σχήμα φαίνεται ένα παράδειγμα στρωματοποίησης μιας εργασίας με 1000 αρχεία σε τρία στρώματα.



Εικόνα 15 Στρωματοποίηση εργασίας

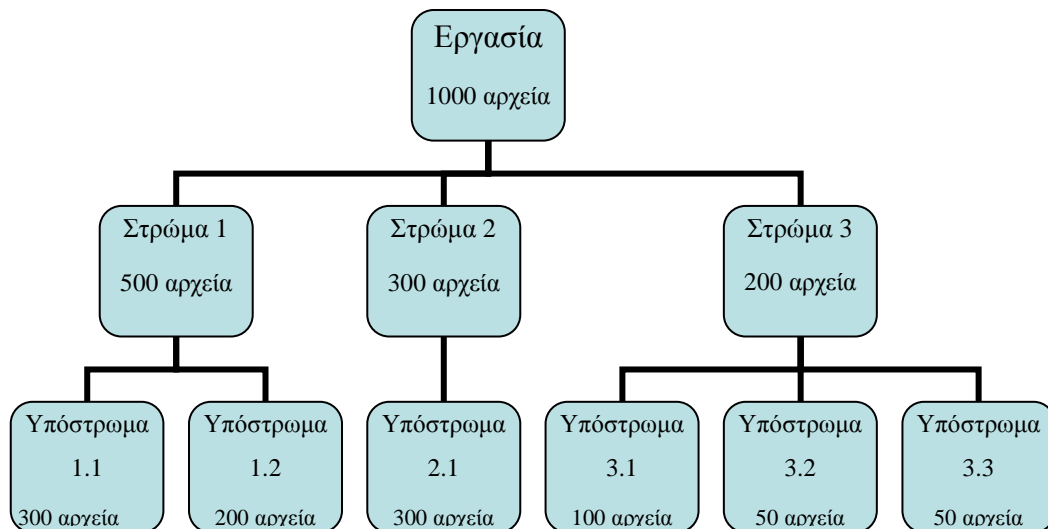
5.3.1.5 Δειγματοληψία κρίσης

Η δειγματοληψία κρίσης δεν μπορεί να εφαρμοστεί απευθείας αλλά μπορεί να εφαρμοστεί αφού πρώτα γίνει η δειγματοληψία ποσόστωσης. Αν ξέραμε ποια αρχεία είναι παρόμοια θα αρκούσε και μόλις ένα δείγμα από αυτά.

5.3.1.6 Δειγματοληψία ποσόστωσης

Μία άλλη τεχνική στρωματοποίησης και χωρισμού του πληθυσμού σε υποσύνολα είναι δυνατή μέσω μιας άλλης τεχνικής. Ο air δέχεται σε command line switch την εντολή ‘-stats’ η οποία στο τέλος του rendering δίνει διάφορες πληροφορίες. Επίσης δέχεται και την εντολή ‘-res Xsize Ysize’ όπου Xsize και Ssize είναι η οριζόντια και κάθετη ανάλυση που θα δημιουργηθεί η εικόνα. Όταν η ανάλυση είναι 1x1 ο χρόνος επεξεργασίας είναι ελάχιστος από μερικά ms μέχρι κάποια δευτερόλεπτα και εξαρτάται από το μέγεθος του αρχείου και όχι από την πολυπλοκότητα της εικόνας. Τα στατιστικά που παίρνουμε τρέχοντας ένα αρχείο σε ανάλυση 1x1 δεν μπορούν να συσχετιστούν αναλογικά με τα στατιστικά που θα παίρναμε σε μία μεγαλύτερη ανάλυση. Όμως το συμπέρασμα που βγάλαμε ήταν ότι τα αρχεία που έχουν περίπου το ίδιο μέγεθος και τα ίδια στατιστικά στο 1x1 είναι παρόμοια κατά την εκτέλεση τους σε οποιαδήποτε ανάλυση. Τα πιο βασικά στατιστικά που μεταβάλλονται με την πολυπλοκότητα της εικόνας είναι το shading structures, το shading instances, το memory in use, το memory peak και ο χρόνος εκτέλεσης από αυτά που φαίνονται στην Εικόνα 14.

Έτσι, αφού προηγηθεί η στρωματοποίηση κατά μέγεθος, τρέχοντας ένα ποσοστό των αρχείων ενός στρώματος σε ανάλυση 1x1 μπορούμε να βρούμε ποια αρχεία είναι παρόμοια και την αναλογία τους στο στρώμα και να τα διαχωρίσουμε δημιουργώντας υποστρώματα. Στην συνέχεια μπορούμε με τη δειγματοληψία κρίσης να εκτελέσουμε μόνο ένα αρχείο από κάθε υπόστρωμα στην κανονική του ανάλυση αφού είμαστε σίγουροι ότι όλα τα υπόλοιπα αρχεία του υποστρώματος αυτού έχουν τα ίδια αποτελέσματα. Στο σχήμα φαίνεται ένα παράδειγμα των στρωμάτων και των υποστρωμάτων μιας εργασίας με 1000 αρχεία. Συγκωνεύοντας τα αποτελέσματα των υποστρωμάτων βρίσκουμε τον φόρτο εργασίας για κάθε στρώμα και στην συνέχεια για ολόκληρη την εργασία.



Εικόνα 16 Στρωματοποίηση και υποστρωματοποίηση εργασίας

5.3.2 Μείωση της ανάλυσης της εικόνας

Ο χρόνος δημιουργίας μίας εικόνας από έναν renderer είναι ανάλογος των pixels που θα δημιουργηθούν. Για ένα σταθερό επίπεδο λεπτομέρειας αν διπλασιαστεί η ανάλυση, π.χ από 800x600 pixels σε 1600x1200 pixels, τότε τετραπλασιάζεται ο χρόνος εκτέλεσης [20]. Αυτό συμβαίνει γιατί με τον διπλασιασμό της ανάλυσης τετραπλασιάζεται ο αριθμός των pixels. Έτσι αν κάθε pixel θέλει τον ίδιο χρόνο για να δημιουργηθεί, ο συνολικός χρόνος αυξάνεται κατά τέσσερις φορές. Ένας τρόπος να αποκτηθεί μια γρήγορη εκτίμηση είναι να υπολογιστεί ο χρόνος που απαιτείται για την δημιουργία μίας εικόνας σε μικρότερη ανάλυση από την κανονική και στην συνέχεια να υπολογιστεί ο χρόνος στην κανονική της ανάλυση. Βέβαια μια πολύ μικρή ανάλυση, π.χ 160x80, δεν είναι ικανή να δώσει μια ακριβή αντιπροσώπευση του χρόνου εκτέλεσης. Η ελάχιστη ανάλυση είναι 640x480. Οι εικόνες με μεγάλη ανάλυση έχουν μεγάλο χρόνο εκτέλεσης όσο απλές και αν είναι. Η τεχνική αυτή μπορεί να βοηθήσει πολύ στην μείωση του χρόνου εκτέλεσης του ΕΦΕ στην περίπτωση που η δοσμένη εργασία έχει απαιτήσεις υψηλής ανάλυσης εικόνας. Μειώνοντας την ανάλυση κατά δύο ή τέσσερις φορές μειώνουμε τον χρόνο εκτέλεσης κατά τέσσερις ή δεκαέξι φορές αντίστοιχα. Στο κεφάλαιο έξι επιβεβαιώσαμε και πειραματικά αυτήν την τεχνική.

5.3.3 Σχεδιασμός του ΕΦΕ

5.3.3.1 Βασικός μηχανισμός

Στο έκτο κεφάλαιο πραγματοποιήσαμε μετρήσεις για να αξιοποιήσουμε τις μεθόδους δειγματοληψίας και τις τεχνικές μας. Τα συμπεράσματα αυτά μας βοήθησαν στον σχεδιασμό του ΕΦΕ για αυτό στην παρούσα περιγραφή θεωρούμε ως δεδομένα τα αποτελέσματα αυτά.

Ο ΕΦΕ δέχεται ως είσοδο μία εργασία που αποτελείται από αρχεία *rib*. Το πρώτο βήμα είναι να διαχωρίσει τα αρχεία σε στρώματα κατά μέγεθος. Ίδια σε μέγεθος αρχεία είναι αυτά που η απόκλιση του μεγέθους τους δεν ξεπερνάει το 5%. Για κάθε στρώμα θα εφαρμόσει την δειγματοληψία ποσοστωσίας με τον εξής τρόπο. Θα εκτελέσει σε ανάλυση 1x1 το 5% των αρχείων του στρώματος. Από τα αποτελέσματα των στατιστικών τους θα δημιουργήσει νέα υποστρώματα που το κάθε υπόστρωμα θα περιέχει τα αρχεία που τα στατιστικά τους δεν αποκλίνουν περισσότερο από 10%. Από κάθε υπόστρωμα θα εκτελέσει ένα μόνο αρχείο στην κανονική του ανάλυση και θα καταγράψει τον χρόνο εκτέλεσης, την απαιτούμενη μνήμη και τον αποθηκευτικό χώρο. Έπειτα από τις αναλογίες των δειγμάτων στα υποστρώματα θα υπολογίσει τις παραμέτρους για όλο το στρώμα. Αν N είναι τα αρχεία ενός στρώματος και κάθε υπόστρωμα έχει $n_1 \dots n_k$ αρχεία όπου k ο αριθμός των υποστρωμάτων, τότε η αναλογία του υποστρώματος k στο στρώμα είναι $\frac{n_p}{0.05}$ αρχεία στα N αρχεία, όπου $p=1..k$. Η διαδικασία επαναλαμβάνεται για όλα τα στρώματα και στο τέλος συγχωνεύονται τα αποτελέσματα.

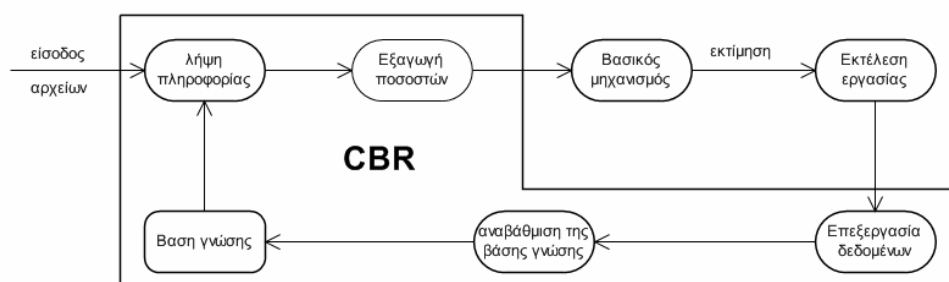
5.3.3.2 Βελτιστοποίηση του βασικού μηχανισμού

Η αξιοπιστία του βασικού μηχανισμού και ο χρόνος εκτέλεσης του ΕΦΕ εξαρτώνται σε μεγάλο βαθμό από τα τρία ποσοστά που χρησιμοποιεί. Το ποσοστό απόκλισης του μεγέθους των αρχείων 5%, το ποσοστό αρχείων που θα εκτελεστούν σε ανάλυση 1x1 5% και το ποσοστό απόκλισης των στατιστικών των αρχείων 10%. Πειραματικά είδαμε ότι με τα ποσοστά αυτά επιτυγχάνουμε αξιόπιστες και γρήγορες εκτιμήσεις. Η σταθερότητα όμως αυτών των ποσοστών μπορεί να οδηγήσει είτε σε λανθασμένες εκτιμήσεις που μειώνουν την αξιοπιστία του ΕΦΕ είτε σε άσκοπη εκτέλεση δειγμάτων που αυξάνουν τον χρόνο εκτέλεσής του. Για αυτό θα θέλαμε ο μηχανισμός να συμπεριφέρεται 'έξυπνα' και να μεταβάλλει τα ποσοστά αυτά ανάλογα με την περίπτωση. Δηλαδή θέλουμε το σύστημα μας να λαμβάνει αποφάσεις και να βελτιστοποιεί τις επιλογές του. Ένας πολύ καλός τρόπος για να γίνει αυτό είναι να αξιοποιήσει την εμπειρία του παρελθόντος και με γνώμονα τις επιτυχημένες εκτιμήσεις προηγούμενων εργασιών να αποφασίσει πως θα συμπεριφερθεί σε μία νέα εκτίμηση. Έτσι βρίσκοντας παλιότερες επιτυχημένες εκτιμήσεις που μοιάζουν με την καινούργια θα λαμβάνει αποφάσεις για την μεταβολή των ποσοστών που χρησιμοποιεί. Για αυτή την επίτευξη ενσωματώσαμε στον ΕΦΕ ένα έμπειρο σύστημα Case-Based Reasoning. Το CBR, όπως είδαμε στο τρίτο κεφάλαιο, χρησιμοποιεί τη γνώση των προηγούμενων πεπειραμένων συγκεκριμένων καταστάσεων προβλήματος και ένα νέο πρόβλημα λύνεται με την εύρεση μιας παρόμοιας προηγούμενης περίπτωσης. Οι τρεις βασικές λειτουργίες που χρειάζεται ένα CBR είναι η αναπαράσταση της γνώσης, η ανάκτηση των προηγούμενων

περιπτώσεων και η εκμάθηση της μηχανής. Δηλαδή μέσα στο πέρασμα του χρόνου και από την εκτίμηση και εκτέλεση εργασιών το σύστημα θα πρέπει να αυξάνει τη γνώση του και να μαθαίνει από τις αποτυχημένες εκτιμήσεις. Στην συνέχεια θα μελετήσουμε τον σχεδιασμό του CBR για την περίπτωση μας.

5.3.3.3 Σχεδιασμός του CBR

Για να γίνει πιο κατανοητός ο σχεδιασμός θα περιγράψουμε την λειτουργία του CBR μέσα στο γενικότερο πλαίσιο της εκτίμησης και της εκτέλεσης μιας εργασίας rendering. Όπως βλέπουμε και στο σχήμα το CBR δέχεται ως είσοδο τα αρχεία rib. Μέσα από την ανάκτηση των προηγούμενων παρόμοιων περιπτώσεων θα βρίσκει για κάθε στρώμα τα τρία ποσοστά με τα οποία θα εκτελεστεί ο βασικός μηχανισμός και που θα κάνει την εκτίμηση του φόρτου. Μετά το πέρας της εκτέλεσης όλης της εργασίας το CBR θα είναι σε θέση να επεξεργαστεί τα αποτελέσματα, να βρει ποια θα ήταν τα ιδανικότερα ποσοστά επιτυχημένης δειγματοληψίας και να αναβαθμίσει τη βάση γνώσης.



Εικόνα 17 Βασική λειτουργία του CBR

Το πρώτο κομμάτι του σχεδιασμού που θα μελετήσουμε είναι τι θεωρούμε ως γνώση που μπορεί να εξαχθεί από την εμπειρία παρελθόντος. Μια προηγούμενη εργασία μπορεί να χαρακτηριστεί από τον συνολικό αριθμό των αρχείων rib που περιέχει, τον αριθμό των στρωμάτων, τον αριθμό των αρχείων που περιέχει κάθε στρώμα, το ποσοστό μέγιστης απόκλισης μεγέθους (%ΠΑΜ) των αρχείων του κάθε στρώματος, το ποσοστό εκτέλεσης αρχείων σε ανάλυση 1x1 (%ΠΕ) για κάθε στρώμα, τον αριθμό των υποστρωμάτων και το ποσοστό μέγιστης απόκλισης των στατιστικών για κάθε υπόστρωμα.

Η πρώτη πληροφορία που χρειάζεται να ανακτήσει το σύστημα είναι με πιο ποσοστό απόκλισης μεγέθους θα στρωματοποιήσει τα αρχεία. Στον βασικό μηχανισμό το ποσοστό αυτό ήταν 5% αλλά μπορεί με την πάροδο του χρόνου και την εκτέλεση εργασιών να αποδειχτεί ότι είναι πολύ μικρό ή πολύ μεγάλο. Ένας πιο αποτελεσματικός τρόπος θα ήταν να υπολογίζει τον μέσο όρο της απόκλισης από προηγούμενες παρόμοιες δουλειές. Σε πρώτο επίπεδο παρόμοιες εργασίες μπορούν να θεωρηθούν αυτές που έχουν περίπου τον ίδιο αριθμό αρχείων. Τις περιπτώσεις αυτές θα τις ονομάσουμε περιπτώσεις Α επιπέδου. Αφού ανακτήσει

όλες τις προηγούμενες ταιριαστές περιπτώσεις θα υπολογίζει τον μέσο όρο του ποσοστού της απόκλισης του μεγέθους όλων των στρωμάτων από όλες τις ταιριασμένες εργασίες. Έπειτα η στρωματοποίηση θα γίνεται με αυτό το ποσοστό στην νέα εργασία. Αφού γίνει η στρωματοποίηση το σύστημα θα είναι σε θέση να γνωρίζει ποια είναι η πραγματική απόκλιση του κάθε στρώματος και να καταγράψει αυτές. Για παράδειγμα μπορεί να γίνει μια στρωματοποίηση με απόκλιση μεγέθους 5% και να δημιουργηθούν δύο στρώματα όπου στο πρώτο το μέγεθος των αρχείων να αποκλίνει κατά 0,5% και στο δεύτερο κατά 1%. Αυτές είναι και οι πραγματικές τιμές απόκλισης του μεγέθους. Στο επόμενο βήμα το σύστημα θα ψάξει στις περιπτώσεις Α επιπέδου και για κάθε στρώμα θα ανακτήσει προηγούμενα στρώματα που ταιριάζουν με αυτό. Ως ταιριασμα θεωρούμε το νέο με τα παλιά στρώματα να έχουν τον ίδιο αριθμό αρχείων και την ίδια απόκλιση μεγέθους. Αυτές θα τις ονομάσουμε περιπτώσεις Β επιπέδου. Από αυτήν την ανάκτηση θα βρει τα ποσοστά εκτέλεσης αρχείων στην ανάλυση 1x1. Είναι φανερό ότι στα ποσοστά αυτά θα υπάρχουν διάφορες τιμές. Τα πιο μικρά ποσοστά θα αποφέρουν γρηγορότερες εκτιμήσεις ενώ τα μεγαλύτερα ποσοστά θα αποφέρουν ακριβέστερες εκτιμήσεις. Ο υπολογισμός του τελικού ποσοστού που θα εφαρμοστεί στο στρώμα μπορεί να υπολογιστεί με δύο τρόπους. Ο ένας είναι να υπολογίζεται ο μέσος όρος αυτών ενώ ο δεύτερος είναι να δίνεται από τον χρήστη μία παράμετρος που θα επιλέγει αν θέλει γρήγορη εκτίμηση ή αν θέλει ακριβή εκτίμηση. Την παράμετρο αυτή θα την ονομάσουμε δείκτη αξιοπιστίας και θα υλοποιηθεί με τον εξής τρόπο. Τα ποσοστά θα ταξινομηθούν κατά αύξουσα σειρά και ο δείκτης θα παίρνει τιμές από το -100 ως το 100. Το -100 σημαίνει ότι ο χρήστης θέλει μία γρήγορη εκτίμηση και θα πάρει το μικρότερο ποσοστό. Το 100 σημαίνει ότι ο χρήστης θέλει μία ακριβή εκτίμηση και θα πάρει το μεγαλύτερο ποσοστό. Στο μηδέν θα τοποθετηθεί ο μέσος όρος των ποσοστών. Για τις ενδιάμεσες τιμές από [-100,0) το ποσοστό θα παίρνει τιμές μεταξύ του μικρότερου ποσοστού και του μέσου όρου αναλογικά ενώ για τις τιμές (0,100] η τιμή του ποσοστού θα μεταβάλλεται από τον μέσο όρο μέχρι το μέγιστο ποσοστό αναλογικά. Ας υποθέσουμε ότι για μια συγκεκριμένη εργασία ανακτά τα ποσοστά 1,2,3,4,5,6,7,8 %. Ο μέσος όρος είναι το $m=4,5$ και αντιστοιχεί στην τιμή μηδέν του δείκτη αξιοπιστίας. Για οποιαδήποτε τιμή d του δείκτη στο διάστημα $[0,100]$

το ποσοστό δίνεται από τον τύπο $p = \left(\frac{8 - 4,5}{100} \right) \cdot d + m$. Βλέπουμε ότι για $d=0$ έχουμε $p=m$

και για $d=100$ έχουμε $p=8\%$ που είναι το μέγιστο ποσοστό. Για τις τιμές από $[-100,0]$ ο τύπος

$$\text{είναι } p = \left(\frac{4,5 - 1}{100} \right) \cdot d + m.$$

Αυτό που μένει είναι το CBR να υπολογίσει το τρίτο και τελευταίο ποσοστό απόκλισης των στατιστικών των αρχείων που θα εκτελεστούν σε ανάλυση 1x1. Ο βασικός μηχανισμός έχει ως προκαθορισμένη τιμή το 10%. Όμως μπορεί σε κάποιες περιπτώσεις τα αρχεία να είναι

της ίδιας πολυπλοκότητας αλλά να εμφανίζουν μεγαλύτερη απόκλιση στα στατιστικά τους. Ένα μεγάλο ποσοστό σημαίνει μικρότερο αριθμό υποστρωμάτων και γρηγορότερη εκτίμηση ενώ ένα μικρό ποσοστό σημαίνει μεγάλο αριθμό υποστρωμάτων και ακριβέστερη εκτίμηση. Από τις περιπτώσεις Β επιπέδου το σύστημα θα βρει τα στρώματα που περιέχουν υποστρώματα και θα ανακτήσει τα ποσοστά αυτά. Έπειτα με τον ίδιο τρόπο όπως και πριν με τον δείκτη αξιοπιστίας θα υπολογίσει το ποσοστό. Η μόνη διαφορά είναι ότι για τις θετικές τιμές του δείκτη το ποσοστό πρέπει να μικραίνει ενώ για τις αρνητικές τιμές να μεγαλώνει. Δηλαδή αν ανακτήσει τα ποσοστά 1,2,3,4,5,6,7,8 % το τελικό ποσοστό θα είναι

$$p = \left(\frac{1 - 4,5}{100} \right) \cdot d + m \text{ για } d=[0,100] \text{ και } m \text{ ο μέσος όρος και } p = \left(\frac{4,5 - 8}{100} \right) \cdot d + m \text{ για } d=[-$$

100,0]. Αφού το CBR βρει τα τρία αυτά ποσοστά για κάθε στρώμα η εκτίμηση θα γίνει από τον βασικό μηχανισμό με την διαφορά ότι για κάθε στρώμα θα εφαρμόζονται διαφορετικά ποσοστά και συγκεκριμένα αυτά που έδωσε η αξιοποίηση της εμπειρίας παρελθόντος. Μετά την υποστρωματοποίηση θα βρίσκονται τα πραγματικά ποσοστά απόκλισης στατιστικών για κάθε υπόστρωμα και θα καταγράφονται αυτά. Στην περίπτωση που δεν βρεθεί καμία προηγούμενη ταιριαστή περίπτωση ο βασικός μηχανισμός θα εκτελείται με τα προκαθορισμένα ποσοστά. Ως τεχνική ανάκτησης και ταιριάσματος χρησιμοποιούμε την τεχνική πλησίον-γειτόνων που περιγράψαμε στο τρίτο κεφάλαιο. Βέβαια επειδή τα χαρακτηριστικά γνωρίσματα που ταιριάζουμε για να γίνει η ανάκτηση είναι αριθμοί δεχόμαστε μια απόκλιση της τάξεως του 5%. Έτσι μία εργασία με 1000 αρχεία είναι παρόμοια με μια άλλη που έχει 950 αρχεία. Η απόκλιση αυτή μπορεί να θεωρηθεί ότι είναι η προσαρμογή περίπτωσης του CBR.

Το τελευταίο στάδιο της σχεδίασης του CBR είναι η εκμάθηση της μηχανής δηλαδή η αναβάθμιση της βάσης της γνώσης. Αφού εκτελεστεί η εργασία θα είναι πλέον γνωστός ο φόρτος για κάθε στρώμα και υπόστρωμα οπότε το CBR θα είναι σε θέση να εκτιμήσει ποια θα ήταν η καλύτερη δειγματοληψία. Στην περίπτωση επιτυχημένης εκτίμησης η διαδικασία αναβάθμισης θα προσπαθήσει να βελτιώσει την δειγματοληψία και θα εκτελέσει με τη σειρά τους ακόλουθους συλλογισμούς.

- Αν δύο ή περισσότερα υποστρώματα, ενός στρώματος, έχουν τον ίδιο φόρτο τότε συνένωσε τα υποστρώματα σε ένα υπόστρωμα και όρισε ως νέο ποσοστό απόκλισης στατιστικών την μέγιστη απόκλιση των στατιστικών του νέου υποστρώματος. Το νέο ποσοστό δεν θα πρέπει σε καμία περίπτωση να ξεπερνάει το 20%. Αν το ξεπερνάει δεν θα γίνεται η συνένωση. Η διαδικασία αυτή δείχνει ότι κατά την υποστρωματοποίηση το ποσοστό απόκλισης ήταν μικρό και αδίκως δημιουργήθηκαν διαφορετικά υποστρώματα.

- Αν ένα στρώμα δεν περιέχει υποστρώματα τότε όρισε ως μηδέν το νέο ποσοστό εκτέλεσης αρχείων στην ανάλυση 1x1. Το μηδέν σημαίνει ότι στο στρώμα αυτό δεν θα χρειαζόταν γίνει υποστρωματοποίηση. Ένα αρχείο είναι αρκετό για την εκτίμηση του φόρτου αυτού του στρώματος.
- Αν δύο ή περισσότερα στρώματα έχουν τον ίδιο φόρτο τότε συνένωσε τα στρώματα σε ένα στρώμα και όρισε το νέο ποσοστό απόκλισης μεγέθους και μηδέν το νέο ποσοστό εκτέλεσης αρχείων στην ανάλυση 1x1. Το νέο ποσοστό δεν θα πρέπει σε καμία περίπτωση να ξεπερνάει το 10%. Αν το ξεπερνάει δεν θα γίνεται η συνένωση.

Ο λόγος που περιορίζουμε τις συνενώσεις με τα ανώτατα όρια που ορίσαμε είναι γιατί θέλουμε να αποφύγουμε συμπτώσεις. Για παράδειγμα αν τύχει ένα στρώμα να έχει αρχεία μεγέθους 1KB και ένα άλλο στρώμα να έχει αρχεία 5KB, αλλά τυχαίνει να έχουν τον ίδιο φόρτο, δεν θα θέλαμε να γίνει συνένωση και το ποσοστό απόκλισης να φτάσει την τιμή 400%.

Σε περίπτωση αποτυχημένης εκτίμησης η διαδικασία αναβάθμισης θα πρέπει να ψάξει να βρει σε ποια στρώματα ή υποστρώματα έγινε το λάθος. Λανθασμένη εκτίμηση σε ένα στρώμα μπορεί να γίνει γιατί δεν εκτελέστηκαν τα σωστά δείγματα στο στρώμα. Για το λάθος αυτό ευθύνεται το ποσοστό εκτέλεσης των αρχείων σε ανάλυση 1x1. Αν το ποσοστό είναι μικρότερο του 5% θα διορθώνεται και θα ορίζεται 5%. Αν είναι μεγαλύτερο ή ίσο του 5% θα ορίζεται 10%. Λάθος εκτίμηση σε ένα υπόστρωμα μπορεί να γίνει γιατί η απόκλιση των στατιστικών ήταν πολύ μεγάλη με αποτέλεσμα να θεωρηθούν ίδια κάποια υποστρώματα που τελικά δεν ήταν. Αναλύοντας τα αποτελέσματα των αρχείων του υποστρώματος το σύστημα θα είναι σε θέση να υπολογίσει το σωστό ποσοστό.

5.3.3.4 Ενσωμάτωση της τεχνικής μείωσης της ανάλυσης

Στο κεφάλαιο αυτό θα περιγράψουμε πως μπορεί να ενσωματωθεί η τεχνική αυτή μέσα στον μηχανισμό του ΕΦΕ. Η τεχνική αυτή μπορεί να βελτιώσει τον μηχανισμό ώστε να εκτελείται πολύ πιο γρήγορα με αντίτιμο την απόκλιση της ακρίβειας. Στην περίπτωση που μια εργασία έχει απαιτήσεις σε υψηλή ανάλυση μεγαλύτερη από την 640x480 θα ενεργοποιείται αυτή η τεχνική. Την εκτίμηση αυτή θα την ονομάσουμε εξαναγκασμένη εκτίμηση (forced estimation). Ανάλυση 640x480 σημαίνει 307200 pixels. Όταν μια εργασία έχει ανάλυση $X \times Y$ και $XY > 307200$ τότε θα γίνεται εξαναγκασμένη εκτίμηση και τα δείγματα θα εκτελούνται στην ανάλυση 640x480 απ' όπου θα υπολογίζεται ο εκτιμώμενος χρόνος. Αυτός ο χρόνος όμως αφορά την εκτίμηση της εργασίας σε ανάλυση 640x480. Για την μετατροπή του στην κανονική εκτίμηση θα πολλαπλασιάζεται με τον αριθμό $\left(\frac{X \cdot Y}{640 \cdot 480} \right)$.

6

Πειράματα και μετρήσεις

6.1 Παράμετροι αξιολόγησης

Αυτό που μας ενδιαφέρει να αξιολογήσουμε είναι η συμπεριφορά των μεθόδων δειγματοληψίας και των τεχνικών μας ώστε να εξάγουμε βάσιμα αποτελέσματα τα οποία θα τα χρησιμοποιήσουμε για την σχεδίαση ενός αποτελεσματικού και αξιόπιστου ΕΦΕ.. Είδαμε ότι για να επιτύχουμε μία εκτίμηση του φόρτου εργασίας θέλουμε να έχουμε γνώση του συνολικού χρόνου διεκπεραίωσης, της μέγιστης μνήμης που χρησιμοποιήθηκε και του απαιτούμενου αποθηκευτικού χώρου. Μέσα από τα πειράματα θα δούμε ποιες μέθοδοι δειγματοληψίας δίνουν επαρκές δείγμα που να είναι αντιπροσωπευτικό ολόκληρου του πληθυσμού ώστε τις παραμέτρους που μετρήσαμε στο δείγμα να μπορούμε να τις γενικεύσουμε για ολόκληρη την εργασία. Οι παράμετροι που θα μετρήσουμε και που θα μας βοηθήσουν να αξιολογήσουμε τις τεχνικές είναι οι εξής:

- Η απόκλιση του χρόνου εκτίμησης προς τον πραγματικό χρόνο εκτέλεσης της εργασίας.
- Ο χρόνος εκτέλεσης του ΕΦΕ για να κάνει την εκτίμηση.
- Σύγκριση της εκτιμώμενης μνήμης σε σχέση με την πραγματική μνήμη που απαιτείται.

- Σύγκριση του εκτιμώμενου αποθηκευτικού χώρου σε σχέση με τον πραγματικό χώρο που απαιτείται.
- Σύγκριση του χρόνου εκτέλεσης σε σχέση με την ανάλυση.

6.2 Σύστημα αξιολόγησης

Στην περίπτωση μας το σύστημα με το οποίο θα αξιολογήσουμε τις τεχνικές δειγματοληψίας είναι πολύ απλό. Για κάθε εργασία και για κάθε μετρούμενο μέγεθος θα εφαρμόσουμε την τυχαία, την στρωματοποιημένη και την δειγματοληψία ποσόστωσης όπως περιγράφηκαν στο προηγούμενο κεφάλαιο. Κάθε πείραμα θα εκτελείται από τρεις φορές με ποσοστά 1, 5 και 10%. Ο λόγος που γίνεται αυτό είναι γιατί θέλουμε να δούμε πώς ανταποκρίνονται σε διαφορετικά ποσοστά ενώ η επανάληψη της μέτρησης αυξάνει την εγκυρότητα της μεθόδου αφού λαμβάνει μέρος και ο τυχαίος παράγοντας. Στην τυχαία και στην στρωματοποιημένη δειγματοληψία το ποσοστό αναφέρεται στο μέγεθος του δείγματος που θα εκτελεστεί ενώ στην δειγματοληψία ποσόστωσης αναφέρεται στο μέγεθος του δείγματος που θα εφαρμοστεί η τεχνική 1x1. Έπειτα θα γενικεύουμε τα αποτελέσματα του δείγματος για ολόκληρο τον πληθυσμό και θα συγκρίνουμε την απόκλιση τους με τις πραγματικές τιμές. Τις πραγματικές τιμές μιας εργασίας μπορούμε να τις γνωρίζουμε αφού εκτελέσουμε ολόκληρη την εργασία.

6.3 Οργάνωση πειραμάτων

Για τα πειράματα χρησιμοποιήσαμε έτοιμα animation σε μορφή gif αρχείων. Η κάθε συλλογή είχε από 100 ως 600 αρχεία. Ο Πίνακας 3 απεικονίζει τις συλλογές. Στην πρώτη στήλη είναι το όνομα της συλλογής, στην δεύτερη το πλήθος των αρχείων που περιέχει, στην τρίτη το μέγεθος των αρχείων, στην τέταρτη ο χρόνος διεκπεραίωσης της εργασίας και στην τελευταία η απαιτούμενη μνήμη που χρησιμοποιεί κάθε αρχείο. Επειδή σε κάθε συλλογή τα αρχεία έχουν σχεδόν τα ίδια χαρακτηριστικά και την ίδια πολυπλοκότητα κατά την εκτέλεση τους είναι φανερό ότι οποιαδήποτε δειγματοληψία σε μία συλλογή θα δώσει σωστά αποτελέσματα. Για αυτό οι μετρήσεις έγιναν σε συνδυασμούς συλλογών ώστε να αυξήσουμε την πολυπλοκότητα των εργασιών. Σκοπός μας λοιπόν ήταν να δημιουργήσουμε συνδυασμούς τέτοιους που να δυσκολεύουν την εκτίμηση για να δούμε ποια μέθοδος δειγματοληψίας είναι πιο αξιόπιστη. Όλες οι μετρήσεις έγιναν σε ανάλυση εικόνας 320x240 και σε tif format εκτός αν αναφέρεται κάτι διαφορετικό. Ίδια αρχεία σε μέγεθος θεωρήθηκαν αυτά που δεν είχαν μεταξύ τους μεγαλύτερη απόκλιση από το 5%. Στην τεχνική 1x1 η απόκλιση των στατιστικών των αρχείων ήταν 10% για να θεωρηθούν όμοια. Τα πειράματα που θα παρουσιάσουμε είναι ένα μικρό μέρος των πειραμάτων που εκτελέσαμε. Τα

αποτελέσματα που εξάγαμε βασίζονται σε πλήθος πειραμάτων αλλά για οικονομία χώρου θα παρουσιάσουμε τα πιο αντιπροσωπευτικά.

Πίνακας 3 Πειραματικό υλικό από συλλογές με αρχεία rib

COLLECTION	FILES	SIZE (KB)	REAL TIME (sec)	MEMORY (MB)
Rib1	600	148	700	4,7
Rib2	300	2900	601	12,7
Rib3	300	152	457	6,8
Rib4	300	148	390	4,7
Rib7	300	152	404	5,4
Rib8	300	148	300	4,5
Rib9	100	52440	2520	58,4

6.4 Αποτελέσματα

6.4.1 Χρονικές μετρήσεις

Τα πρώτα πειράματα που εκτελέσαμε ήταν στις χρονικές μετρήσεις αφού ο χρόνος είναι η σημαντικότερη παράμετρος για τον προσδιορισμό του φόρτου εργασίας. Στις παρακάτω μετρήσεις καταγράφονται ο πραγματικός χρόνος εκτέλεσης της εργασίας (REAL TIME), ο εκτιμώμενος χρόνος (ESTIMATED TIME), η απόκλιση του εκτιμώμενου χρόνου προς τον πραγματικό σύμφωνα με τον τύπο $var = \frac{|estimateTime - realTime|}{realTime} \cdot 100 \%$, ο χρόνος

εκτέλεσης του ΕΦΕ (WEP TIME, Workload Estimator Predictor Time), ο αριθμός των αρχείων που περιέχει η συλλογή, τα ονόματα των συλλογών που συνδυάσαμε προς εκτέλεση π.χ Rib1+Rib9 και ο αύξων αριθμός του πειράματος. Με τα κόκκινα νούμερα απεικονίζονται οι αποκλίσεις που έχουν ξεπεράσει το 20% και θεωρούνται αποτυχημένες εκτιμήσεις.

6.4.1.1 Μετρήσεις στην τυχαία δειγματοληψία

Οι πρώτες μετρήσεις πραγματοποιήθηκαν στην τυχαία δειγματοληψία που είναι και η απλούστερη.

Μέτρηση 1 Τυχαία δειγματοληψία στις συλλογές 1+9 (Χρόνος)

Rib1+Rib9	#files	REAL TIME (sec)	VAR (%)			ESTIMATED (sec)			WEP TIME (sec)		
experiment			1%	5%	10%	1%	5%	10%	1%	5%	10%
1	700	3220	17,1	3,3	9,1	3772	3327	3516	37	166	351
2	700	3220	74,8	14,7	28,1	810	2744	4127	8	137	412
3	700	3220	74,7	18,0	31,1	813	3800	2217	8	189	221

Μέτρηση 2 Τυχαία δειγματοληψία στις συλλογές 2+8 (Χρόνος)

Rib2+Rib8	#files	REAL TIME (sec)	VAR (%)			ESTIMATED (sec)			WEP TIME (sec)		
experiment			1%	5%	10%	1%	5%	10%	1%	5%	10%
1	600	901	15,0	6,5	0,7	1037	960	908	10	48	90
2	600	901	18,8	5,7	0,3	731	849	898	7	42	90
3	600	901	10,6	2,9	0,2	997	928	903	10	46	90

Μέτρηση 3 Τυχαία δειγματοληψία στις συλλογές 8+9 (Χρόνος)

Rib8+Rib9	#files	REAL TIME (sec)	VAR (%)			ESTIMATED (sec)			WEP TIME (sec)		
experiment			1%	5%	10%	1%	5%	10%	1%	5%	10%
1	400	2820	3,5	33,5	24,8	2720	1875	2129	27	93	213
2	400	2820	15,7	34,1	28,9	3264	3784	3637	32	189	363
3	400	2820	85,3	28,3	5,3	414	3619	2972	4	181	297

Μέτρηση 4 Τυχαία δειγματοληψία στις συλλογές 1+2+9 (Χρόνος)

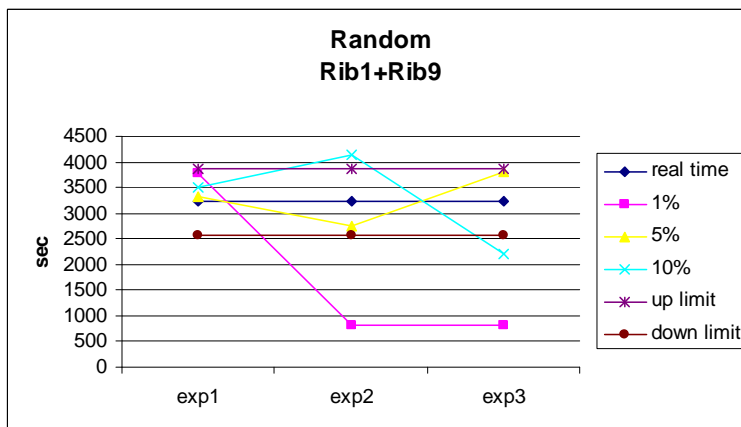
Rib1+Rib2 +Rib9	#files	REAL TIME (sec)	VAR (%)			ESTIMATED (sec)			WEP TIME (sec)		
experiment			1%	5%	10%	1%	5%	10%	1%	5%	10%
1	1000	3821	19,2	20,0	17,7	4558	3056	4500	45	153	450
2	1000	3821	55,5	47,9	3,7	5943	5655	3966	59	283	397
3	1000	3821	6,6	28,1	0,1	4074	4896	3815	40	244	382

Παρακάτω απεικονίζουμε γραφικά τα αποτελέσματα των παραπάνω μετρήσεων. Σε κάθε γραφική παράσταση φαίνονται ο πραγματικός χρόνος εκτέλεσης της εργασίας, το άνω και

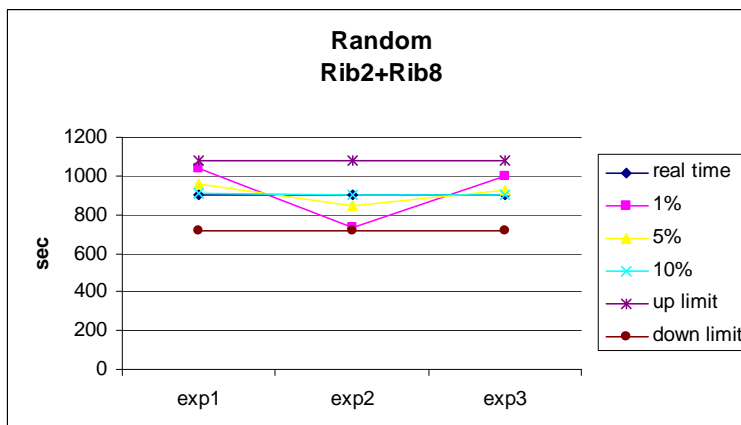
κάτω αποδεκτό όριο που είναι $\pm 20\%$ του πραγματικού χρόνου και τα αποτελέσματα των τριών πειραμάτων για τα ποσοστά 1%, 5% και 10%. Παρατηρούμε ότι από τις τέσσερις μετρήσεις μόνο η Μέτρηση 2 έχει επιτυχημένη εκτίμηση σε όλα τα πειράματα. Αυτό συμβαίνει γιατί η εργασία Rib2 και η Rib8 είναι της ίδιας πολυπλοκότητας. Αντίθετα βλέπουμε ότι η εισαγωγή της εργασίας Rib9 που είναι πιο πολύπλοκη και πιο χρονοβόρα αυξάνει την πιθανότητα λάθος εκτίμησης.

Συμπεραίνουμε λοιπόν ότι το τυχαίο δείγμα δεν είναι αντιπροσωπευτικό του πληθυσμού και δεν παίρνει πάντα την σωστή αναλογία αρχείων που πρέπει να εκτελεστεί. Ειδικότερα οι μεγαλύτερες αποκλίσεις συμβαίνουν στο ποσοστό 1%. Το γεγονός αυτό μας κάνει να πιστεύουμε ότι η τυχαία δειγματοληψία δεν είναι αξιόπιστη στα μικρά ποσοστά και ότι η απόκλιση του χρόνου εκτίμησης από τον πραγματικό χρόνο θα μεγαλώνει όσο μεγαλώνει και η πολυπλοκότητα της εργασίας. Μεγαλώνοντας το ποσοστό του δείγματος θα μεγαλώνει και η πιθανότητα σωστής εκτίμησης αλλά θέλουμε το δείγμα να είναι όσο το δυνατόν μικρότερο.

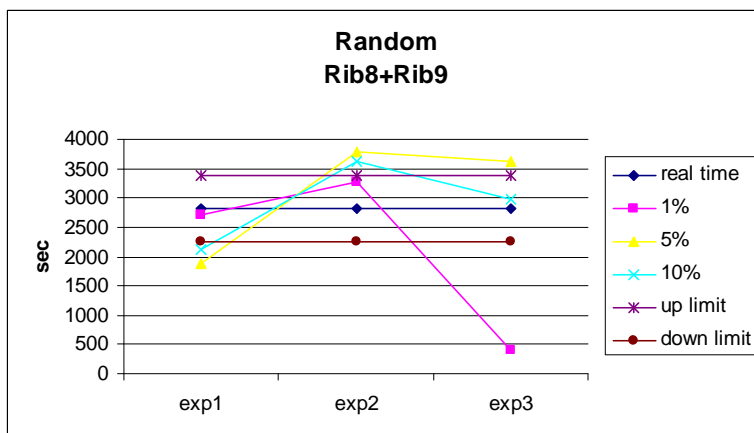
Γραφική παράσταση 1 Τυχαία δειγματοληψία στις συλλογές 1+9 (Χρόνος)



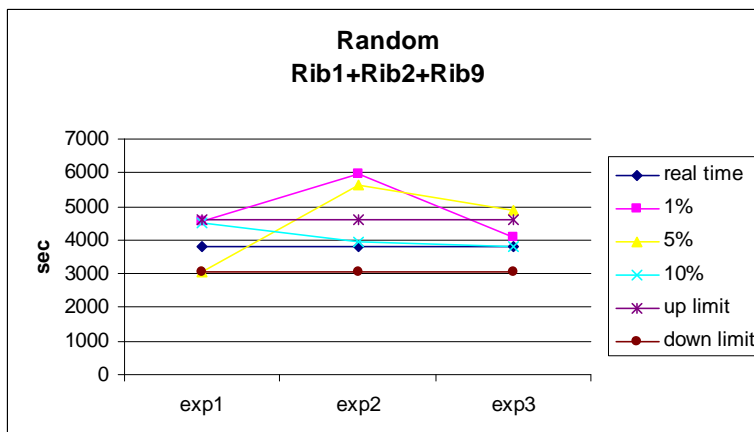
Γραφική παράσταση 2 Τυχαία δειγματοληψία στις συλλογές 2+8 (Χρόνος)



Γραφική παράσταση 3 Τυχαία δειγματοληψία στις συλλογές 8+9 (Χρόνος)



Γραφική παράσταση 4 Τυχαία δειγματοληψία στις συλλογές 1+2+9 (Χρόνος)



6.4.1.2 Μετρήσεις στην στρωματοποιημένη δειγματοληψία

Στην συνέχεια πραγματοποιήσαμε μετρήσεις με την μέθοδο της στρωματοποιημένης δειγματοληψίας όπως περιγράφηκε στο πέμπτο κεφάλαιο. Δηλαδή διαχωρίσαμε τα αρχεία και τα στρωματοποιήσαμε με βάση το μέγεθός τους. Κάθε πείραμα εκτελέστηκε από τρεις φορές με ποσοστά 1, 5 και 10% από κάθε στρώμα.

Μέτρηση 5 Στρωματοποιημένη δειγματοληψία στις συλλογές 1+9 (Χρόνος)

Rib1+Rib9 experiment	#files	REAL TIME (sec)	VAR (%)			ESTIMATED (sec)			WEP TIME (sec)		
			1%	5%	10%	1%	5%	10%	1%	5%	10%
1	700	3220	5,5	0,9	0,4	3041	3252	3358	30	162	335
2	700	3220	0,2	9,7	2,3	3229	3534	3295	32	176	329
3	700	3220	4,7	7,1	6,0	3373	3449	3415	33	172	341

Μέτρηση 6 Στρωματοποιημένη δειγματοληψία στις συλλογές 2+8 (Χρόνος)

Rib2+Rib8	#files	REAL TIME (sec)	VAR (%)			ESTIMATED (sec)			WEP TIME (sec)		
experiment			1%	5%	10%	1%	5%	10%	1%	5%	10%
1	600	901	2,5	1,4	2,1	878	888	882	8	44	88
2	600	901	0,8	1,8	1,6	909	884	886	9	44	88
3	600	901	1,1	1,2	1,7	911	890	885	9	44	88

Μέτρηση 7 Στρωματοποιημένη δειγματοληψία στις συλλογές 8+9 (Χρόνος)

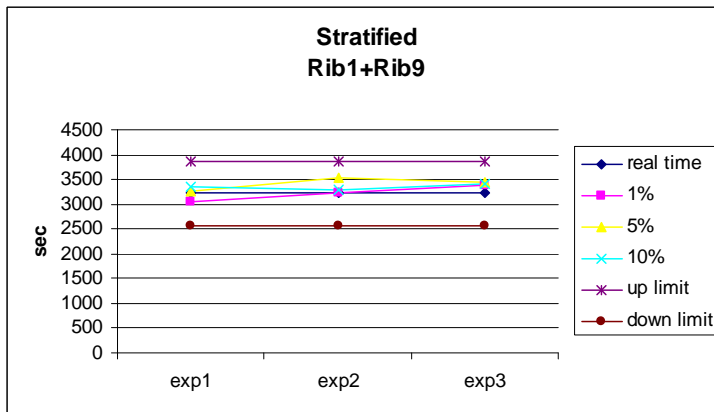
Rib8+Rib9	#files	REAL TIME (sec)	VAR (%)			ESTIMATED (sec)			WEP TIME (sec)		
experiment			1%	5%	10%	1%	5%	10%	1%	5%	10%
1	400	2820	4,9	3,6	3,5	2686	2721	2921	26	136	292
2	400	2820	1,2	0,0	2,5	2856	2822	2892	28	141	289
3	400	2820	8,5	7,8	4,9	3062	3042	2959	30	152	296

Μέτρηση 8 Στρωματοποιημένη δειγματοληψία στις συλλογές 1+2+9 (Χρόνος)

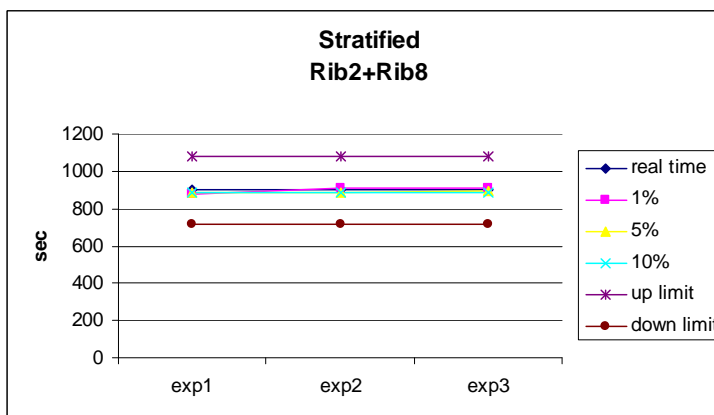
Rib1+Rib2 +Rib9	#files	REAL TIME (sec)	VAR (%)			ESTIMATED (sec)			WEP TIME (sec)		
experiment			1%	5%	10%	1%	5%	10%	1%	5%	10%
1	1000	3821	2,0	4,3	1,0	3899	3989	3863	39	199	386
2	1000	3821	13,6	5,5	3,2	4341	4033	3945	43	201	394
3	1000	3821	1,3	0,2	4,2	3770	3832	3985	37	191	398

Από τις παρακάτω γραφικές παραστάσεις των παραπάνω μετρήσεων παρατηρούμε ότι όλες οι εκτιμήσεις σε όλες τις εργασίες είναι πετυχημένες μέσα στα όρια και συγκεκριμένα πολύ κοντά στην πραγματική τιμή. Αυτό μας κάνει να πιστεύουμε ότι ο διαχωρισμός κατά μέγεθος είναι ένα πολύ καλό κριτήριο στρωματοποίησης. Στην περίπτωση όμως που σε μια εργασία υπάρχουν αρχεία με το ίδιο μέγεθος αλλά με διαφορετική πολυπλοκότητα και διαφορετικό χρόνο εκτέλεσης η στρωματοποιημένη δειγματοληψία λειτουργεί όπως η τυχαία για το κάθε στρώμα. Δηλαδή στα μικρά ποσοστά έχουμε μεγάλες πιθανότητες για λάθος εκτίμηση.

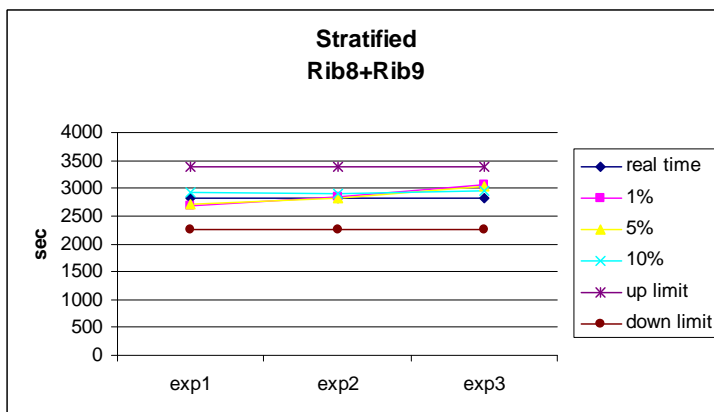
Γραφική παράσταση 5 Στρωματοποιημένη δειγματοληψία στις συλλογές 1+9 (Χρόνος)



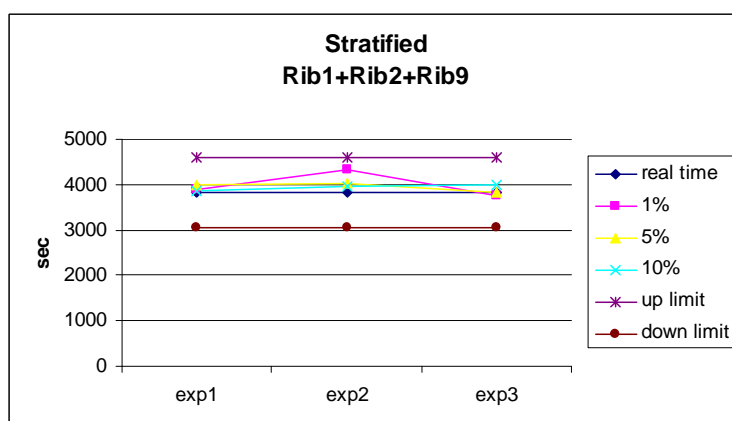
Γραφική παράσταση 6 Στρωματοποιημένη δειγματοληψία στις συλλογές 2+8 (Χρόνος)



Γραφική παράσταση 7 Στρωματοποιημένη δειγματοληψία στις συλλογές 8+9 (Χρόνος)



Γραφική παράσταση 8 Στρωματοποιημένη δειγματοληψία στις συλλογές 1+2+9 (Χρόνος)



6.4.1.3 Μετρήσεις στην δειγματοληψία ποσόστωσης

Παρακάτω παραθέτουμε μετρήσεις και τις γραφικές παραστάσεις τους με την δειγματοληψία ποσόστωσης εφαρμόζοντας την τεχνική μείωσης της ανάλυσης 1x1.

Μέτρηση 9 Δειγματοληψία ποσόστωσης στις συλλογές 1+9 (Χρόνος)

Rib1+Rib9	#files	REAL TIME (sec)	VAR (%)			ESTIMATED (sec)			WEP TIME (sec)		
			1%	5%	10%	1%	5%	10%	1%	5%	10%
1	700	3220	3,8	2,5	7,8	3343	3137	2968	34	86	146
2	700	3220	1,8	6,6	1,9	3279	3005	3158	34	80	117
3	700	3220	7,7	2,1	7,2	2970	3288	3452	31	89	123

Μέτρηση 10 Δειγματοληψία ποσόστωσης στις συλλογές 2+8 (Χρόνος)

Rib2+Rib8	#files	REAL TIME (sec)	VAR (%)			ESTIMATED (sec)			WEP TIME (sec)		
			1%	5%	10%	1%	5%	10%	1%	5%	10%
1	600	901	2,7	1,7	2,2	876	885	881	8	20	37
2	600	901	2,2	1,6	0,6	881	886	895	6	20	36
3	600	901	0,2	2,4	0,1	899	923	900	6	20	35

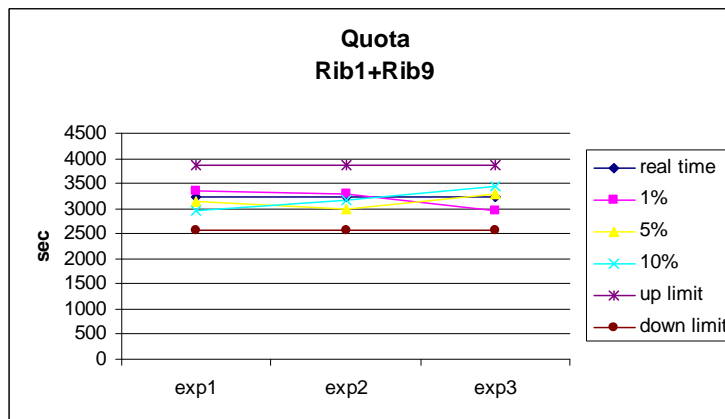
Μέτρηση 11 Δειγματοληψία ποσόστωσης στις συλλογές 8+9 (Χρόνος)

Rib8+Rib9	#files	REAL TIME (sec)	VAR (%)			ESTIMATED (sec)			WEP TIME (sec)		
			1%	5%	10%	1%	5%	10%	1%	5%	10%
1	400	2820	7,3	7,3	2,4	3026	2613	2889	33	50	79
2	400	2820	2,6	8,1	11,8	2896	3051	3153	32	49	78
3	400	2820	13,9	6,7	5,4	2428	2629	2665	26	51	75

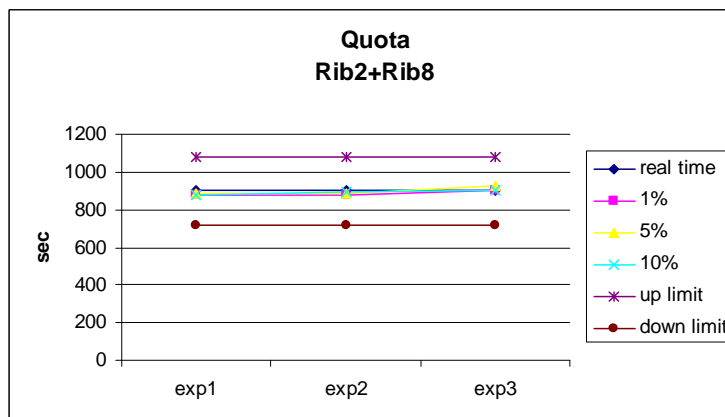
Μέτρηση 12 Δειγματοληψία ποσόστωσης στις συλλογές 1+2+9 (Χρόνος)

Rib1+Rib2 +Rib9 experiment	#files	REAL TIME (sec)	VAR (%)			ESTIMATED (sec)			WEP TIME (sec)		
			1%	5%	10%	1%	5%	10%	1%	5%	10%
1	1000	3821	0,4	5,7	2,2	3837	3601	3734	37	93	138
2	1000	3821	10,0	4,9	4,1	4205	4012	3981	41	99	148
3	1000	3821	3,1	2,8	13,0	3943	3713	4319	39	93	145

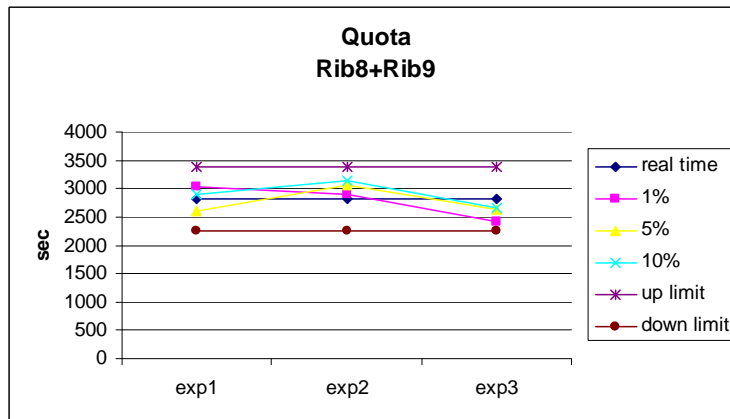
Γραφική παράσταση 9 Δειγματοληψία ποσόστωσης στις συλλογές 1+9 (Χρόνος)



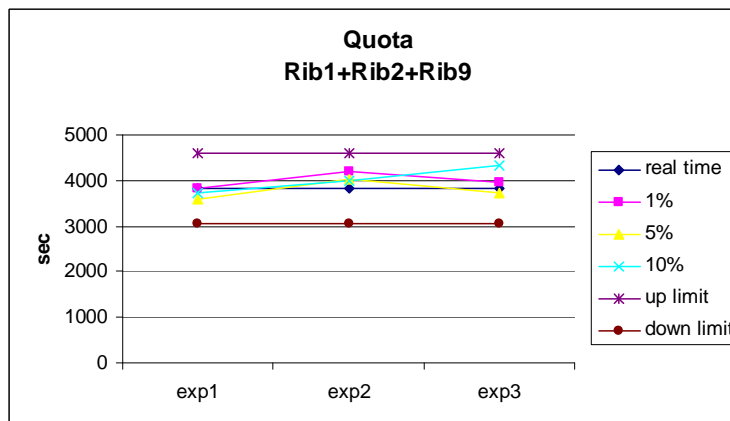
Γραφική παράσταση 10 Δειγματοληψία ποσόστωσης στις συλλογές 2+8 (Χρόνος)



Γραφική παράσταση 11 Δειγματοληψία ποσόστωσης στις συλλογές 8+9 (Χρόνος)

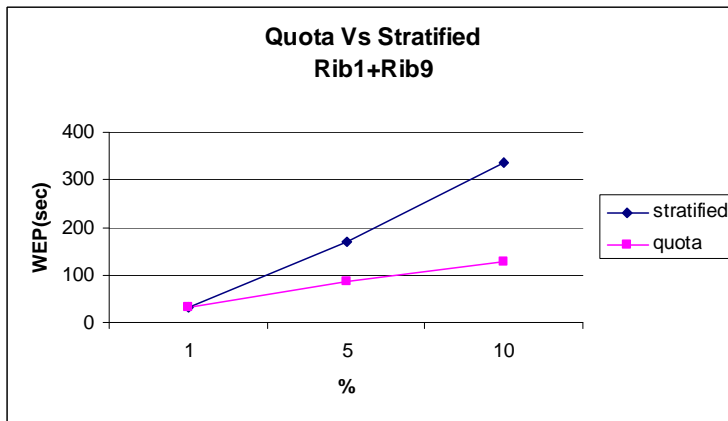


Γραφική παράσταση 12 Δειγματοληψία ποσόστωσης στις συλλογές 1+2+9 (Χρόνος)

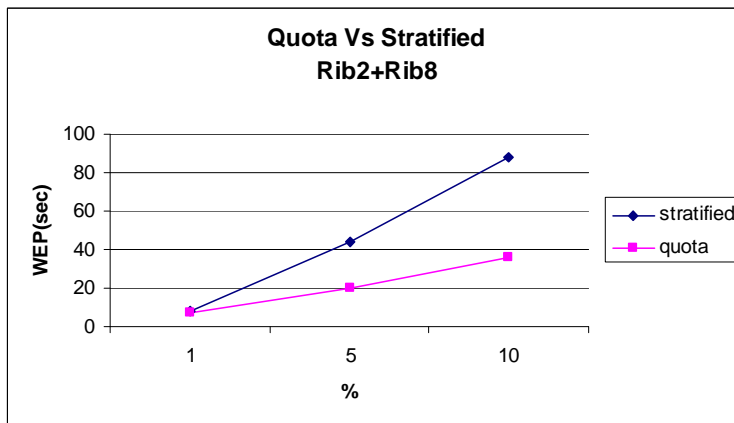


Από τις παραπάνω γραφικές παραστάσεις παρατηρούμε ότι η δειγματοληψία ποσόστωσης δίνει σωστά αποτελέσματα όπως και η στρωματοποιημένη. Όμως στην περίπτωση που είχαμε αρχεία με ίδιο μέγεθος και διαφορετική πολυπλοκότητα τα μικρά ποσοστά δεν εξασφαλίζουν την σωστή εύρεση της αναλογίας των αρχείων μέσα σε ολόκληρη την εργασία. Συγκρίνοντας όμως τους χρόνους εκτέλεσης του ΕΦΕ (WEP time), στις παρακάτω γραφικές παραστάσεις, βλέπουμε ότι στην στρωματοποιημένη δειγματοληψία ο χρόνος αυξάνεται ανάλογα με το ποσοστό του δείγματος ενώ στην δειγματοληψία ποσόστωσης ο χρόνος WEP έχει πολύ μικρότερο ρυθμό αύξησης. Έτσι στην δειγματοληψία ποσόστωσης μπορούμε να μεγαλώσουμε το δείγμα που θα εκτελεστεί στην ανάλυση 1x1 χωρίς να μεγαλώσει πολύ ο χρόνος εκτέλεσης του ΕΦΕ. Κοιτάζοντας λίγο πιο προσεκτικά τις παρακάτω γραφικές παραστάσεις θα δούμε ότι η στρωματοποιημένη έχει μεγαλύτερη ακρίβεια αλλά το κέρδος χρόνου που έχουμε με την ποσόστωση είναι πολύ σημαντικότερο αφού και στην δειγματοληψία ποσόστωσης έχουμε επιτυχημένες εκτιμήσεις μέσα στα αποδεκτά όρια.

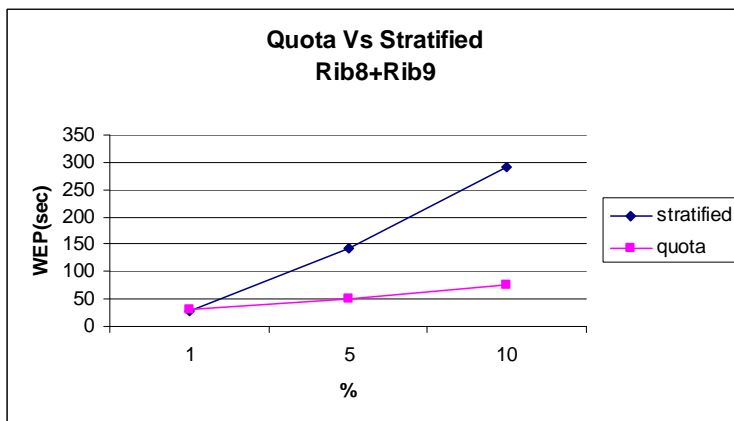
Γραφική παράσταση 13 Ποσόστωσης εναντίον στρωματοποιημένης στις συλλογές 1+9



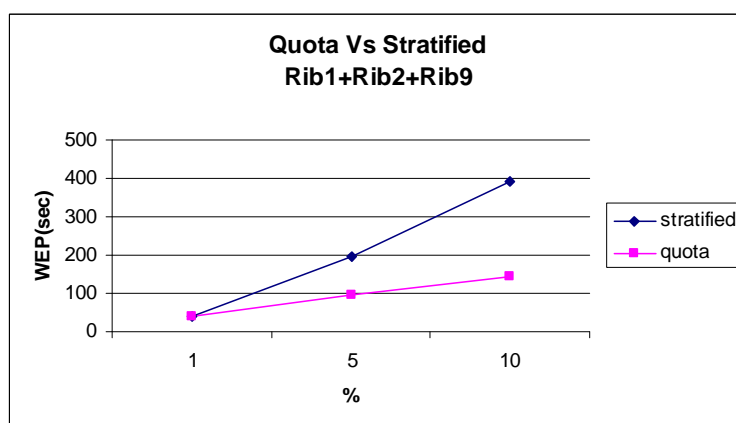
Γραφική παράσταση 14 Ποσόστωσης εναντίον στρωματοποιημένης στις συλλογές 2+8



Γραφική παράσταση 15 Ποσόστωσης εναντίον στρωματοποιημένης στις συλλογές 8+9



Γραφική παράσταση 16 Ποσόστωσης εναντίον στρωματοποιημένης στις συλλογές 1+2+9



6.4.2 Μετρήσεις μνήμης

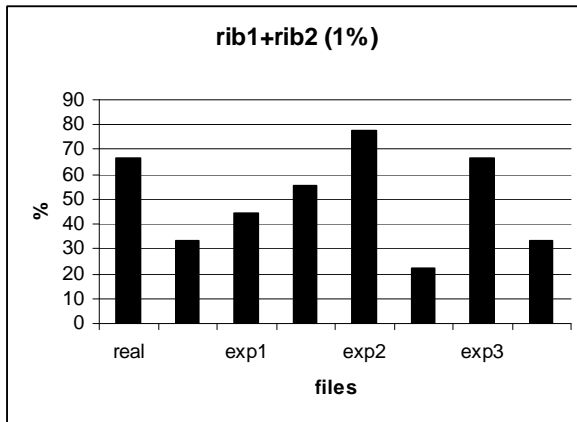
Η παράμετρος που θα αξιολογήσει τις μεθόδους δειγματοληψίας είναι η εύρεση της σωστής αναλογίας των αρχείων στον συνολικό πληθυσμό διαχωρισμένα με βάση τον φόρτο τους. Στις συγκεκριμένες μετρήσεις ο φόρτος χαρακτηρίζεται μόνο από την μέγιστη μνήμη που χρησιμοποιούν τα αρχεία. Αρχεία με ίδιο φόρτο είναι αυτά που η απόκλιση της μνήμης τους δεν ξεπερνάει το 20%. Για παράδειγμα έστω ότι έχουμε μία εργασία 100 αρχείων από τα οποία τα 50 χρησιμοποιούν 5MB μνήμης και τα άλλα 50 χρησιμοποιούν 10MB μνήμης. Η σωστή αναλογία τους είναι 50-50%. Αν μετά την εκτέλεση ενός δείγματος 10% βρεθούν 8 αρχεία των 5 MB και 2 των 10MB η εκτιμώμενη αναλογία είναι 80-20% η οποία είναι αποτυχημένη.

6.4.2.1 Μετρήσεις στην τυχαία δειγματοληψία

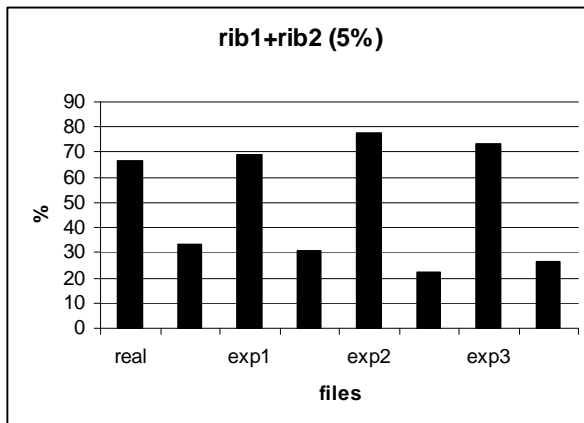
Είναι φανερό ότι για να μετρήσουμε την αξιοπιστία της τυχαίας δειγματοληψίας θα συνδυάσουμε συλλογές που έχουν διαφορετική χρήση μνήμης. Οι μετρήσεις μας θα γίνουν στους συνδυασμούς συλλογών 1-2, 3-4-9 και 1-2-4-7. Στην Γραφική παράσταση 17 φαίνονται στις δύο πρώτες στήλες οι πραγματικές αναλογίες της εργασίας 1-2 όπου το 66,66% είναι αρχεία από την συλλογή Rib1 με απαιτούμενη μνήμη 4,7 MB ενώ το 33,33% είναι από την συλλογή Rib2 με απαιτούμενη μνήμη 12,7 MB. Στα τρία πειράματα που έγιναν με ποσοστό 1% οι αναλογίες που βρήκε ο ΕΦΕ με βάση το δείγμα που διάλεξε ήταν: στο πρώτο πείραμα (exp1) 44,44% αρχεία με μνήμη 4,7 MB και 55,56% αρχεία με μνήμη 12,7 MB, στο δεύτερο πείραμα (exp2) 77,77% και 22,23% αντίστοιχα και στο τρίτο (exp3) βρήκε την σωστή αναλογία. Στην Γραφική παράσταση 18 επαναλάβαμε το ίδιο πείραμα με ποσοστό 5%. Παρατηρούμε ότι στο πρώτο και στο τρίτο πείραμα η εκτίμηση είναι πολύ κοντά στις πραγματικές τιμές. Τέλος στην Γραφική παράσταση 19 έχουμε σωστή εκτίμηση και στα τρία πειράματα αλλά το δείγμα που εκτελέστηκε (10%) έχει αρχίσει να μεγαλώνει

αρκετά. Παρόμοια αποτελέσματα πήραμε και από τους άλλους συνδυασμούς συλλογών των οποίων απλά παραθέτουμε τις γραφικές παραστάσεις.

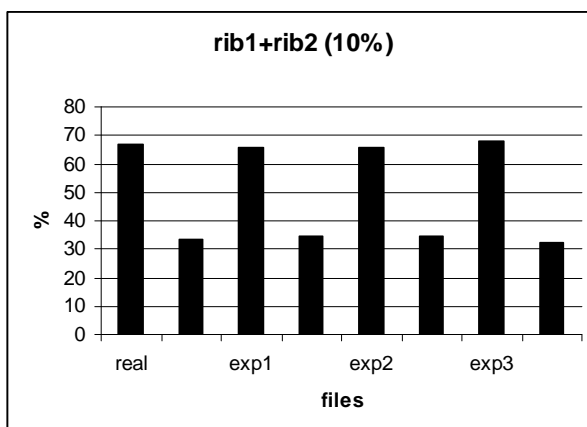
Γραφική παράσταση 17 Τυχαία δειγματοληψία 1% στις συλλογές 1+2 (Μνήμη)



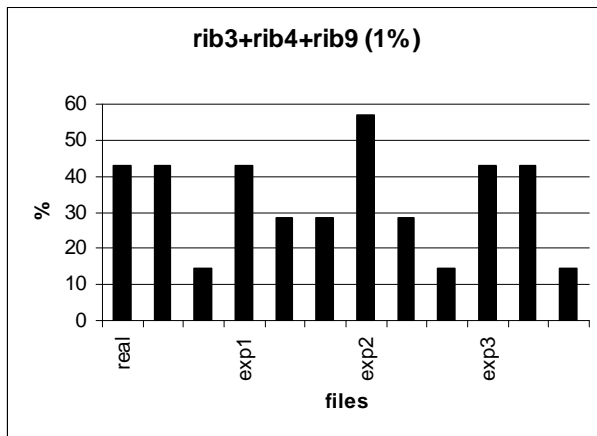
Γραφική παράσταση 18 Τυχαία δειγματοληψία 5% στις συλλογές 1+2 (Μνήμη)



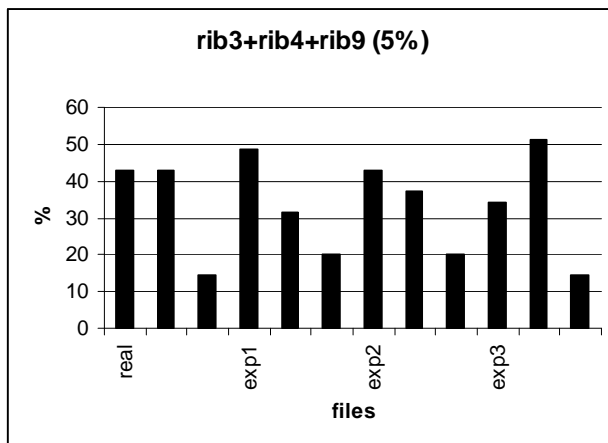
Γραφική παράσταση 19 Τυχαία δειγματοληψία 10% στις συλλογές 1+2 (Μνήμη)



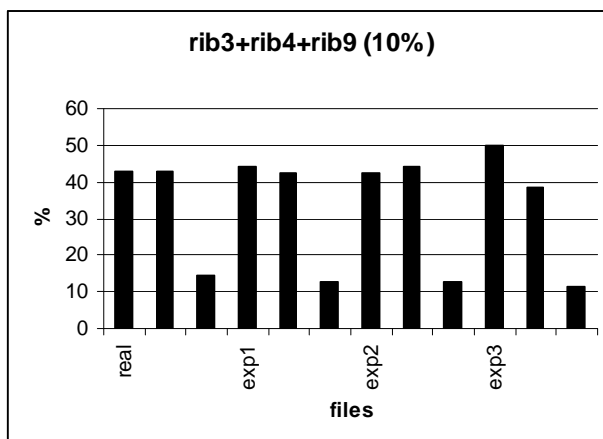
Γραφική παράσταση 20 Τυχαία δειγματοληψία 1% στις συλλογές 3+4+9(Μνήμη)



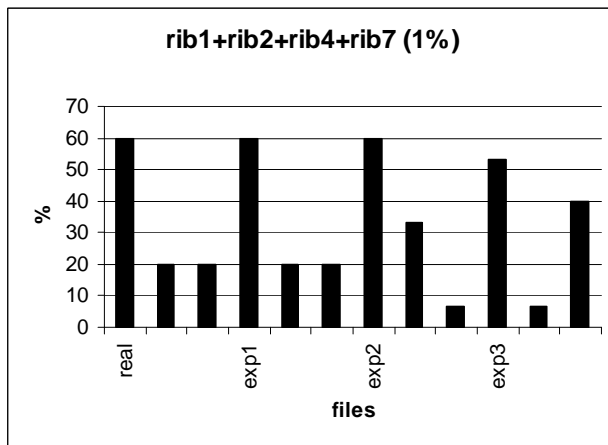
Γραφική παράσταση 21 Τυχαία δειγματοληψία 5% στις συλλογές 3+4+9(Μνήμη)



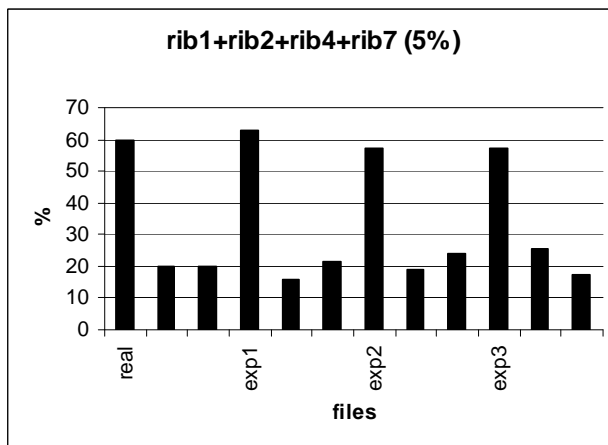
Γραφική παράσταση 22 Τυχαία δειγματοληψία 10% στις συλλογές 3+4+9(Μνήμη)



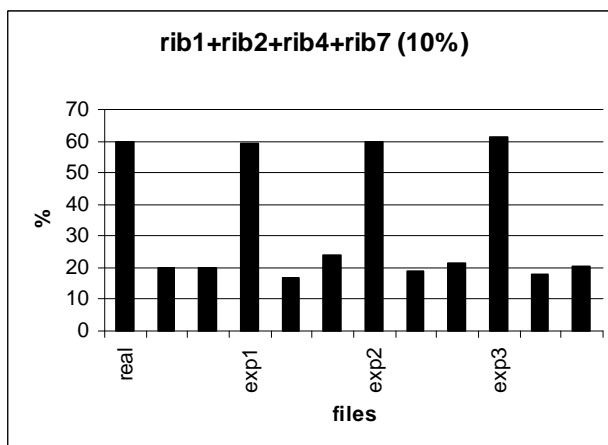
Γραφική παράσταση 23 Τυχαία δειγματοληψία 1% στις συλλογές 1+2+4+7(Μνήμη)



Γραφική παράσταση 24 Τυχαία δειγματοληψία 5% στις συλλογές 1+2+4+7(Μνήμη)



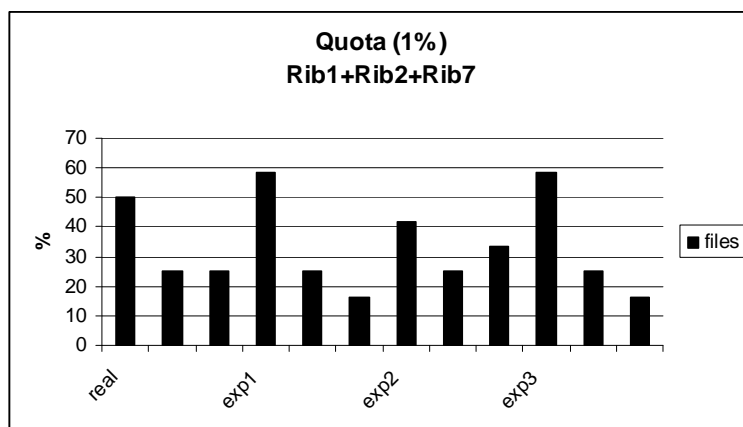
Γραφική παράσταση 25 Τυχαία δειγματοληψία 10% στις συλλογές 1+2+4+7(Μνήμη)



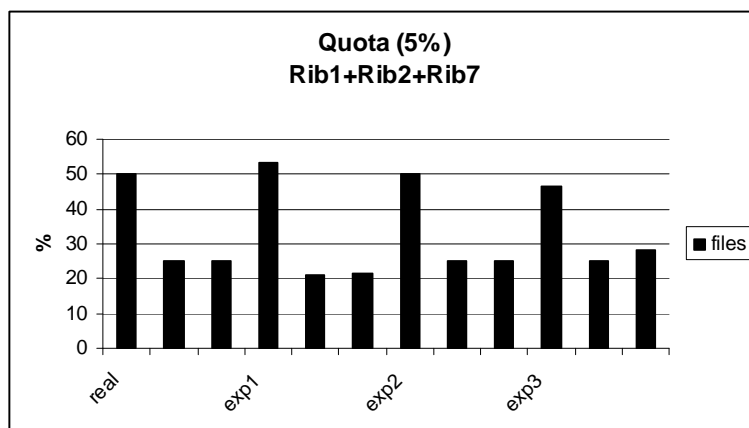
6.4.2.2 Μετρήσεις στην στρωματοποιημένη δειγματοληψία και ποσόστωσης

Τις δύο μεθόδους αυτές θα τις εξετάσουμε μαζί γιατί έχουν την ίδια πιθανότητα επιτυχημένης εκτίμησης. Η διαφορά τους είναι ότι η δειγματοληψία ποσόστωσης εκτελείται πιο γρήγορα. Αυτό συμβαίνει γιατί η στρωματοποιημένη εκτελεί ένα δείγμα από κάθε στρώμα και βρίσκει τις αναλογίες ενώ η ποσόστωσης εκτελεί τα αρχεία σε ανάλυση 1x1 που είναι πολύ πιο γρήγορη. Είναι προφανές ότι για να δυσκολέψουμε την εκτίμηση θα εφαρμόσουμε τις δειγματοληψίες σε συλλογές που έχουν αρχεία με ίδιο μέγεθος αλλά διαφορετική χρησιμοποίηση μνήμης. Δύο τέτοιοι συνδυασμοί συλλογών είναι η Rib1-2-7 και η Rib3-4-7. Με τον ίδιο τρόπο αναπαράστασης όπως και πριν θα δείξουμε τις μετρήσεις. Σε κάθε γραφική παράσταση οι τρεις πρώτες στήλες είναι οι πραγματικές αναλογίες της κάθε εργασίας ενώ ανά τρεις είναι οι αναλογίες που βρήκε η δειγματοληψία σε κάθε επανάληψη του πειράματος. Αυτό που παρατηρούμε είναι ότι ένα δείγμα 5% είναι αρκετά πιθανό να δώσει σωστά αποτελέσματα αλλά οι μετρήσεις έχουν μεγαλύτερη αξιοπιστία όταν το δείγμα είναι 10%. Άρα ένα δείγμα μεγαλύτερο του 10% θα δίνει πολύ αξιόπιστες μετρήσεις.

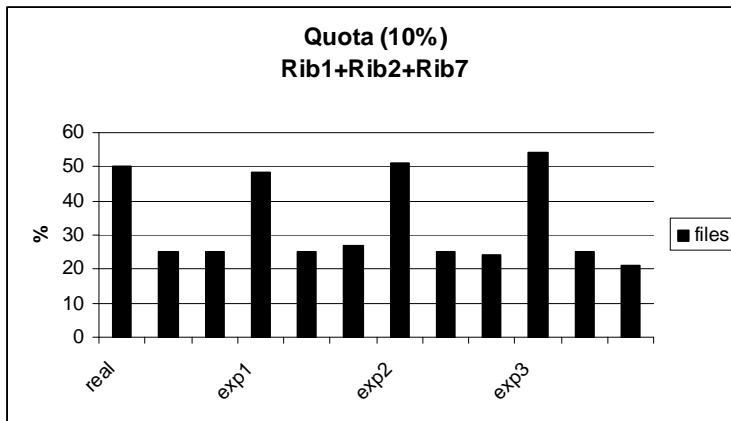
Γραφική παράσταση 26 Δειγματοληψία ποσόστωσης 1% στις συλλογές 1+2+7(Μνήμη)



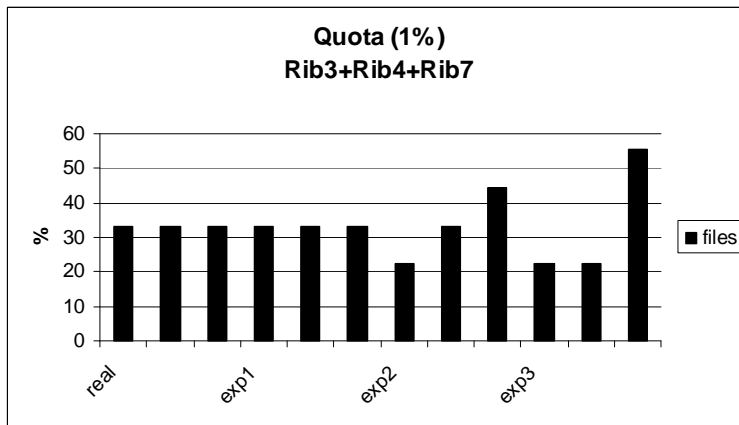
Γραφική παράσταση 27 Δειγματοληψία ποσόστωσης 5% στις συλλογές 1+2+7(Μνήμη)



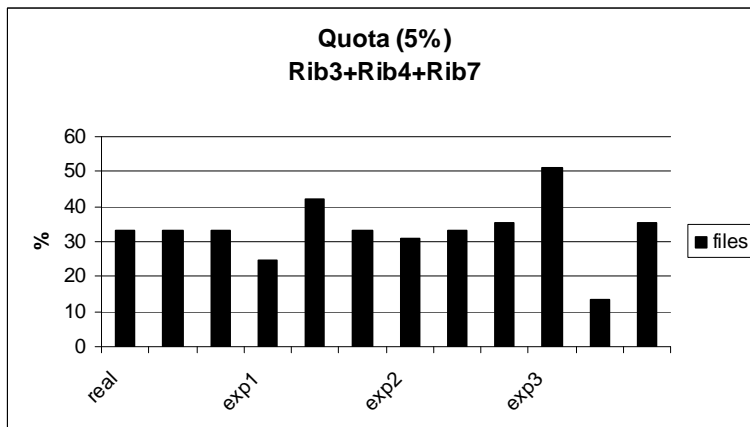
Γραφική παράσταση 28 Δειγματοληψία ποσόστωσης 10% στις συλλογές 1+2+7(Μνήμη)



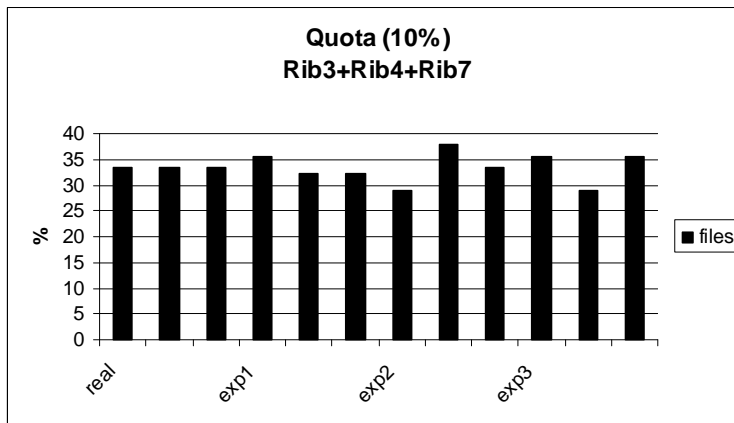
Γραφική παράσταση 29 Δειγματοληψία ποσόστωσης 1% στις συλλογές 3+4+7(Μνήμη)



Γραφική παράσταση 30 Δειγματοληψία ποσόστωσης 5% στις συλλογές 3+4+7(Μνήμη)



Γραφική παράσταση 31 Δειγματοληψία ποσοστώς 10% στις συλλογές 3+4+7(Μνήμη)



6.4.3 Μετρήσεις αποθηκευτικού χώρου

Ο χώρος που καταλαμβάνει μία εικόνα στον δίσκο εξαρτάται από την ανάλυσή της και από το format που αποθηκεύεται. Σε μερικά format ο χώρος είναι ανάλογος της ανάλυσης της εικόνας ενώ σε άλλα εξαρτάται και από την συμπίεση που κάνει το format. Η συμπίεση δεν είναι απαραίτητα ανάλογη της ανάλυσης και εξαρτάται και από την πολυπλοκότητα της εικόνας. Για αυτό θα πραγματοποιήσουμε μετρήσεις σε εικόνες με διαφορετική ανάλυση και κωδικοποίηση. Από τα πιο συνηθισμένα format που αποθηκεύεται μια εικόνα είναι το tif και το jpg. Το tif είναι format που είναι ανάλογο της ανάλυσης της εικόνας. Το jpg είναι format που κάνει μεγάλη συμπίεση στην εικόνα και εξαρτάται και από την πολυπλοκότητά της.

6.4.3.1 Μετρήσεις σε format tif

Για να μετρήσουμε τον χώρο που καταλαμβάνει μια εικόνα tif θα δημιουργήσουμε εικόνες με διαφορετική ανάλυση και θα καταγράψουμε το μέγεθός τους. Στον παρακάτω πίνακα φαίνονται οι μετρήσεις για μία εικόνα με αναλύσεις 320x240, 640x480, 1280x960, 2560x1920.

Πίνακας 4 Αντιστοιχία ανάλυσης εικόνας και μεγέθους αρχείου σε κωδικοποίηση tif.

Resolution	320x240	640x480	1280x960	2560x1920
Size (Kbytes)	233	925	3690	14750

Από τον πίνακα βλέπουμε ότι για κάθε διπλασιασμό της ανάλυσης τετραπλασιάζεται το μέγεθος στον δίσκο. Για να βρούμε πόσο χώρο χρειάζεται το κάθε pixel θα κάνουμε τον εξής συλλογισμό. Στην ανάλυση 1280x960 η εικόνα περιέχει 1.228.800 pixels. Άρα το ένα pixel χρειάζεται 3 bytes. Οπότε για παράδειγμα μία εικόνα με ανάλυση 1024x768 χρειάζεται περίπου 2,36MB. Για την επαλήθευση του συλλογισμού μας θα κάνουμε μετρήσεις σε

διάφορες τυχαίες αναλύσεις και θα συγκρίνουμε το πραγματικό μέγεθος με το εκτιμώμενο. Τα πειράματα αυτά γίνανε κατ' επανάληψη σε διάφορες εικόνες με αμελητέα απόκλιση από την μία μέτρηση που παρουσιάζουμε για οικονομία χώρου. Ο συλλογισμός μας ήταν σωστός καθώς βλέπουμε ότι τα αποτελέσματα έχουν πολύ μικρή απόκλιση.

Μέτρηση 13 Σύγκριση εκτιμώμενου και πραγματικού μεγέθους αρχείο σε κωδικοποίηση tif.

Resolution	600x400	800x600	1024x768	1280x768
Size (Kbytes)	720	1441	2363	2953
Predictor Size(Kbytes)	720	1440	2360	2950

6.4.3.2 Μετρήσεις σε format jpg

Το jpg είναι format που συμπιέζει την εικόνα και δεν είναι δεδομένο ότι δύο εικόνες με την ίδια ανάλυση θα καταλαμβάνουν τον ίδιο χώρο στο δίσκο όπως συμβαίνει στο tif. Εκτελέσαμε κάθε συλλογή σε format jpg και παρατηρήσαμε ότι οι παραγόμενες εικόνες είχαν το ίδιο μέγεθος σε κάθε συλλογή. Αυτό είναι λογικό αφού στις συλλογές που χρησιμοποιήσαμε στα πειράματά μας τα αρχεία κάθε συλλογής έχουν την ίδια πολυπλοκότητα. Έτσι αν η μέθοδος που θα επιλέξει το δείγμα κάνει σωστή εκτίμηση και διαλέξει τις σωστές αναλογίες από κάθε στρώμα θα μπορεί ο ΕΦΕ να υπολογίσει τον απαιτούμενο αποθηκευτικό χώρο που χρειάζεται. Άρα η μέτρηση της αξιοπιστίας των δειγματοληπτικών μεθόδων στον αποθηκευτικό χώρο είναι όμοια με τις μετρήσεις της μνήμης. Για αυτό δεν κάναμε επιπλέον μετρήσεις. Με ανάλογο τρόπο μπορούν να γίνουν μετρήσεις και σε άλλα format.

6.4.4 Μετρήσεις στην ανάλυση της εικόνας

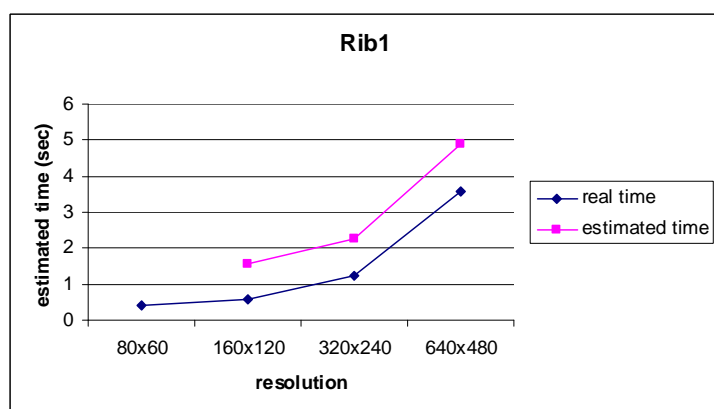
Στο προηγούμενο κεφάλαιο περιγράψαμε μια τεχνική την μείωση της ανάλυσης της εικόνας. Προτού όμως χρησιμοποιήσουμε αυτή την τεχνική για να επιτύχουμε μείωση του χρόνου εκτέλεσης του ΕΦΕ καλό θα ήταν να γίνουν και κάποια πειράματα ώστε να επαληθεύσουμε αν ισχύει ο κανόνας ότι ο διπλασιασμός της ανάλυσης τετραπλασιάζει τον χρόνο εκτέλεσης. Επίσης αναφέραμε ότι ο κανόνας αυτός έχει ελάχιστο όριο την ανάλυση εικόνας 640x480.

Για αυτό από κάθε συλλογή εκτελέσαμε ένα αρχείο στις αναλύσεις 80x60, 160x120, 320x240, 640x480, 1280x960, 2560x1920, 5120x3840 και πήραμε την εξής μέτρηση.

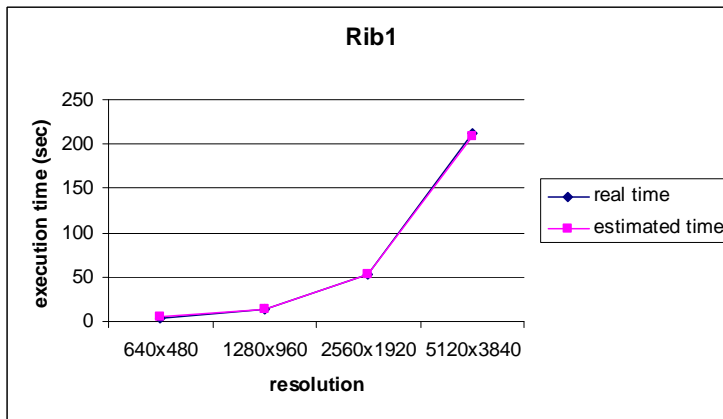
Μέτρηση 14 Σχέση ανάλυσης εικόνας-εκτελέσιμου χρόνου

RESOLUTION	EXECUTION TIME (sec)						
	Rib1	Rib2	Rib3	Rib4	Rib7	Rib8	Rib9
80x60	0,391	0,594	0,406	0,438	0,437	0,406	7,406
160x120	0,562	0,875	0,672	0,578	0,579	0,485	13,578
320x240	1,219	1,984	1,578	1,172	1,281	0,984	30,797
640x480	3,578	6,109	5,000	3,594	4,203	2,891	87,016
1280x960	13,360	22,313	18,985	13,359	15,672	10,640	284,313
2560x1920	52,421	86,531	74,484	52,547	61,781	41,469	1008,578
5120x3840	212,016	347,672	300,125	212,625	249,187	169,203	3807,469

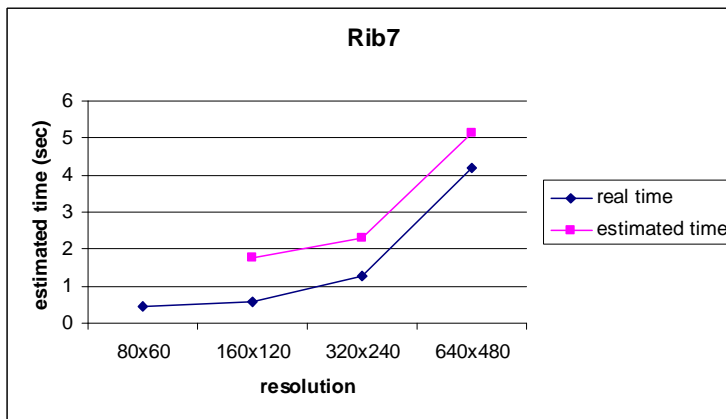
Παρατηρούμε ότι στις αναλύσεις μετά την 640x480 ισχύει ο νόμος του τετραπλασιασμού του χρόνου για κάθε διπλασιασμό της ανάλυσης. Για να γίνει πιο εύκολα αντιληπτό παραθέτουμε μερικές γραφικές παραστάσεις από τις πιο πάνω μετρήσεις. Στην μία καμπύλη φαίνεται ο πραγματικά μετρούμενος χρόνος εκτέλεσης ενώ στην δεύτερη φαίνεται ο εκτιμώμενος χρόνος. Ο υπολογισμός του εκτιμώμενου χρόνου για κάθε ανάλυση γίνεται με τον πολλαπλασιασμό του πραγματικού χρόνου της προηγούμενης ανάλυσης επί 4. Για να φανεί πιο έντονη η διαφορά εμφανίζουμε ξεχωριστά τις μετρήσεις για τις αναλύσεις 80x60 ως την 640x480 και τις αναλύσεις 640x480 ως την 5120x3840. Είναι ολοφάνερο ότι από την 640x480 και μετά οι τιμές σχεδόν ταυτίζονται με τις πραγματικές ενώ στις μικρότερες αναλύσεις ο εκτιμώμενος χρόνος είναι αρκετά μεγαλύτερος από τον πραγματικό.

Γραφική παράσταση 32 Σύγκριση εκτιμώμενου –πραγματικού χρόνου σε μικρές αναλύσεις

Γραφική παράσταση 33 Σύγκριση εκτιμώμενου –πραγματικού χρόνου σε μεγάλες αναλύσεις



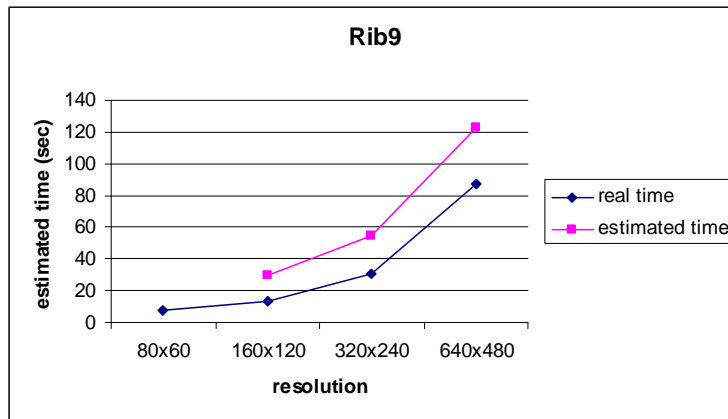
Γραφική παράσταση 34 Σύγκριση εκτιμώμενου –πραγματικού χρόνου σε μικρές αναλύσεις



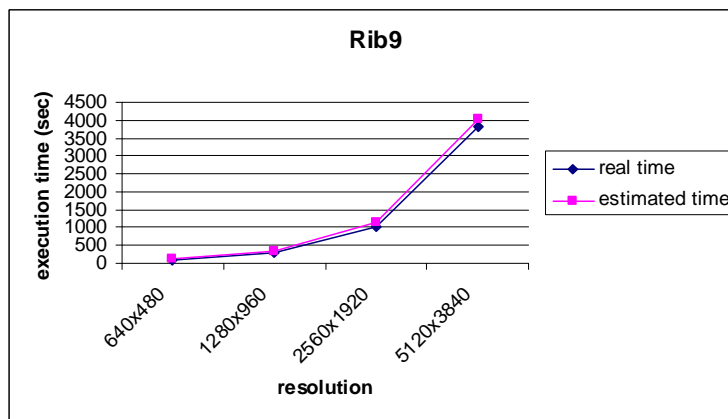
Γραφική παράσταση 35 Σύγκριση εκτιμώμενου –πραγματικού χρόνου σε μεγάλες αναλύσεις



Γραφική παράσταση 36 Σύγκριση εκτιμώμενου –πραγματικού χρόνου σε μικρές αναλύσεις



Γραφική παράσταση 37 Σύγκριση εκτιμώμενου –πραγματικού χρόνου σε μεγάλες αναλύσεις



6.5 Σύνοψη συμπερασμάτων αξιολόγησης

Αφού ολοκληρώσαμε τις μετρήσεις και αξιοποιήσαμε τις μεθόδους, είμαστε σε θέση να εξάγουμε τα συμπεράσματα για την συμπεριφορά των μεθόδων. Τα συμπεράσματα αυτά θα μας βοηθήσουν στην καλύτερη σχεδίαση ενός εκτιμητή φόρτου εργασίας για την εφαρμογή του 3D Rendering. Ολόκληρος ο σχεδιασμός που έγινε στο κεφάλαιο 5.3.3 βασίστηκε στα αποτελέσματα των μετρήσεων αυτού του κεφαλαίου.

Από τις μεθόδους δειγματοληψίας που εφαρμόσαμε παρατηρήσαμε ότι η τυχαία δειγματοληψία είναι αναξιόπιστη και αποτυγχάνει πολλές φορές στην εκτίμηση ειδικά όταν το ποσοστό του δείγματος είναι μικρό της τάξεως του 1 με 5%. Σε ποσοστά μεγαλύτερα του 10% έχει αρκετές πιθανότητες σωστής εκτίμησης αλλά δεν μπορεί να αντεπεξέλθει σε πιο σύνθετες και πολύπλοκες εργασίες. Το μειονέκτημα με τα μεγάλα ποσοστά είναι ότι αυξάνεται πάρα πολύ ο χρόνος εκτέλεσης του ΕΦΕ ο οποίος είναι μία παράμετρος που θέλουμε να την μειώσουμε στο ελάχιστο δυνατόν. Επίσης η τυχαία δειγματοληψία δεν μπορεί να δώσει πολλές πληροφορίες για την βαρύτητα των αρχείων και να τα κατηγοριοποιήσει

ανάλογα με τον φόρτο τους ακόμα και αν στο δείγμα έχουν συμπεριληφθεί όλες οι κατηγορίες πληροφορίες που είναι χρήσιμες για τον χρονοδρομολογητή του πλέγματος.

Η επόμενη δειγματοληψία που μελετήσαμε ήταν η στρωματοποιημένη δειγματοληψία. Η εφαρμογή αυτής της μεθόδου μας επιβεβαίωσε ότι σε μια εργασία τα αρχεία με ίδιο μέγεθος έχουν πολλές πιθανότητες να έχουν την ίδια πολυπλοκότητα. Τα πλεονεκτήματά της είναι ότι δίνει ακριβείς εκτιμήσεις ακόμα και στα μικρά ποσοστά και μπορεί να δώσει πληροφορίες για τον φόρτο των αρχείων αφού λαμβάνει αποτελέσματα για το κάθε στρώμα ξεχωριστά. Ειδικότερα στην περίπτωση που όλα τα δείγματα του στρώματος έχουν την ίδια πολυπλοκότητα αυξάνεται η εγκυρότητα της πληροφορίας. Το μειονέκτημά της είναι ότι στην περίπτωση που σε ένα στρώμα υπάρχουν αρχεία με διαφορετική πολυπλοκότητα τα μικρά ποσοστά δείγματος την καθιστούν αναξιόπιστη όπως και την τυχαία. Το γεγονός αυτό σε συνδυασμό με το ότι ο χρόνος εκτέλεσης του ΕΦΕ αυξάνεται αναλογικά με την αύξηση του δείγματος περιορίζει την ευελιξία της μεθόδου και την κάνει είτε χρονοβόρα αλλά ακριβή στις εκτιμήσεις είτε γρήγορη αλλά αναξιόπιστη.

Γενικότερα οι έννοιες ταχύτητα και ακρίβεια είναι εντελώς αντίθετες στην περίπτωσή μας. Όσο θέλουμε γρηγορότερες εκτιμήσεις τόσο θα χάνουμε σε ακρίβεια ενώ εκτιμήσεις με μεγάλη ακρίβεια είναι εφικτές αλλά με μεγάλο χρονικό κόστος. Η χρυσή τομή των δύο αυτών παραμέτρων μπορεί να επιτευχθεί με την δειγματοληψία ποσόστωσης. Το μεγάλο πλεονέκτημα της μεθόδου αυτής είναι ότι ο χρόνος εκτέλεσης του ΕΦΕ δεν μεγαλώνει αναλογικά με την αύξηση του δείγματος όπως συμβαίνει στην στρωματοποιημένη αλλά με πολύ μικρότερο ρυθμό. Αυτό την κάνει πιο ευέλικτη από την στρωματοποιημένη καθώς μπορεί να αντεπεξέλθει και στην περίπτωση που σε ένα στρώμα υπάρχουν αρχεία με διαφορετική πολυπλοκότητα χωρίς να αυξηθεί πολύ ο χρόνος εκτέλεσης του ΕΦΕ. Επίσης μπορεί να δώσει πληροφορίες για τον φόρτο των αρχείων αφού λαμβάνει αποτελέσματα για το κάθε στρώμα ξεχωριστά και δίνει αξιόπιστα αποτελέσματα. Το μειονέκτημά της είναι ότι χάνει λίγο στην ακρίβεια σε σχέση με την στρωματοποιημένη αλλά μπροστά στο μεγάλο χρονικό κέρδος που εξασφαλίζει η απόκλιση αυτή είναι αμελητέα καθώς οι εκτιμήσεις παραμένουν μέσα στα αποδεκτά όρια.

Έτσι καταλήγουμε στο συμπέρασμα ότι η χρυσή τομή ταχύτητας και ακρίβειας στις εκτιμήσεις του φόρτου είναι η δειγματοληψία ποσόστωσης. Ένα δείγμα της τάξεως του 5% είναι ικανό να δώσει ικανοποιητικά αποτελέσματα και σε μικρό χρόνο. Το 5% είναι ένα ποσοστό καλό για την εκτέλεση αρχείων σε ανάλυση 1x1 αλλά είναι μεγάλο για την εκτέλεσή τους στην κανονική ανάλυση. Όλοι αυτοί οι λόγοι μας προτρέπουν να σχεδιάσουμε έναν ΕΦΕ βασισμένο στην δειγματοληψία ποσόστωσης.

Πέρα από τις μεθόδους δειγματοληψίας μελετήσαμε και την τεχνική της μείωσης της ανάλυσης της εικόνας. Πειραματικά επιβεβαιώσαμε ότι ο διπλασιασμός της ανάλυσης μίας

εικόνας τετραπλασιάζει τον χρόνο εκτέλεσης του αρχείου `tib`. Αυτό συμβαίνει για αναλύσεις μεγαλύτερες από την 640x480. Η τεχνική αυτή είναι ένα πολύ δυνατό εργαλείο καθώς μπορεί να μειώσει πάρα πολύ τον χρόνο WEP στις περιπτώσεις που μια εργασία έχει απαιτήσεις υψηλής ανάλυσης χρησιμοποιώντας ανάποδα την τεχνική αυτή. Δηλαδή για κάθε υποδιπλασιασμό της ανάλυσης θα υποτετραπλασιάζεται ο χρόνος εκτέλεσης. Βέβαια πάλι χάνουμε λίγο στην ακρίβεια αλλά το κέρδος χρόνου είναι πολύ μεγάλο.

Τελικά τα οφέλη και οι πληροφορίες που μπορούμε να αποκομίσουμε με την δειγματοληψία ποσόστωσης και τις τεχνικές μας είναι τα εξής:

- Εύρεση του χρόνου διεκπεραίωσης της εργασίας. Αυτή είναι και η σημαντικότερη παράμετρος για τον προσδιορισμό του φόρτου εργασίας.
- Εκτίμηση σε πολύ μικρό χρόνο. Παράμετρος που θέλουμε να ελαχιστοποιείται στο μέγιστο δυνατό. Με την τεχνική της μείωσης της ανάλυσης η απαίτηση αυτή μπορεί να ικανοποιηθεί και σε εργασίες με υψηλή ανάλυση εικόνας.
- Ακριβής γνώση του φόρτου εργασίας για τα στρώματα που δεν περιέχουν υποστρώματα. Πολλή χρήσιμη πληροφορία για τον χρονοδρομολογητή του πλέγματος καθώς θα γνωρίζει επακριβώς τις απαιτήσεις του στρώματος σε υπολογιστική ισχύ. Τα βαρύτερα στρώματα μπορούν να χρονοδρομολογούνται σε ισχυρότερους υπολογιστές.
- Σχετική γνώση του φόρτου εργασίας για τα στρώματα που περιέχουν υποστρώματα. Η διαφορά με την προηγούμενη περίπτωση είναι ο ΕΦΕ δεν μπορεί να προσδιορίσει τα αρχεία των υποστρωμάτων. Απλά γνωρίζει ότι σε ένα στρώμα υπάρχουν αρχεία με διαφορετική πολυπλοκότητα και τις αναλογίες τους. Όμως και αυτή η πληροφορία μπορεί να αξιοποιηθεί από τον χρονοδρομολογητή αφού θα έχει μία γενικότερη εικόνα για το στρώμα.
- Ειδικότερα ο αποθηκευτικός χώρος είναι το μόνο μέγεθος που σε κάποιες περιπτώσεις μπορούμε να έχουμε γνώση χωρίς την εκτέλεση των αρχείων. Οι περιπτώσεις αυτές είναι όταν το `format` της παραγόμενης εικόνας ανήκει στην κατηγορία των `format` που εξαρτώνται μόνο από την ανάλυση της εικόνας όπως είναι το `tif` και το `bmp`. Η γνώση της ανάλυσης προσδιορίζει και το μέγεθος του αρχείου που αποθηκεύεται η εικόνα.

7

Τεχνικές λεπτομέρειες

Στο παρόν κεφάλαιο θα περιγράψουμε ορισμένα τμήματα που έχουν τεχνικό ενδιαφέρον από τα προγράμματα με τα οποία υλοποιήσαμε τις μεθόδους δειγματοληψίας και κάναμε τις μετρήσεις. Επίσης θα δούμε ποιες γλώσσες προγραμματισμού χρησιμοποιήσαμε και γιατί καθώς και την περιγραφή της πλατφόρμας ανάπτυξης και των προγραμματιστικών εργαλείων.

7.1 Λεπτομέρειες υλοποίησης

Ένα από τα πιο ενδιαφέροντα τεχνικά μέρη είναι αυτό της εκτέλεσης του `air` στο παρασκήνιο και της μέτρησης του χρόνου εκτέλεσής του μέσα από τα δικά μας προγράμματα. Θα δούμε πως μπορεί να εκτελεστεί ένα εξωτερικό πρόγραμμα μέσα από ένα πρόγραμμα Java καθώς και με ποιους τρόπους επικοινωνεί με το δικό μας πρόγραμμα. Επίσης ιδιαίτερο ενδιαφέρον παρουσιάζει και ο τρόπος με τον οποίο δημιουργούμε τα στρώματα και τα υποστρώματα, που ενδεχομένως να περιέχουν τα στρώματα, χρησιμοποιώντας την τεχνική της ανάλυσης `1x1`.

7.1.1 Εκτέλεση του `Air` στο παρασκήνιο

Η μέθοδος `Runtime.exec` δημιουργεί μια φυσική διεργασία και επιστρέφει ένα στιγμιότυπο της υποκλάσης `Process` το οποίο μπορεί να χρησιμοποιηθεί για τον έλεγχο της διεργασίας καθώς και να αποκομίσει πληροφορίες για αυτή. Η μέθοδος `Runtime.exec` δέχεται ως παράμετρο έναν πίνακα τύπου `String` όπου το πρώτο στοιχείο του πίνακα είναι το πρόγραμμα

που θέλουμε να εκτελεστεί ενώ τα υπόλοιπα είναι τα ορίσματα που του περνάμε. Ο Air είναι ένα πρόγραμμα που μπορεί να τρέξει σε command prompt δεχόμενος και ορίσματα. Στη συνέχεια με τις μεθόδους της κλάσης Process getInputStream(), getErrorStream() και waitFor() μπορεί το πρόγραμμά μας να διαβάσει το ρεύμα εξόδου του Air, να διαβάσει το ρεύμα εξόδου λαθών και να περιμένει την ολοκλήρωση της διεργασίας του Air. Αυτό που παρατηρήσαμε ήταν ότι αν δεν διαβάζονται τα ρεύματα εξόδου μπλοκάρει η διεργασία μέχρι που να διαβαστούν. Για αυτό καλό είναι τα ρεύματα εξόδου να διαβάζονται συνέχεια από διαφορετικά νήματα και τα αποτελέσματα να αποθηκεύονται σε ένα αρχείο. Παραθέτουμε ένα μικρό πρόγραμμα που εκτελεί τον Air με τα εξής ορίσματα και μετράει τον χρόνο εκτέλεσής του.

```
air -file c:\temp\outfile.tif -stats -res 320 240 ribFile.rib
```

Προτείνεται ο φάκελος που βρίσκεται το εκτελέσιμο αρχείο του Air να υπάρχει στο path του συστήματος αλλιώς θα πρέπει να δοθεί η πλήρης διαδρομή πχ. C:\Program Files\air\bin.

```
long startTime=0; //ορισμός του χρόνου έναρξης της διεργασίας
long endTime=0; //ορισμός του χρόνου λήξης της διεργασίας
String env[]={ "air", "-file", "c:\\temp\\outFileName.tif", "-stats",
"-res", "320", "240", "ribFile.rib"} //αυτά είναι τα ορίσματα
try{
    Runtime r=Runtime.getRuntime();
    startTime=System.currentTimeMillis(); //χρόνος έναρξης
    Process proc=r.exec(env); //δημιουργία διεργασίας
    InputStream in=proc.getInputStream(); //ρεύμα εισόδου
    InputStream errIn=proc.getErrorStream(); // ρεύμα λαθών
    ReadAndWrite wrIn=new ReadAndWrite(in); //νήμα που διαβάζει το
    ρεύμα και το καταγράφει σε αρχείο
    wrIn.start(); //έναρξη του νήματος
    ReadAndWrite wrErrIn=new ReadAndWrite(errIn); //νήμα που
    διαβάζει το ρεύμα λαθών και το καταγράφει σε αρχείο
    wrErrIn.start(); //έναρξη του νήματος
    proc.waitFor(); // αναμονή μέχρι να τερματίσει ο air
    endTime=System.currentTimeMillis(); //χρόνος λήξης
    while (w.isAlive() || we.isAlive()); //αναμονή να τερματίσουν
    τα νήματα
}
catch(Exception e){System.out.println("cannot create process"+e);}
System.out.println("Χρόνος εκτέλεσης="+ (endTime-startTime));
```

7.1.2 Δημιουργία στρωμάτων

Για τον διαχωρισμό των αρχείων σε στρώματα κατά μέγεθος δημιουργήσαμε την κλάση MyGroup η οποία είναι μια συνδεδεμένη λίστα σε συνδεδεμένες λίστες αρχείων. Ο κατασκευαστής της κλάσης δέχεται ως ορίσματα έναν πίνακα αρχείων (files) και την μέγιστη

απόκλιση του μεγέθους (var) που θεωρούμε ότι τα αρχεία έχουν το ίδιο μέγεθος. Η var παίρνει τιμές από 0 ως 1. Το μηδέν σημαίνει απόκλιση 0% ενώ το 1 σημαίνει απόκλιση 100%. Η κλάση χρησιμοποιεί την μέθοδο `java.util.Arrays.sort(Object []a , Comparator c)` η οποία ταξινομεί τον πίνακα `a` σύμφωνα με τις οδηγίες του `Comparator`. Η μέθοδος `compare` του `interface Comparator` συγκρίνει δύο ορίσματα και επιστρέφει 1,-1 και 0 αν το πρώτο όρισμα είναι μεγαλύτερο, μικρότερο ή ίσο του δεύτερου αντίστοιχα. Ως παράμετρο σύγκρισης ορίσαμε το μέγεθος των αρχείων.

```
public class Comp implements Comparator<File>{
    public int compare(File f1, File f2){
        if (f1.length()>f2.length()) return 1;// αν το μέγεθος
του πρώτου αρχείου είναι μεγαλύτερο του δεύτερου επέστρεψε 1
        else if (f1.length()<f2.length()) return -1;// αλλιώς αν
είναι μικρότερο επέστρεψε -1
        else return 0;// αλλιώς είναι ίσα και επέστρεψε 0
    }
}

import java.io.File;
import java.util.Arrays;
import java.util.LinkedList;

public class MyGroup extends LinkedList<LinkedList<File>>{

    public MyGroup(File[] files, double conf) {
        Arrays.sort(files,new Comp());//ταξιινόμηση των αρχείων
κατά μέγεθος σε αύξουσα σειρά
        File fPointer=files[0];// δείκτης στο μικρότερο αρχείο
        LinkedList<File> listFile=new LinkedList<File>();
//δημιουργία της πρώτης λίστας αρχείων
        listFile.add(files[0]); //πρόσθεσε το πρώτο αρχείο στη
λίστα

        for (int i = 1; i < files.length; i++) {
            if ( almostEqual(fPointer,files[i], conf) ) {
                listFile.add(files[i]);
            }// για κάθε αρχείο σύγκρινε το με τον δείκτη και
αν είναι περίπου ίσο πρόσθεσέ το στη λίστα αλλιώς
            else {
                this.add(listFile);// πρόσθεσε την λίστα
αρχείων στη λίστα
                listFile=new LinkedList<File>();//δημιούργησε
νέα λίστα αρχείων
                listFile.add(files[i]);
                fPointer=files[i];// πρόσθεσε το αρχείο στην
νέα λίστα και όρισέ το νέο δείκτη
            }
        }
    }
}
```

```

    }
    this.add(listFile); //πρόσθεσε την τελευταία λίστα
    αρχείων στη λίστα
}

private boolean almostEqual(File f1, File f2, double var){

    long diff=(f2.length()-f1.length()) ;
    if ( ( diff/(double)f1.length()) <= var) return true;
    else return false; //η μέθοδος αυτή συγκρίνει το μέγεθος
    δύο αρχείων και επιστρέφει αληθές αν είναι περίπου ίσα ή ψευδές αν
    δεν είναι. Περίπου ίσο σημαίνει η απόκλιση του δεύτερου από το
    πρώτο να μην ξεπερνάει την τιμή var.
}
}

```

Έτσι ένα αντικείμενο της κλάσης MyGroup είναι μία λίστα που κάθε στοιχείο της αντιπροσωπεύει ένα στρώμα. Τα στοιχεία της λίστας είναι λίστες αρχείων που τα στοιχεία τους είναι τα αρχεία του κάθε στρώματος.

7.1.3 Δημιουργία υποστρωμάτων

Αφού είδαμε πως υλοποιήθηκε το πρώτο στάδιο στρωματοποίησης θα δούμε πως υλοποιήθηκε και η διαδικασία διαχωρισμού των στρωμάτων σε υποστρώματα. Λόγω του μεγάλου όγκου του κώδικα που απαιτεί αυτή η διαδικασία θα την περιγράψουμε αλγοριθμικά. Καταρχάς δημιουργήσαμε μία βάση δεδομένων με το σχήμα dbstats που περιέχει τον πίνακα stats με τις εξής στήλες. Επειδή ο πίνακας είναι μεγάλος οι στήλες θα παρουσιαστούν σε δύο γραμμές. Έχουμε μία στήλη για κάθε στατιστικό που παρέχει ο Air στο τέλος της εκτέλεσης όπως φαίνεται και στην Εικόνα 14.

Πίνακας 5 Βάση δεδομένων- πίνακας stats του σχήματος dbstats

stats							
fileName	primitives	polygons	polygonsK	shadingStructures	shadingK	shaderInstances	memory_use
memory_peak		hits	misses	rayPrimitives	rayPolygons	rayK	shadowRays

Θα περιγράψουμε τα βήματα της διαδικασίας για ένα στρώμα καθότι είναι η ίδια και για τα υπόλοιπα.

1. Διάλεξε τυχαία το x% ποσοστό αρχείων από το στρώμα. Τα αρχεία αυτά τα ονομάζουμε προ-δείγματα.
2. Άδειασε την βάση δεδομένων. Αυτό γίνεται για λόγους ασφαλείας σε περίπτωση που από κάποιο λάθος υπάρχουν ξεχασμένα δεδομένα στον πίνακα stats.

3. Εκτέλεσε τα προ-δείγματα σε ανάλυση 1x1 και εισήγαγε τα στατιστικά τους στη βάση. Το ρεύμα εξόδου του Air διαβάζεται ως μεμονωμένοι συνεχόμενοι χαρακτήρες. Κάνοντας χρήση της κλάσης StreamTokenizer μπορούμε εύκολα να διαχωρίσουμε το ρεύμα σε λέξεις και αριθμούς και στη συνέχεια να εισαχθούν στην βάση.
4. Βρες τον μέγιστο αριθμό εγγραφών N και τις εγγραφές, που τα στατιστικά τους δεν αποκλίνουν παραπάνω από d%. Από τις εγγραφές διάλεξε τυχαία μία και αποθήκευσέ την ως δείγμα με βάρος N. Διέγραψε αυτές τις εγγραφές.
5. Επανέλαβε το βήμα 4 μέχρι να αδειάσει η βάση.

Αφού ολοκληρωθούν τα βήματα κάθε δείγμα αντιπροσωπεύει ένα υπόστρωμα με το αντίστοιχο βάρος του.

7.2 Πλατφόρμες και προγραμματιστικά εργαλεία

7.2.1 Πλατφόρμα ανάπτυξης

Όλη η ανάπτυξη της μεθοδολογίας και η εκτέλεση των μετρήσεων έγιναν σε PC με επεξεργαστή Intel Core to Duo E6400 στα 2.1Ghz, με 1.5Gb μνήμη και σκληρό δίσκο western digital Sata 250 MB. Το λειτουργικό σύστημα ήταν Microsoft Windows XP Pro + Service Pack 2. Η έκδοση του air ήταν η ver 2.3.4 για windows.

7.2.2 Γλώσσες προγραμματισμού

7.2.2.1 JAVA

Η προτιμότερη γλώσσα για την δημιουργία μιας εφαρμογής που θα κάνει εκτίμηση του φόρτου εργασίας σε περιβάλλον πλέγματος και που χρησιμοποιήσαμε είναι η γλώσσα προγραμματισμού Java. Η επιλογή αυτή γίνεται λόγω των πλεονεκτημάτων που έχει η συγκεκριμένη γλώσσα και των τεράστιων δυνατοτήτων που προσφέρει. Σύμφωνα με την κατασκευάστριά της εταιρία την Sun, «Η Java είναι μια απλή, αντικειμενοστραφής, κατανεμημένη, ερμηνευόμενη, εύρωστη, αρχιτεκτονικά ουδέτερη, μεταφερόμενη, υψηλής απόδοσης, πολυνηματική και ασφαλής γλώσσα».

Κατασκευαστικά η Java είναι προσανατολισμένη ως προς την ασφάλεια των εφαρμογών της. Έχει δυνατότητες ασφάλειας και προστασίας του απορρήτου που χρησιμοποιούνται πλήρως από τα πλέγματα. Γνωρίζουμε επίσης ότι από την αρχή η Java δημιουργήθηκε για εφαρμογές που θα λειτουργούσαν πάνω από το Internet. Έχει λοιπόν σαφές προβάδισμα σε σχέση με οποιαδήποτε άλλη γλώσσα σε χρήσεις που απαιτούν εύκολη, γρήγορη και ασφαλή

διαδικτυακή επικοινωνία κάτι αναγκαίο σε κάθε επαγγελματικό Grid. Ακόμη είναι πολύ σημαντικό ότι κάθε εφαρμογή γραμμένη σε Java κώδικα μετασχηματίζεται πολύ εύκολα σε υπηρεσίας ιστού κάτι που απαιτείται ιδιαίτερα στην εποχή μας.

Το κυριότερο ίσως χαρακτηριστικό της Java είναι ότι είναι ανεξάρτητη πλατφόρμας (Platform Independent). Κάθε εφαρμογή δηλαδή μπορεί να λειτουργήσει σε κάθε υπολογιστικό σύστημα είτε αυτό είναι προσωπικός υπολογιστής με Linux ή Windows είτε ένα οποιοδήποτε ενσωματωμένο σύστημα (embedded system) το οποίο ενσωματώνει τεχνολογία Java. Με αυτόν τον τρόπο τόσο ο πελάτης όσο και ο πάροχος μπορούν να συνεχίσουν να χρησιμοποιήσουν το πλέγμα χωρίς να πρέπει να αλλάξουν τα λειτουργικά τους συστήματα ή τον εξοπλισμό του.

7.2.2.2 *SQL (Structured Query Language)*

Η SQL είναι μια αναγνωρισμένη τυποποιημένη γλώσσα με σκοπό να οργανώσει, να διαχειριστεί και να ανακτήσει τα στοιχεία από μια βάση δεδομένων. Είναι ουσιαστικά μια γλώσσα προγραμματισμού για τις σχεσιακές βάσεις δεδομένων πολύ φιλική προς τον χρήστη και λειτουργεί ως διαδραστικό μέσο για την επικοινωνία του χρήστη με την βάση. Ο πυρήνας του SQL αποτελείται από μια γλώσσα που να επιτρέπει την ανάκτηση, την εισαγωγή, την ενημέρωση και την διαγραφή των δεδομένων, καθώς και τον έλεγχο της διαχείρισης της βάσης [24]. Επίσης η SQL είναι συμβατή με τους περισσότερους εμπορικούς database server, ένα γεγονός που ανεξαρτητοποιεί την υλοποίηση ενός προγράμματος που κάνει χρήση της SQL από την πλατφόρμα που θα εκτελεστεί. Τέλος τα εμπορικά πακέτα παρέχουν επίπεδα ασφαλείας των δεδομένων καθώς και βέλτιστους αλγόριθμους αναζήτησης κάτι που αυξάνει την ταχύτητα των προγραμμάτων μας από το να υλοποιούσαμε μία δικιά μας εσωτερική βάση μέσα στο πρόγραμμα.

7.2.3 *Προγραμματιστικά εργαλεία*

Για την ανάπτυξη των εφαρμογών σε αυτή την διπλωματική χρησιμοποιήσαμε δύο προγραμματιστικά εργαλεία. Το Eclipse IDE και τον MySQL server. Το Eclipse IDE είναι ένα πολύ εύχρηστο ολοκληρωμένο περιβάλλον για την ανάπτυξη εφαρμογών σε Java. Περιέχει ενσωματωμένο editor, debugger και compiler που κάνουν πιο εύκολη την διαδικασία του προγραμματισμού. Επίσης το eclipse είναι μια κοινότητα ανοιχτού κώδικα όπου τα έργα της εστιάζουν στην δημιουργία ανοιχτών πλατφόρμων ανάπτυξης που περιέχουν εργαλεία για την ανάπτυξη και την διαχείριση λογισμικού [21]. Ο MySQL server είναι από τις πιο δημοφιλείς ανοιχτού κώδικα βάση δεδομένων στον κόσμο για τον λόγο ότι είναι συνεπής, αποδοτικός, αξιόπιστος και εύκολος στην χρήση. Είναι ιδανικός για εφαρμογές δικτύου καθώς τρέχει σε περισσότερες από 20 πλατφόρμες συμπεριλαμβανομένων

και των Linux, Windows, OS/X, HP-UX, AIX, Netware δίνοντας μία αίσθηση ευελιξίας [22]. Επίσης το εργαλείο Gui Administrator που περιέχει προσφέρει διαχείριση των βάσεων μέσω γραφικής διασύνδεσης, ένα γεγονός που κάνει πιο ξεκούραστη τη χρήση του αφού αποφεύγονται από τον προγραμματιστή οι command line.

8

Επίλογος

8.1 Σύνοψη και συμπεράσματα

Στην παρούσα μελέτη αναπτύξαμε μία μεθοδολογία για την εκτίμηση του φόρτου εργασίας σε περιβάλλον υπολογιστικού πλέγματος με έμφαση στην περίπτωση της εφαρμογής του 3D Rendering. Είδαμε πόσο σημαντική είναι η εκ των προτέρων γνώση του φόρτου εργασίας αφού συμβάλλει στην καλύτερη αξιοποίηση των πόρων του πλέγματος και την χρονοδρομολόγηση των εργασιών καθώς και στην αύξηση της ποιότητας της υπηρεσίας που προσφέρεται από τον πάροχο στον χρήστη-πελάτη τηρώντας τις συμφωνίες SLAs.

Ο τρόπος με τον οποίο προσεγγίσαμε το πρόβλημα ήταν τελείως διαφορετικός από οποιεσδήποτε άλλες προσεγγίσεις και πρωτότυπος. Θα μπορούσαμε να πούμε ότι ήταν καθαρά πρακτικός και πειραματικός. Ο συνδυασμός της εκτέλεσης δείγματος της εργασίας για την άντληση πληροφοριών του φόρτου και της αξιοποίηση της εμπειρίας του παρελθόντος, καθιστούν την μεθοδολογία μας ένα ισχυρό εργαλείο με προοπτικές εξέλιξης.

Μέσα από τις μετρήσεις που πραγματοποιήσαμε η μεθοδολογία μας αποδείχτηκε εφικτή και αξιόπιστη αφού μπόρεσε να φέρει εις πέρας το δύσκολο έργο της εκτίμησης του φόρτου για την ιδιαίτερη εφαρμογή του rendering. Καταλήξαμε σε έναν ολοκληρωμένο μηχανισμό χρησιμοποιώντας μεθόδους δειγματοληψίας για την διεκπεραίωση δείγματος της εργασίας με σκοπό την άντληση πληροφοριών φόρτου για ολόκληρη την εργασία και ενσωματώσαμε ένα Case-Based Reasoning έμπειρο σύστημα για την αξιοποίηση της εμπειρίας του παρελθόντος

και την βελτιστοποίηση του μηχανισμού μας. Έτσι καταφέραμε να μετρήσουμε μερικές από τις σημαντικότερες παραμέτρους που αντιπροσωπεύουν τον φόρτο εργασίας όπως ο χρόνος διεκπεραίωσης της εργασίας, η μέγιστη χρησιμοποίηση μνήμης και ο απαιτούμενος αποθηκευτικός χώρος. Ειδικότερα ο χρόνος διεκπεραίωσης είναι η πιο βασική πληροφορία για την αποδοτικότερη λειτουργία του πλέγματος και την τήρηση των συμφωνιών SLAs.

Το πιο μεγάλο πλεονέκτημα της μεθοδολογίας αυτής είναι η γενικότητά της και η προσαρμοστικότητά της σε άλλες εφαρμογές πλέγματος. Δηλαδή ο σχεδιαστής, ενός ΕΦΕ μιας εφαρμογής μελετώντας τα ιδιαίτερα χαρακτηριστικά της εφαρμογής και κάνοντας τις κατάλληλες τροποποιήσεις μπορεί να προσαρμόσει την μεθοδολογία στην εκάστοτε περίπτωση. Βέβαια επειδή η βασική ιδέα βασίζεται στην δειγματοληψία η μέθοδος αδυνατεί να αντεπεξέλθει σε συνθήκες όπου μέσα σε μια εργασία δεν υπάρχει αντιπροσωπευτικό δείγμα του πληθυσμού.

8.2 Μελλοντικές επεκτάσεις

Η μεθοδολογία της εκτίμησης του φόρτου εργασίας για την περίπτωση του 3D Rendering έχει πολλά περιθώρια επέκτασης και βελτίωσης. Όσο μεγαλύτερη γνώση μπορούμε να συλλέγουμε για μια εργασία τόσο μεγαλύτερη αξιοποίηση της εμπειρίας του παρελθόντος μπορούμε να πετύχουμε. Η ιδανική περίπτωση θα ήταν να έχουμε μία πολλή μεγάλη βάση γνώσης έτσι ώστε μόνο από τα αποτελέσματα των στατιστικών της εκτέλεσης ενός αρχείου σε ανάλυση 1x1 και την σύγκρισή τους με προηγούμενες περιπτώσεις να μπορούμε να εξάγουμε τις πληροφορίες του φόρτου εργασίας. Αυτή η περίπτωση θα δημιουργούσε έναν γρήγορο και αξιόπιστο εκτιμητή αφού θα είχε την δυνατότητα να εκτελεί περισσότερα δείγματα σε πολύ μικρό χρόνο. Μια άλλη ιδέα είναι να εφαρμοστούν οι μέθοδοι δειγματοληψίας με τελείως διαφορετικό τρόπο και η εύρεση των δειγμάτων-αρχείων που θα εκτελεστούν να γίνεται με λεκτική ανάλυση (parsing) στα rib αρχεία. Δηλαδή η ομοιότητα των αρχείων θα βρίσκεται από την λεκτική τους ανάλυση και την καταγραφή των ιδιοτήτων τους. Τέλος σε πολλά σημεία της μεθοδολογίας μας χρησιμοποιούμε αλγόριθμους ταιριάσματος. Η εισαγωγή βελτιωμένων αλγόριθμων ταιριάσματος στην σχεδίαση ενός ΕΦΕ θα βελτιώνε ολόκληρη την συμπεριφορά του ΕΦΕ.

9

Βιβλιογραφία και αναφορές

- [1] [ASI05] A. Asiki, Introduction to Grid Technologies JANUARY 2005 available at <http://www.csl.ee.upatras.gr/egee/files/Eisagogi.pdf>
- [2] [BON07] J. Bonebakker Finding Representative Workloads for Computer System Design DECEMBER 2007 available at http://research.sun.com/techrep/2007/sml_i_tr-2007-174.pdf
- [3] [DDP+04] N. Doulamis, A. Doulamis, A. Panagakis, Konstantinos Dolkas, T. Varvarigou, E. Varvarigos A Combined Fuzzy-Neural Network Model for Non-Linear Prediction of 3-D Rendering Workload in Grid Computing vol34 2 APRIL 2004 available at <http://ieeexplore.ieee.org/iel5/3477/28545/01275553.pdf>
- [4] [DOU04] A. Doulamis Fair QoS resource management and non-linear prediction of 3D Rendering applications vol 3 MAY 2004 available at http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1328890
- [5] [JMS] L. Jin, V. Machiraju, A. Sahai Analysis on Service Level Agreement of Web Services available at <http://www.hpl.hp.co.uk/techreports/2002/HPL-2002-180.pdf>
- [6] [KS91] G. Kokolakis, I. Spiliotis Εισαγωγή στη θεωρία πιθανοτήτων και στατιστική(2), SYMEON PUBLISHING Co MAY 1991
- [7] [LTV05] A. Litke, K. Tserpes, T. Varvarigou Computational workload prediction for Grid oriented industrial applications: the case of 3D-Image Rendering 2005 available at http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1558665

- [8] [MEN02] D. A. Menascé, QoS Issues in web services DECEMBER 2002 available at <http://ieeexplore.ieee.org/search/wrapper.jsp?arnumber=1067740>
- [9] [MEN04] A. Menychtas Προγραμματισμός σε περιβάλλον πολυπλέγματος OCTOBER 2004
- [10] [MSK+] J. MacLaren, R. Sakellariou, K. T. Krishnakumar, J. Garibaldi, D. Ouelhadj Towards Service Level Agreement Based Scheduling on the Grid available at <http://www.cs.man.ac.uk/~rizos/papers/icaps04.pdf>
- [11] [PS] S. K. Pal SIMON C. K. Shiu Foundations of soft case-based reasoning John Wiley & Sons, INC., Publication
- [12] [SCH02] J. M. Schopf A General Architecture for Scheduling on the Grid 2002 available at <http://www-unix.mcs.anl.gov/~schopf/Pubs/sched.arch.2002.pdf>
- [13] [ST05] L. Srinivasan, J. Treadwell, An Overview of Service-oriented Architecture, Web Services and Grid Computing. NOVEMBER 2005 available at <http://h71028.www7.hp.com/ERC/downloads/SOA-Grid-HP-WhitePaper.pdf>
- [14] [SYC+07] Wen-Chung Shih, Chao-Tung Yang, Chun-Jen Chen, Shian-Shyong Tsen Parallel Loop Scheduling Using Knowledge-Based Workload Estimation on Grid Environments JANUARY 2007 available at <http://ieeexplore.ieee.org/search/wrapper.jsp?arnumber=4090042>
- [15] [TZA96] S. Tzafestas, Εισαγωγή στην τεχνητή νοημοσύνη και τα έμπειρα συστήματα (2), FEBRUARY 1996
- [16] http://en.wikipedia.org/wiki/Grid_computing 2008
- [17] <http://en.wikipedia.org/wiki/Workload> 2008
- [18] [http://en.wikipedia.org/wiki/Sampling_\(statistics\)](http://en.wikipedia.org/wiki/Sampling_(statistics)) 2008
- [19] <http://www.statpac.com/surveys/sampling.htm> 2008
- [20] <http://www.astrosurf.com/luxorion/terrigen-rendering-time.htm> 2008
- [21] <http://www.eclipse.org/> 2008
- [22] <http://www.mysql.com/why-mysql/> 2008
- [23] [http://en.wikipedia.org/wiki/Rendering_\(computer_graphics\)](http://en.wikipedia.org/wiki/Rendering_(computer_graphics)) 2008
- [24] <http://en.wikipedia.org/wiki/SQL> 2008
- [25] http://grid.ece.ntua.gr/NextGRIDWiki/index.php/Integration_Diagram 2008
- [26] http://grid.ece.ntua.gr/NextGRIDWiki/index.php/NextGRID_Architecture 2008