

Ατομική Διπλωματική Εργασία

CHARACTERIZATION OF A WEB SERVING WORKLOAD

Βάκης Χριστοφόρου

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ



ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Μάιος 2014

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

CHARACTERIZATION OF A WEB SERVING WORKLOAD

Βάκης Χριστοφόρου

Επιβλέπων Καθηγητής
Γιάννος Σαζεΐδης

Η Ατομική Διπλωματική Εργασία υποβλήθηκε προς μερική εκπλήρωση των
απαιτήσεων απόκτησης του πτυχίου Πληροφορικής του Τμήματος Πληροφορικής του
Πανεπιστημίου Κύπρου

Μάιος 2014

Ευχαριστίες

Θα ήθελα να εκφράσω τις θερμές μου ευχαριστίες στον επιβλέποντα καθηγητή μου, κ. Γιάννο Σαζεΐδη, ο οποίος υπήρξε εξάίρετος καθοδηγητής σε όλη την πορεία εκπλήρωσης της διπλωματικής μου εργασίας. Υπήρξε βοηθητικός και πάντοτε πρόθυμος να συζητήσει την πρόοδο που επιτεύχθηκε. Επίσης θα ήθελα να ευχαριστήσω την οικογένεια μου για την στήριξη που μου παρείχαν καθ' όλη την διάρκεια των σπουδών μου.

Περίληψη

Η ραγδαία εξάπλωση των κοινωνικών εφαρμογών στο διαδίκτυο, οδηγεί την παρούσα υποδομή των data centers να μην μπορεί να ανταποκριθεί αποδοτικά. Αυτό οφείλεται στο γεγονός ότι οι ανάγκες αυτών των εφαρμογών είναι διαφορετικές από τις παραδοσιακές υπηρεσίες που προσφέρονταν μέχρι στιγμής στο διαδίκτυο. Η παρούσα διπλωματική εργασία στοχεύει στην αξιολόγηση του web serving benchmark του Cloudsuite το οποίο αντιπροσωπεύει μία web 2.0 social εφαρμογή. Έχουν διενεργηθεί πειράματα και εξαχθεί αποτελέσματα και συμπεράσματα που οδηγούν σε κατανόηση των αναγκών μιας τέτοιας εφαρμογής όσον αφορά το υλικό. Αυτό θα βοηθήσει στην περαιτέρω βελτίωση της υποδομής των data centers με σκοπό την μείωση της καταναλωμένης ενέργειας, αλλά και γενικότερα του κόστους διαχείρισης, χωρίς να περιορίζεται η ποιότητα των παρεχόμενων υπηρεσιών.

Περιεχόμενα

Κεφάλαιο 1	Εισαγωγή.....	1
	1.1 Το web serving	1
	1.2 Σκοπός Εργασίας	2
	1.3 Συνεισφορά Εργασίας	2
	1.4 Οργάνωση Κεφαλαίων	3
Κεφάλαιο 2	Σχετική Εργασία.....	4
	2.1 Τα data centers και η σημασία ορθής διαχείρισης	4
	2.2 Η έννοια του Total Cost of Ownership για ένα data center	5
	2.3 Σημασία της ενέργειας	6
	2.4 Χρήση λιγότερο αποδοτικών επεξεργαστών και ετερογένειας	7
Κεφάλαιο 3	To Web Serving Benchmark.....	8
	3.1 Περιγραφή	8
	3.2 Αρχιτεκτονική Συστήματος	9
	3.3 Ο Εξυπηρετητής	10
	3.4 Βάση Δεδομένων	12
	3.5 Προσομοίωση Χρηστών	14
	3.6 Εγκατάσταση των επιμέρους συστημάτων	14
	3.7 Πρόσβαση και αλληλεπίδραση των επιμέρους συστημάτων	14
	3.8 Περιγραφή της ροής των αιτημάτων	16
Κεφάλαιο 4	Πειράματα και Αποτελέσματα.....	19
	4.1 Μεθοδολογία πραγματοποιηθέντων πειραμάτων	19
	4.2 Ανάλυση του χρόνου ανταπόκρισης αιτημάτων	22
	4.3 Ανάλυση του throughput από τον εξυπηρετητή	27
	4.4 Ανάλυση των δυνατοτήτων του εξυπηρετητή	28
	4.5 Ανάλυση της χρήσης του δίσκου	30
	4.6 Ανάλυση του συστήματος με χαμηλότερη συχνότητα	31

Κεφάλαιο 5	Συμπεράσματα.....	37
	5.1 Συγκεντρωτικά συμπεράσματα	37
	5.2 Μελλοντική εργασία	38
Βιβλιογραφία		39

Κεφάλαιο 1

Εισαγωγή

1.1 Το Web Serving	1
1.2 Σκοπός Εργασίας	2
1.3 Συνεισφορά Εργασίας	2
1.4 Οργάνωση Κεφαλαίων	3

1.1 Το Web Serving

Το web serving αποτελεί μία από τις κυριότερες υπηρεσίες του διαδικτύου. Ο τρόπος όμως, που προσφέρεται το περιεχόμενο από τους εξυπηρετητές έχει ξεκινήσει να αλλάζει. Αυτό οφείλεται κυρίως στην ευρεία εξάπλωση των κοινωνικών και άλλων εφαρμογών, των οποίων οι ανάγκες διαφοροποιούνται σε σχέση με τις μέχρι τώρα παραδοσιακές εφαρμογές που συναντούσε κανείς στο διαδίκτυο. Αυτές οι νέες εφαρμογές, κάνουν χρήση δυναμικού και στατικού περιεχομένου, όπου μεγάλος αριθμός χρηστών έχουν πρόσβαση ταυτόχρονα και μεταβάλλουν συνεχώς το περιεχόμενο. Με βάση αυτό, οι εφαρμογές πρέπει να είναι σε θέση να προσφέρουν διαχείριση σφαλμάτων και συνάμα τη μέγιστη δυνατή ικανοποίηση των χρηστών τους.

Η ιδέα της υλοποίησης και προσφοράς περιεχομένου από αυτές τις εφαρμογές βασίζεται στα εξής χαρακτηριστικά: Τα αιτήματα των χρηστών λαμβάνονται από έναν διαχειριστή φόρτου (load balancer) για τους διαθέσιμους εξυπηρετητές. Ακολούθως το αίτημα λαμβάνεται από έναν διαθέσιμο εξυπηρετητή, ο οποίος είτε απαντά άμεσα με ένα αρχείο το οποίο είναι αποθηκευμένο στατικά, είτε προωθεί το αίτημα σε ένα script, το οποίο είναι υπεύθυνο για τη δημιουργία δυναμικού περιεχομένου. Όλες οι απαιτούμενες πληροφορίες κατάστασης για την διεκπεραίωση των αιτημάτων είναι αποθηκευμένες σε βάση δεδομένων.

1.2 Σκοπός εργασίας

Η παρούσα εργασία στοχεύει στην αξιολόγηση του Web Serving Benchmark του Cloudsuite ως προς την χρήση των αλληλεπιδρώντων συστημάτων. Γίνεται ένας χαρακτηρισμός της απόδοσης του συστήματος και της χρήσης του υλικού κάτω από διαφορετικές ρυθμίσεις.

1.3 Συνεισφορά εργασίας

1.3.1 Εγκατάσταση

Το Benchmark αποτελείται από 3 μέρη· τον εξυπηρετητή, τη βάση δεδομένων και τον Faban Driver, ο οποίος υλοποιεί τους χρήστες που διενεργούν αιτήματα στον εξυπηρετητή. Το κάθε ένα από τα 3 μέρη εγκαταστάθηκε σε διαφορετική μηχανή, όπως περιγράφεται από το Cloudsuite. Ολόκληρο το benchmark υλοποιεί μία web 2.0 εφαρμογή, η οποία χρησιμοποιείται σε αυτή την εργασία με σκοπό τον χαρακτηρισμό της απόδοσης.

1.3.2 Παράμετροι που εξετάστηκαν

Σε αυτή την εργασία γίνεται μια προσπάθεια να γίνουν αντιληπτά τα διάφορα σημεία που επηρεάζουν την απόδοση μιας web 2.0 εφαρμογής. Έχουν διενεργηθεί πειράματα με διαφορετικές ρυθμίσεις στον εξυπηρετητή, ως προς την συχνότητα του CPU και τις ταυτόχρονες διαθέσιμες συνδέσεις. Επίσης, έχουν διενεργηθεί πειράματα με διαφορετικό αριθμό ταυτόχρονων χρηστών. Η εργασία αυτή διαφοροποιείται από τις προηγούμενες ως προς το γεγονός ότι εξετάζονται πολλοί παράμετροι της λειτουργίας της εφαρμογής, και δίδεται μία σφαιρική εικόνα σχετικά με τους συντελεστές που επηρεάζουν την απόδοση της συγκεκριμένης εφαρμογής.

1.3.3 Αποκόμιση γνώσης

Η παρούσα εργασία, συνετέλεσε στο να γίνουν αντιληπτά σε ατομικό επίπεδο όλα τα στάδια της διενέργειας και ανταπόκρισης αιτημάτων σε μία Web 2.0 εφαρμογή, καθώς επίσης και τους σημαντικούς συντελεστές που επηρεάζουν την προσφορά του περιεχομένου από αυτές τις εφαρμογές.

1.4 Οργάνωση κεφαλαίων

Η οργάνωση της εργασίας έχει γίνει ως ακολούθως: Στο κεφάλαιο 2 γίνεται μία αναφορά στο θεωρητικό υπόβαθρο που διέπει τα data centers καθώς και στα κόστη και στην λειτουργία αυτών. Στο 3^ο κεφάλαιο γίνεται μία εκτενής περιγραφή του υπό ανάλυση benchmark. Ακολουθεί η ανάλυση και τα αποτελέσματα των διενεργηθέντων πειραμάτων στο 4^ο κεφάλαιο, και εν συνεχεία στο 5^ο κεφάλαιο καταγράφονται τα συμπεράσματα.

Κεφάλαιο 2

Σχετική Εργασία

2.1 Τα data centers και η σημασία ορθής διαχείρισης	4
2.2 Η έννοια του Total Cost of Ownership (TCO) για ένα data center	5
2.3 Σημασία της ενέργειας	6
2.4 Χρήση λιγότερο αποδοτικών επεξεργαστών και ετερογένειας	7

2.1 Τα data centers και η σημασία ορθής διαχείρισης

Το cloud computing αναδεικνύεται ως κυρίαρχη πλατφόρμα υπολογισμού για την παροχή online υπηρεσιών σε μια παγκόσμια βάση χρηστών. Δημοφιλής online υπηρεσίες, όπως η αναζήτηση στο διαδίκτυο, τα κοινωνικά δίκτυα και τα βίντεο, φιλοξενούνται σε μεγάλα data centers. Λόγω της ραγδαίας ανάπτυξης αυτής της βιομηχανίας, οι φορείς παροχής αυτών των υπηρεσιών αναπτύσσουν νέα κέντρα, έτσι ώστε να αυξήσουν την υπάρχουσα υποδομή για την κάλυψη της αυξανόμενης ζήτησης.

Έχει διαπιστωθεί, ότι τα scale-out workloads έχουν παρόμοια χαρακτηριστικά. Συγκεκριμένα, αλληλεπιδρούν σε μεγάλα σύνολα δεδομένων που είναι σκορπισμένα σε έναν μεγάλο αριθμό μηχανών, εξυπηρετούν μεγάλο αριθμό εντελώς ανεξάρτητων αιτημάτων που δεν διαμοιράζουν οποιοδήποτε state, και η επικοινωνία υπάρχει μεταξύ των servers για εργασίας διαχείρισης και συντονισμού.

Αυτού του είδους οι εφαρμογές απαιτούν τεράστιες ποσότητες υπολογιστικών πόρων. Ωστόσο τα σημερινά data centers αντιμετωπίζουν φυσικούς περιορισμούς σε σχέση με τον χώρο και την δύναμη, περιορίζοντας έτσι την περαιτέρω επέκταση, καλώντας έτσι για βελτιώσεις όσον αφορά την υπολογιστική ικανότητα ανά server και στην ενέργεια που αναλώνεται ανά λειτουργία. Οι επεξεργαστές που κυριαρχούν σήμερα, δεν είναι αποδοτικοί στο να τρέχουν αυτού του είδους τις εφαρμογές [5].

Η αναποτελεσματικότητα προέρχεται από την μεγάλη διαφορά που υπάρχει μεταξύ των αναγκών των εργασιών και των επεξεργαστών. Συνεχίζοντας τις σημερινές τάσεις αύξησης των επεξεργαστών θα επιδεινώσουν περαιτέρω την αναποτελεσματικότητα στο μέλλον, και επομένως υφίσταται άμεση ανάγκη για αλλαγή στους επεξεργαστές που θα οδηγήσουν σε μία βελτιωμένη υπολογιστική πυκνότητα, και στην εξοικονόμηση ενέργειας στα data centers.

Τα διάφορα workloads έχουν διαφορετικές απαιτήσεις. Κάποια είναι διαδραστικά, άλλα κάνουν ευρεία χρήση της μνήμης, άλλα έχουν μεγάλες υπολογιστικές απαιτήσεις, κάποια έχουν όρια καθυστέρησης και κάποια όχι. Επομένως, μία μέση λύση όσον αφορά την αρχιτεκτονική στα data centers θα μπορούσε να ικανοποιήσει αυτές τις αντικρουόμενες ανάγκες, ή εναλλακτικά θα μπορούσε να αναπτυχθεί ένα ετερογενές σύστημα, όπου διαφορετικά workloads θα τρέχουν σε διαφορετικό hardware.

Σημαντικό στοιχείο είναι να μπορούμε να προβλέψουμε τι είδους εφαρμογές θα αναπτυχθούν στο μέλλον και με ποιες απαιτήσεις, καθώς και πως θα εξελιχθούν οι σημερινές. Όμως το σημαντικό κόστος ανάπτυξης ενός data center, απαιτεί τις σωστές επιλογές σχετικά με το hardware που θα αγοραστεί. Τα high-end hardware είναι ακριβά και καταναλώνουν σημαντικές ποσότητες power, που οδηγεί σε αυξημένο TCO, όμως θα αποτελεί την σωστή επιλογή αν αυτό που χρειαζόμαστε είναι υψηλή απόδοση. Διαφορετικά, πιο φτηνό και οικονομικότερο ενεργειακά hardware πιθανόν να αποτελεί καλύτερη επιλογή. Η απάντηση στην ερώτηση του τι hardware να αγοραστεί θα ήταν εύκολη, αν είχαμε μόνο ένα είδος workload. Λόγω του ότι σε ένα data center που τρέχει μία Web 2.0 εφαρμογή (π.χ. social networking) και τρέχουν πέραν της μίας υπηρεσίας, αυτό το ερώτημα δεν είναι εύκολο να απαντηθεί.

2.2 Η έννοια του Total Cost of Ownership (TCO) για ένα data center

Η δημιουργία και ενός data center στοιχίζει πολλά εκατομμύρια δολάρια και αποτελεί μία σημαντική επένδυση για έναν οργανισμό ή εταιρεία. Επιπροσθέτως της δημιουργίας, ένα αξιοσημείωτο ποσό αναλώνεται στην λειτουργία και συντήρηση ενός κέντρου δεδομένων. Επομένως είναι σημαντικό το συνολικό αυτό κόστος να διατηρείται όσο πιο χαμηλό είναι δυνατόν. [9]

Το TCO αναλύεται στο κόστος ανέγερσης του κτηρίου, στην ενέργεια που θα αναλίσκεται, στην υποδομή παροχής της ενέργειας, στο κόστος του μηχανισμού cooling, στο κόστος αγοράς του εξοπλισμού και τέλος στα κόστη λειτουργίας και συντήρησης [1]. Το κόστος των εξυπηρετητών και η ενέργεια κατέχουν τα μεγαλύτερα ποσοστά του συνολικού κόστους και επομένως η μείωση του συνολικού TCO εστιάζεται σε αυτές τις δύο παραμέτρους, χωρίς αυτό να σημαίνει ότι η βελτίωση της αποδοτικότητας των υπόλοιπων μετρικών δεν είναι σημαντική.

2.3 Σημασία της ενέργειας

2.3.1 Το κόστος της ενέργειας

Ένα πολύ σημαντικό κόστος για την λειτουργία ενός data center είναι η υποδομή διανομής της ενέργειας αλλά και η ίδια η ενέργεια που καταναλώνεται. Αυτό παίζει σημαντικό ρόλο στη διαμόρφωση του TCO, και επομένως πρέπει να μπορέσουμε να μειώσουμε αυτό το κόστος. Επιπλέον η μεγάλη κατανάλωση ενέργειας έχει σημαντικές περιβαλλοντικές επιπτώσεις, που λόγω και της αυξανόμενης μόλυνσης του πλανήτη τα τελευταία χρόνια, απασχολεί σε μεγάλο βαθμό. Ο σημαντικότερος παράγοντας που συντελεί στον μετριασμό της απαιτούμενης ενέργειας, είναι το hardware που έχει αγοραστεί καθώς και ο τρόπος που αυτό χρησιμοποιείται και διαχειρίζεται.

2.3.2 Κατανάλωση Ενέργειας ανάλογη με τον φόρτο εργασίας

Ένα σημαντικό πρόβλημα που αντιμετωπίζουν τα data-centers είναι το γεγονός ότι η κατανάλωση ενέργειας δεν είναι ανάλογη του φόρτου εργασίας. Αυτό παρατηρείται κυρίως όταν η εργασία που πρέπει να διεκπεραιωθεί είναι σχετικά περιορισμένη σε σχέση με τις δυνατότητες του data-center. Σε αυτές τις περιπτώσεις παρατηρείται το φαινόμενο υψηλής κατανάλωσης ενέργειας, σε αναλογική σχέση με την κατανάλωση που υπάρχει όταν ο φόρτος εργασίας είναι μεγάλος. Έχουν προταθεί διάφορες τεχνικές για τον μετριασμό αυτού του προβλήματος όσον αφορά την μνήμη [7,10], και τον διαμοιρασμό της εργασίας στους διαθέσιμους εξυπηρετητές [8].

2.4 Χρήση λιγότερο αποδοτικών επεξεργαστών και ετερογένειας

Μέχρι και το πρόσφατο παρελθόν, στα διάφορα data centers χρησιμοποιούνταν συμβατικοί επεξεργαστές οι οποίοι καταναλώνουν σημαντικά ποσοστά ενέργειας ακόμη και όταν αυτοί βρίσκονται σε αδράνεια. Έχουν γίνει διάφορες έρευνες οι οποίες καταδεικνύουν ότι η αλλαγή της μέχρι τώρα χρήσης συμβατικών επεξεργαστών και η εφαρμογή διαφόρων τεχνικών, αποφέρει σημαντικές μειώσεις στην απαιτούμενη ενέργεια, και ως εκ τούτου μειώνεται και το TCO. Αυτές οι τεχνικές περιλαμβάνουν νέους επεξεργαστές που είναι αποδοτικότεροι για αυτές τις εφαρμογές [1,2,4], καθώς και η χρήση ετερογένειας ούτως ώστε να επιτυγχάνεται κατανάλωση ενέργειας ανάλογη με τον φόρτο εργασίας [3,6].

Το σημαντικότερο στοιχείο, είναι να μειώσουμε όσο το δυνατόν την απόδοση των χρησιμοποιούμενων υλικών, ούτως ώστε να ανταποκρίνονται εντός των ορίων στα αιτήματα που καλούνται να διεκπεραιώσουν. Με βάση αυτό, επιδιώκεται η μέγιστη δυνατή απόδοση του data center καθώς και η ενέργεια που καταναλώνεται να μεταβάλλεται ανάλογα με τον φόρτο εργασίας του data center.

Κεφάλαιο 3

To Web Serving Benchmark

3.1 Περιγραφή	8
3.2 Αρχιτεκτονική Συστήματος	9
3.3 Ο Εξυπηρετητής	10
3.4 Βάση Δεδομένων	12
3.5 Προσομοίωση Χρηστών	14
3.6 Εγκατάσταση των επιμέρους συστημάτων	14
3.7 Πρόσβαση και αλληλεπίδραση των επιμέρους συστημάτων	14
3.8 Περιγραφή της ροής των αιτημάτων	16

3.1 Περιγραφή

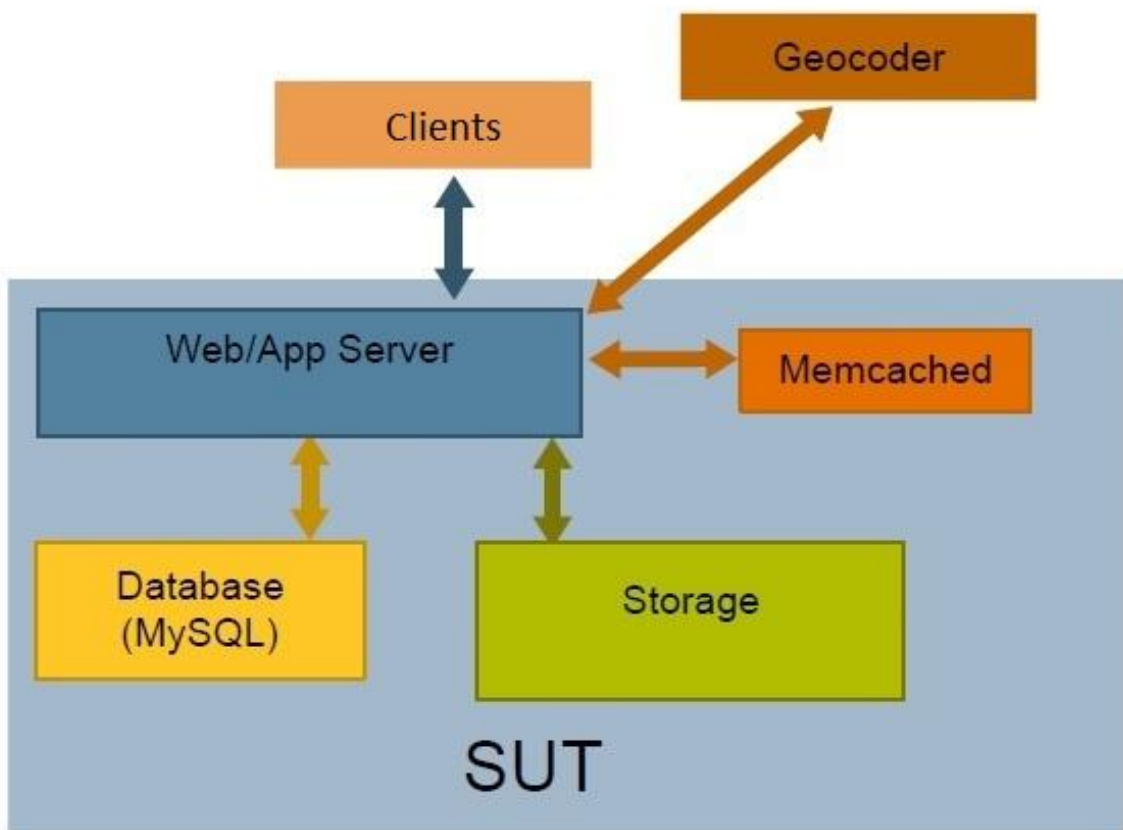
Το benchmark που χρησιμοποιείται για αξιολόγηση είναι το Cloudstone, το οποίο είναι ένα Web 2.0 benchmark. Περιλαμβάνει την PHP έκδοση του Olio, το οποίο είναι μία Web 2.0 κοινωνική εφαρμογή, και τους χρήστες / πελάτες οι οποίοι υλοποιούνται με την χρήση του Faban workload generator. Η Olio εφαρμογή, επιτρέπει την εισαγωγή καινούργιων χρηστών και τη σύναψη φιλίας μεταξύ αυτών. Επιτρέπει επίσης την εισαγωγή διάφορων γεγονότων (events) στο σύστημα από εγγεγραμμένους χρήστες, όπου για αυτά μπορούν να ενημερωθούν οι υπόλοιποι χρήστες και να δηλώσουν συμμετοχή, ακόμη και να τα σχολιάσουν. Τα άτομα και τα γεγονότα έχουν αποθηκευμένη στο σύστημα μία φωτογραφία (image) και ένα thumbnail. Επιπρόσθετα, τα γεγονότα έχουν αποθηκευμένη και μία περιγραφή η οποία είναι σε μορφή pdf αρχείου.

Συγκεντρωτικά, διενεργούνται 7 είδη αιτημάτων:

Home, Login, TagSearch, EventDetail, PersonDetail, AddPerson, AddEvent

3.2 Αρχιτεκτονική Συστήματος

Το benchmark αποτελείται από 3 κύρια μέρη, τα οποία πρέπει να τρέχουν σε διαφορετική μηχανή: Τον εξυπηρετητή, τους χρήστες και την βάση δεδομένων. Η αλληλεπίδραση αυτών λαμβάνει χώρα ως εξής: Οι χρήστες, οι οποίοι υλοποιούνται μέσω του Faban workload generator, στέλνουν κάποια αιτήματα στον εξυπηρετητή τα οποία δημιουργούνται δυναμικά. Ακολούθως το αίτημα επεξεργάζεται, και με την βοήθεια της βάσης δεδομένων, αποστέλλεται απάντηση στον χρήστη, ανάλογα με το αίτημα του. Κατωτέρω παρουσιάζεται μία σχηματική απεικόνιση της αρχιτεκτονικής του συστήματος.



Σχήμα 3.1: Σχηματική απεικόνιση του υλοποιημένου συστήματος.

Το System Under Test (SUT), αποτελείται από τον εξυπηρετητή, ο οποίος αλληλεπιδρά με την βάση δεδομένων, το memcached, και τα στατικά αποθηκευμένα αρχεία (εικόνες και τα αρχεία pdf) τα οποία αναπαριστώνται με το πλαίσιο storage. Αυτό το κομμάτι του benchmark (SUT) είναι αυτό που ουσιαστικά αξιολογείται. Η προσομοίωση των χρηστών αναπαρίσταται στο σχήμα από τους clients, οι οποίοι υλοποιούνται μέσω του faban driver, ο οποίος βρίσκεται εξωτερικά του SUT και αλληλεπιδρά με τον εξυπηρετητή στέλνοντας σε αυτόν αιτήματα προς εξυπηρέτηση. Εξωτερικά του SUT βρίσκεται και το geocoder, το οποίο χρησιμοποιείται για αποκωδικοποίηση της τοποθεσίας των events, έτσι ώστε αυτά να αναπαρίστανται σε χάρτη.

3.2.1 Χαρακτηριστικά των χρησιμοποιούμενων μηχανών

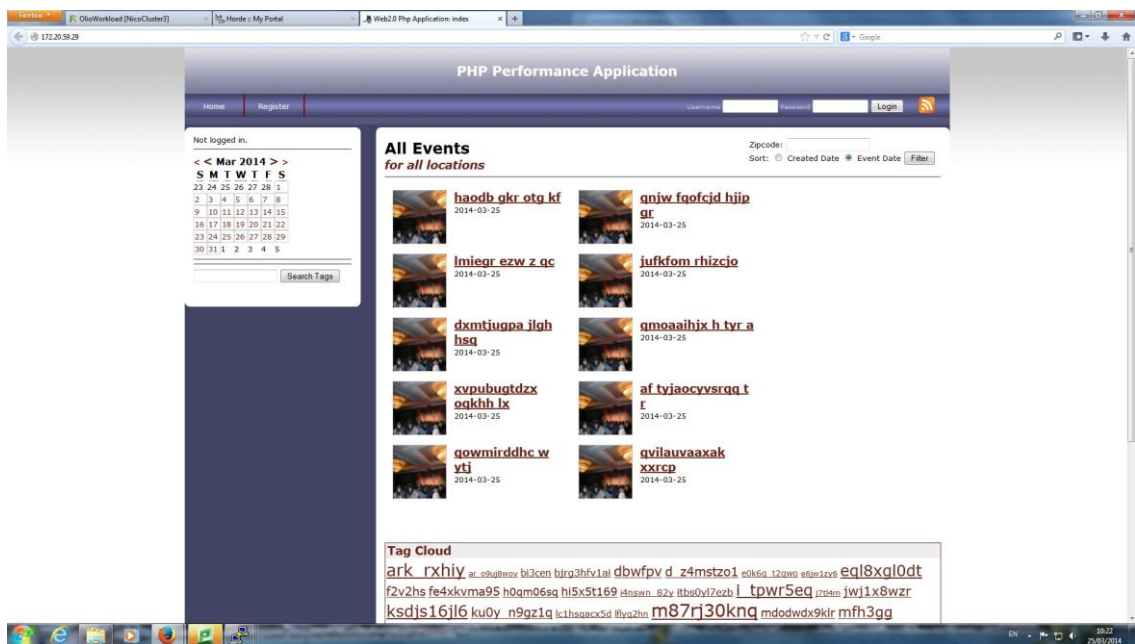
Οι 3 μηχανές που χρησιμοποιούνται για τα πειράματα είναι συνδεδεμένες μεταξύ τους με 1Gbit Ethernet. Η κάθε μηχανή περιέχει 1 Dual Socket Motherboard με δύο επεξεργαστές Intel Xeon @ 2.40 GHz , όπου ο καθένας έχει 4 cores SMT. Ο κάθε επεξεργαστής έχει 12 MB Cache και η κάθε μηχανή 32 GB Memory.

3.3 Ο Εξυπηρετητής

Ο εξυπηρετητής είναι εκείνο το κομμάτι του benchmark, το οποίο αποτελεί τον συνδετικό κρίκο όλων των συστατικών της εφαρμογής. Έχει γίνει υλοποίηση αυτού σε μία μηχανή μαζί με τα επιμέρους απαραίτητα συστατικά που απαιτούνται για να ικανοποιηθούν οι ανάγκες του Olio Application. Ακολουθεί περιγραφή των μερών.

3.3.1 Το Web Application (Olio)

Στην μηχανή του εξυπηρετητή βρίσκεται το root της εφαρμογής, όπου έχει εγκατασταθεί σε ένα φάκελο η PHP έκδοση του Olio. Μέσω του IP του εξυπηρετητή, είναι προσβάσιμο το homepage της εφαρμογής.



Εικόνα 3.1: Παρουσίαση της αρχικής σελίδας της εφαρμογής.

3.3.2 Web Server (Nginx)

Σε αυτή τη μηχανή είναι εγκατεστημένος ο Nginx web server, ο οποίος εξυπηρετεί τα εισερχόμενα αιτήματα. Οι χρήστες που μπορούν να εξυπηρετούνται ανά δευτερόλεπτο από τον εξυπηρετητή, εξαρτάται από 2 παραμέτρους οι οποίοι ρυθμίζονται από το σχετικό configuration file. Τα worker_processes αναπαριστούν τις διεργασίες που εξυπηρετούν τους χρήστες. Κάθε worker_process τρέχει σε ένα available core. Η άλλη παράμετρος είναι τα worker_connections, το οποίο καθορίζει τον μέγιστο αριθμό συνδέσεων που μπορούν να ανοιχθούν από ένα worker_process. Το γινόμενο των δύο αυτών παραμέτρων αποτελεί και το συνολικό αριθμό διαθέσιμων συνδέσεων από τον εξυπηρετητή. Είναι σημαντικό να αναφερθεί ότι η παράμετρος worker_connections δεν πρέπει να υπερβαίνει τον μέγιστο αριθμό ανοιχτών αρχείων για κάθε process καθώς επίσης ούτε και τον μέγιστο αριθμό των sockets που μπορούν να ανοιχθούν.

3.3.3 PHP

Σε αυτή την μηχανή έχει εγκατασταθεί η PHP για να υποστηρίζεται το FastCGI Process Manager (FPM), το οποίο χρειάζεται για την παραγωγή του δυναμικού περιεχομένου που απαιτείται από την εφαρμογή Olio.

3.3.4 APC (Alternative PHP Cache)

Ένα άλλο framework το οποίο είναι σημαντικό για την απόδοση είναι το APC, το οποίο αποθηκεύει το αποτέλεσμα του PHP bytecode compiler στην shared memory, για μελλοντική χρήση από άλλα requests.

3.3.5 Filestore

Στην μηχανή του εξυπηρετητή υπάρχει το directory υπό την ονομασία filestore, στο οποίο είναι αποθηκευμένα στον δίσκο όλα τα καταχωρημένα στατικά αρχεία της εφαρμογής. Περιέχει τα images και thumbnails των ατόμων και των events, καθώς επίσης και τις περιγραφές των events υπό τη μορφή pdf αρχείου.

FILESTORE CONTENTS	ΠΟΣΟΤΗΤΑ	ΜΕΓΕΘΟΣ (KB)
Person Image	1 για κάθε άτομο	670
Person Thumbnail	1 για κάθε άτομο	7
Event Image	1 για κάθε event	670
Event Thumbnail	1 για κάθε event	7
Event Literature (pdf)	1 για κάθε event	130

Πίνακας 3.1: Τα είδη των αρχείων που είναι αποθηκευμένα στο filestore.

3.4 Βάση Δεδομένων

3.4.1 MySQL Database

Η βάση δεδομένων MySQL είναι υλοποιημένη σε μια δεύτερη μηχανή και περιέχει τα στοιχεία των καταχωρημένων χρηστών και των events. Η φόρτωση της βάσης δεδομένων γίνεται από ένα script, το οποίο προσφέρεται από το cloudsuite για αυτό το benchmark. Ακολουθεί πίνακας με τα tables της βάσης δεδομένων.

TABLE	ΠΕΡΙΕΧΟΜΕΝΑ
ADDRESS	Περιέχει τα στοιχεία σχετικά με τις διευθύνσεις των ατόμων
COMMENTS_RATING	Περιέχει τα σχόλια και τα ratings για όλα τα events
PERSON	Περιέχει τις πληροφορίες για τα άτομα όπως όνομα και επίθετο
PERSON_PERSON	Οι σχέσεις φιλίας μεταξύ των ατόμων
PERSON_SOCIALEVENT	Οι σχέσεις που παρουσιάζουν τα άτομα που θα παραστούν σε συγκεκριμένο event
SOCIALEVENT	Πληροφορίες σχετικά με ένα event
SOCIALEVENTTAG	Τα tags τα οποία έχουν τα events προς κάποια άτομα
SOCIALEVENTTAG_SOCIALEVENT	Τα tags τα οποία έχουν τα events προς κάποια άλλα events

Πίνακας 3.2: Τα tables της βάσης δεδομένων

3.4.2 Apache Tomcat

Σε αυτή την μηχανή έχει υλοποιηθεί το Catalina servlet container του Apache Tomcat, το οποίο είναι υπεύθυνο για τον έλεγχο των προσβάσεων στο σύστημα από τους χρήστες, διατηρώντας μία μορφή βάσης δεδομένων με username και password. Αυτή η μέθοδος υλοποιεί το authentication για τους χρήστες, καθορίζοντας ποιες ενέργειες μπορούν να επιτελέσουν.

3.4.3 Geocoder Emulator

Το Geocoder Emulator χρησιμοποιείται από το benchmark για την αποκωδικοποίηση των τοποθεσιών, έτσι ώστε αυτές να εμφανίζονται σε χάρτη. Αυτή η λειτουργία βρίσκει εφαρμογή όταν ένας χρήστης επιλέγει να δει τις λεπτομέρειες για ένα event. Η σελίδα που ανοίγει παρουσιάζει και έναν χάρτη με τις συντεταγμένες που είναι αποθηκευμένες για το συγκεκριμένο event στην βάση δεδομένων.

3.5 Προσομοίωση Χρηστών

Ο Faban Driver είναι εγκατεστημένος σε μία 3^η μηχανή, ξεχωριστά από τον εξυπηρετητή και την βάση δεδομένων. Μέσω του web interface του driver, δημιουργούνται τα αιτήματα, τα οποία αποστέλλονται προς τον εξυπηρετητή.

3.6 Εγκατάσταση των επιμέρους συστημάτων

Το κάθε επιμέρους κομμάτι του benchmark (εξυπηρετητής, faban driver και βάση δεδομένων), έχουν εγκατασταθεί όπως περιγράφεται από το Cloudsuite. Το κάθε κομμάτι έχει εγκατασταθεί σε ξεχωριστή μηχανή, ούτως ώστε να έχουν νόημα τόσο οι προσομοιώσεις, όσο και τα αποτελέσματα. Χρησιμοποιήθηκαν 3 μηχανές του Πανεπιστημίου Κύπρου, στις οποίες εγκαταστάθηκαν και τα επιπλέον προγράμματα και υπηρεσίες ώστε να μπορέσει να υποστηριχθεί η λειτουργία όλων των συστημάτων. Αφού είχε ολοκληρωθεί η εγκατάσταση, ακολούθησε η ενημέρωση της βάσης δεδομένων και του filestore, με χρήση του dbloader.sh και fileloader.sh αντίστοιχα, τα οποία προσφέρονται από το benchmark. Το load scale για την ενημέρωση της βάσης και του filestore, ήταν 250. Αυτό το scaling μεταφράζεται σε 250*100 μοναδικούς χρήστες, ήτοι 25,000.

3.7 Πρόσβαση και αλληλεπίδραση των επιμέρους συστημάτων

3.7.1 Πρόσβαση στον εξυπηρετητή (web server)

Ο εξυπηρετητής είναι εγκατεστημένος σε μία μηχανή, μαζί με το web application και το filestore. Μέσω του IP της μηχανής αυτής, είναι προσβάσιμο το Olio Application το οποίο είναι ουσιαστικά η εφαρμογή μέσω της οποίας δίνεται νόημα στα αιτήματα που λαμβάνει ο εξυπηρετητής. Η χρήση της εφαρμογής γίνεται μέσω της ιστοσελίδας που παρουσιάζεται με κλήση του συγκεκριμένου IP Address του εξυπηρετητή από ένα πλοηγό. Αρχικά εμφανίζεται το homepage, μέσω της οποίας μπορούν να διενεργηθούν όλα τα αιτήματα που περιγράφονται από την εφαρμογή και που χρησιμοποιούνται για αξιολόγηση αυτού του benchmark.

3.7.2 Πρόσβαση στο Faban Driver

Ο Faban Driver, προσφέρει ένα web interface στο οποίο μπορεί να γίνει πρόσβαση μέσω του IP της μηχανής στην οποία είναι εγκατεστημένος, σε συνδυασμό με συγκεκριμένο port. Από αυτό το web interface, γίνονται οι κατάλληλες ρυθμίσεις για το πείραμα που θα διενεργηθεί.

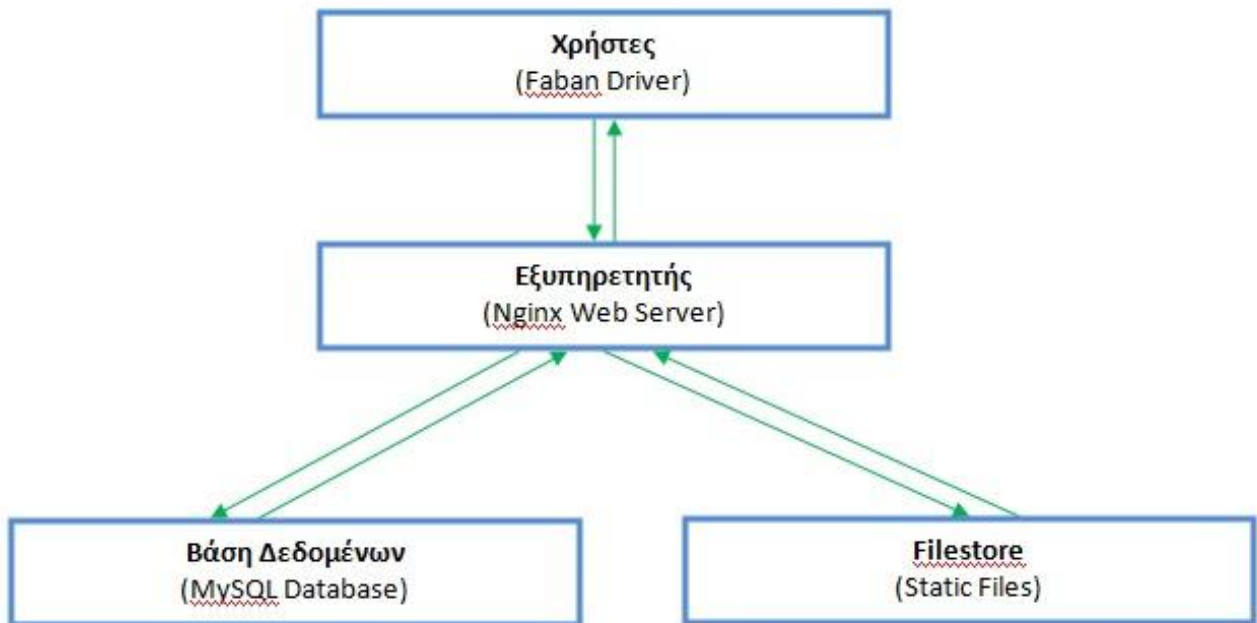
3.7.3 Πρόσβαση στη βάση δεδομένων

Η πρόσβαση στη βάση δεδομένων είναι δυνατή μέσω του command line της μηχανής στην οποία είναι εγκατεστημένη η MySQL.

3.7.4 Αλληλεπίδραση των επιμέρους συστημάτων

Τα 3 συστήματα τα οποία εγκαταστάθηκαν και ενημερώθηκαν με δεδομένα όπως περιγράφεται ανωτέρω, έπρεπε να είναι σε θέση να αλληλεπιδράσουν, ούτως ώστε να μπορούν να διενεργούνται τα πειράματα. Η αλληλεπίδραση αυτή επιτεύχθηκε τόσο με την λειτουργία ssh, καθώς και με χρήση των IP των μηχανών στα configuration files των επιμέρους συστατικών. Επίσης στο web interface του faban driver, εισάγονται τα στοιχεία για την πρόσβαση στις δύο άλλες μηχανές όπου βρίσκονται ο εξυπηρετητής και η βάση δεδομένων.

3.8 Περιγραφή της ροής των αιτημάτων



Σχήμα 3.2: Σχηματική αναπαράσταση της ροής αιτημάτων

1. Οι χρήστες διενεργούν αριθμό αιτημάτων από τα 7 διαθέσιμα στον frontend server.
2. Ανάλογα με το διενεργηθέν αίτημα ο εξυπηρετητής ανταποκρίνεται ως ακολούθως:
 - **Home:** Ο εξυπηρετητής αποστέλλει την αρχική σελίδα της εφαρμογής.
 - **Login:** Ο εξυπηρετητής διενεργεί έλεγχο στην βάση δεδομένων για εξακρίβωση των στοιχείων που εισήχθησαν και ακολούθως επιτρέπει την είσοδο του χρήστη.
 - **TagSearch:** Διενεργείται έλεγχος στην βάση δεδομένων για εύρεση των events στα οποία το συγκεκριμένο πεδίο που εισήχθηκε για έλεγχο είναι tagged. Επιστρέφεται το σύνολο εκείνων των events.
 - **EventDetail:** Λαμβάνονται τα στοιχεία του συγκεκριμένου event από την βάση δεδομένων και ακολούθως δημιουργείται η σελίδα με την χρήση php, η οποία περιλαμβάνει τα στοιχεία που περιέχει η βάση δεδομένων, τα στατικά αρχεία που περιλαμβάνουν την περιγραφή του event (pdf file) και η εικόνα (image), καθώς και ένας χάρτης με την τοποθεσία του event, ο οποίος δημιουργείται με την βοήθεια του Geocoder Emulator.

- **PersonDetail:** Λαμβάνονται τα στοιχεία του συγκεκριμένου ατόμου από την βάση δεδομένων και ακολούθως δημιουργείται η σελίδα με την χρήση php, η οποία περιλαμβάνει τα στοιχεία που περιέχει η βάση δεδομένων και το στατικό αρχείο της εικόνας του ατόμου από το filestore.
- **AddPerson:** Ο εξυπηρετητής ζητά από τον χρήστη τα στοιχεία του ατόμου, τα οποία εισάγονται στην βάση δεδομένων, καθώς μία εικόνα η οποία αποθηκεύεται στο filestore, μαζί με ένα thumbnail.
- **AddEvent:** Ο εξυπηρετητής ζητά από τον χρήστη τα στοιχεία του event, τα οποία εισάγονται στην βάση δεδομένων. Η περιγραφή του event αποθηκεύεται στο filestore υπό την μορφή pdf, όπως επίσης η εικόνα και το thumbnail του συγκεκριμένου event.

Τα πιο πάνω αιτήματα δημιουργούνται από τον Faban Driver και αποστέλλονται στον εξυπηρετητή. Με αυτό τον τρόπο προσομοιώνονται οι χρήστες. Τα περιεχόμενα της βάσης δεδομένων και του filestore έχουν δημιουργηθεί μέσω 2 script τα οποία προσφέρονται από το Cloudsuite. Τα περιεχόμενα αυτά δεν έχουν κανένα νόημα και είναι τυχαίες συμβολοσειρές που χρησιμοποιούνται μόνο για σκοπούς των πειραμάτων.

Ο Faban Driver έχει πρόσβαση στην βάση δεδομένων και χρησιμοποιεί τα απαιτούμενα στοιχεία για να δημιουργήσει τα αιτήματα που αποστέλλονται στον εξυπηρετητή. Συγκεκριμένα, με την χρήση των συνθηματικών username και password για την βάση olio, λαμβάνει τα στοιχεία που χρειάζονται για κάθε αίτημα. Επί παραδείγματι, για το login λαμβάνει από το table PERSON τα συνθηματικά username και password ενός τυχαίου ατόμου, τα οποία αργότερα εισάγει στο πεδίο login της αρχικής σελίδας του Olio Application. Επίσης για τα αιτήματα AddPerson και AddEvent, χρησιμοποιούνται από τον Faban επίσης τυχαίες συμβολοσειρές.

Ο αριθμός των αιτημάτων που διενεργούνται σε κάθε πείραμα εξαρτάται από τον αριθμό των ταυτόχρονων χρηστών που δηλώνεται πριν την έναρξη κάθε πειράματος. Να σημειωθεί ότι αν οι ταυτόχρονοι χρήστες που δηλωθούν είναι για παράδειγμα 25,000 αυτό δεν συνεπάγεται ότι θα διενεργούνται 25,000 ταυτόχρονα αιτήματα προς τον εξυπηρετητή. Υπάρχει συγκεκριμένη αναλογία αιτημάτων/ταυτόχρονων χρηστών,

ούτως ώστε να αντικατοπτρίζει την λειτουργία και την έννοια αυτής της εφαρμογής. Λεπτομέρειες σχετικά με αυτού του είδους τις παραμέτρους δίνονται κατά το ακόλουθο κεφάλαιο.

Κεφάλαιο 4

Πειράματα και αποτελέσματα

4.1 Μεθοδολογία πραγματοποιηθέντων πειραμάτων	19
4.2 Ανάλυση του χρόνου ανταπόκρισης αιτημάτων	22
4.3 Ανάλυση του throughput από τον εξυπηρετητή	27
4.4 Ανάλυση των δυνατοτήτων του εξυπηρετητή	28
4.5 Ανάλυση της χρήσης του δίσκου	30
4.6 Ανάλυση του συστήματος σε διαφορετικές συχνότητες	31

4.1 Μεθοδολογία πραγματοποιηθέντων πειραμάτων.

Τα πειράματα έχουν διενεργηθεί σε 3 μηχανές του nicocluster, όπου τρέχουν τα 3 μέρη του benchmark (web server, database και clients). Πριν από την έναρξη των πειραμάτων, πρέπει η βάση δεδομένων, ο εξυπηρετητής και ο Faban Workload Generator να τρέχουν το καθένα σε διαφορετική μηχανή και να είναι σε θέση να αλληλεπιδράσουν μεταξύ τους. Ακολούθως, μέσω του web interface του Faban, εισάγονται τα στοιχεία για την εκτέλεση των πειραμάτων.

4.1.1 Χρονικοί παράμετροι των διενεργηθέντων πειραμάτων:

Ramp Up: Χρόνος κατά τον οποίο το σύστημα είναι σε κατάσταση warm-up.

Steady State: Χρόνος κατά τον οποίο το σύστημα είναι σε σταθερή κατάσταση. Τα αποτελέσματα των πειραμάτων είναι τα αιτήματα που διενεργούνται κατά αυτό το χρονικό διάστημα.

Ramp Down: Cool Down του συστήματος.

Οι χρόνοι οι οποίοι επιλέχθηκαν για τις πιο πάνω παραμέτρους είναι 120-120-20 για όλα τα πειράματα. Η επιλογή έγινε μετά από δοκιμασίες, με σκοπό να επιτευχθεί ο

κατάλληλος συνδυασμός ούτως ώστε τα αποτελέσματα που λαμβάνονται κατά το steady state να είναι ορθά και με νόημα.

4.1.2 Παράμετροι αναφοράς πειραμάτων

Οι κάτωθι παράμετροι ισχύουν σε κάθε πείραμα, εκτός όπου αναφέρεται διαφορετικά, για σκοπούς αποφυγής επαναλήψεων.

Παράμετρος	Τιμή
Loaded Users	25,000
Ramp Up	120 δευτερόλεπτα
Steady State	120 δευτερόλεπτα
Ramp Down	20 δευτερόλεπτα
Reload Database	Ναι
Server Clock Frequency	1.6 GHz
Database Clock Frequency	1.6 GHz
Nginx worker_processes	1
Nginx worker_connections	900

Πίνακας 4.1: Παράμετροι αναφοράς διενεργηθέντων πειραμάτων

Οι χρήστες για τους οποίους είναι φορτωμένη η βάση δεδομένων και το filestore είναι 25,000. Αυτό το working set επιλέχθηκε έτσι ώστε να είναι memory resident, με σκοπό στην επικέντρωση της αξιολόγησης των επιμέρους συστημάτων πλην του δίσκου, ο οποίος αποτελεί bottleneck. Το μέγεθος του filestore είναι 26 GByte και τα περιεχόμενα της βάσης δεδομένων 300 Mbyte. Η μνήμη στις συγκεκριμένες μηχανές στις οποίες διενεργούνται τα πειράματα έχουν χωρητικότητα 32 Gbyte. Επίσης πριν την αρχή κάθε πειράματος, η βάση δεδομένων επαναφορτώνεται, ούτως ώστε να μην υπολογιστούν στα αποτελέσματα του πειράματος πιθανές καθυστερήσεις που προκύπτουν από εισαγωγές ατόμων και events τα οποία έχουν εισαχθεί από προηγούμενα πειράματα. **Να σημειωθεί ότι τα αιτήματα που διενεργούν οι χρήστες και εκτελούνται από τον εξυπηρετητή ανά δευτερόλεπτο είναι περίπου το 1/500 των ταυτόχρονων χρηστών.**

4.1.3 Πείραμα αναφοράς με 25,000 χρήστες

Το πείραμα αναφοράς ακολουθεί τις παραμέτρους αναφοράς. Οι ταυτόχρονοι χρήστες σε αυτό το πείραμα είναι 25,000.

	Count	Average	90th%	Maximum
Home	1557	0.021	0.04	0.189
Login	636	0.019	0.02	0.267
TagSearch	2078	0.029	0.06	0.212
EventDetail	1475	0.021	0.04	0.175
PersonDetail	147	0.034	0.06	0.098
AddPerson	48	0.103	0.12	0.271
AddEvent	121	0.124	0.14	0.275

Πίνακας 4.2: Τα αποτελέσματα του πειράματος αναφοράς. Στην πρώτη στήλη αναφέρονται τα ονόματα των αιτημάτων και στην δεύτερη ο αριθμός αυτών. Στην 3^η, 4^η και 5^η παρουσιάζονται ο μέσος χρόνος, το 90th%, και ο μέγιστος χρόνος αντίστοιχα (Οι χρόνοι είναι σε δευτερόλεπτα).

Ακολουθεί πίνακας που περιγράφει την αναλογία των αιτημάτων που παράγονται από τους χρήστες (Fabian) επί του συνολικού αριθμού αυτών. Η αναλογία ισχύει σε κάθε πείραμα και οι αριθμοί υπόκεινται σε 2% deviation.

Ποσοστό (2% deviation)	
Home	26.15%
Login	10.22%
TagSearch	33.45%
EventDetail	24.68%
PersonDetail	2.61%
AddPerson	0.84%
AddEvent	2.04%

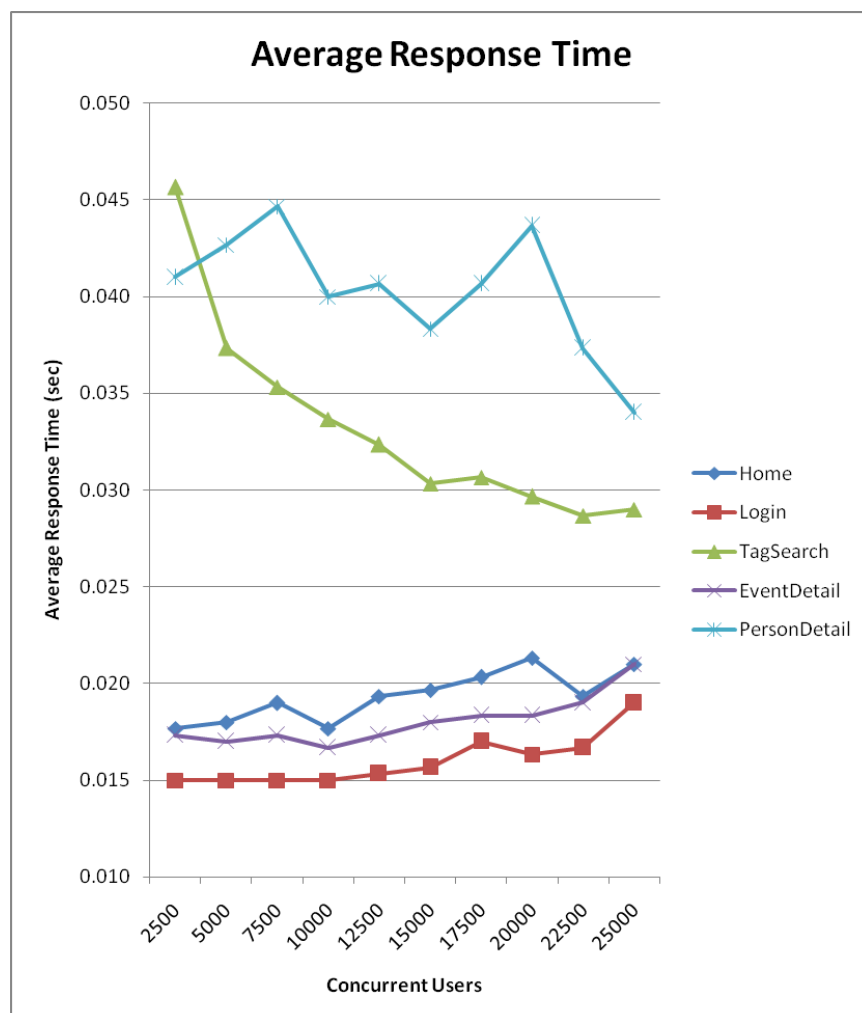
Πίνακας 4.3: Τα ποσοστά των αιτημάτων που διενεργούνται σε κάθε πείραμα.

4.2 Ανάλυση του χρόνου ανταπόκρισης αιτημάτων

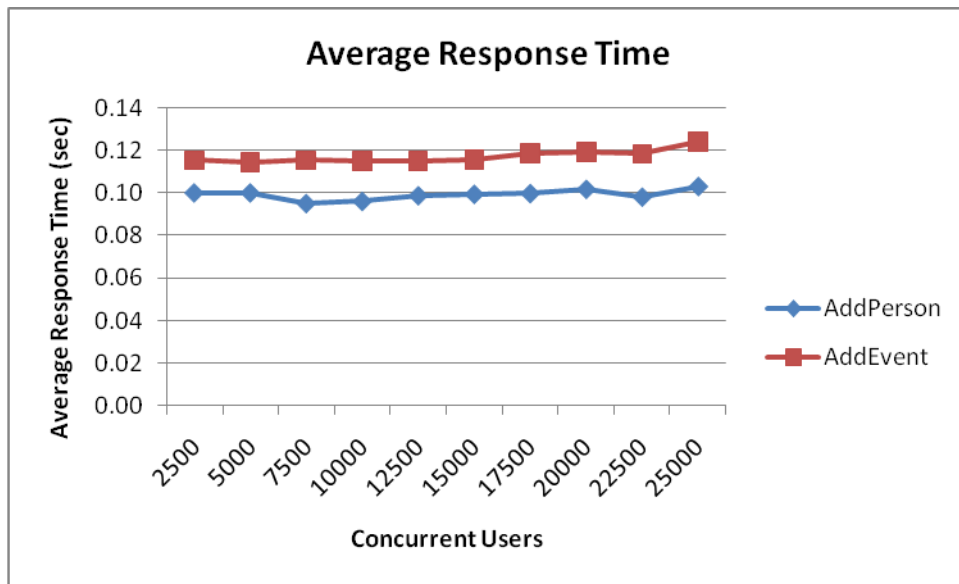
Ένας σημαντικός παράγοντας όσον αφορά το Quality of Service (QoS) σε υπηρεσίες web serving, είναι ο χρόνος ανταπόκρισης στα διενεργηθέντα αιτήματα (Response Time). Σε κάθε πείραμα υπολογίζεται από τον Faban Driver ο μέσος χρόνος ανταπόκρισης για κάθε ένα από τα 7 είδη αιτημάτων που διενεργούνται προς τον εξυπηρετητή (Homepage, Login, TagSearch, EventDetail, PersonDetail, AddPerson και AddEvent).

4.2.1 Πειραματική διαδικασία με διαφορετικό αριθμό ταυτόχρονων χρηστών

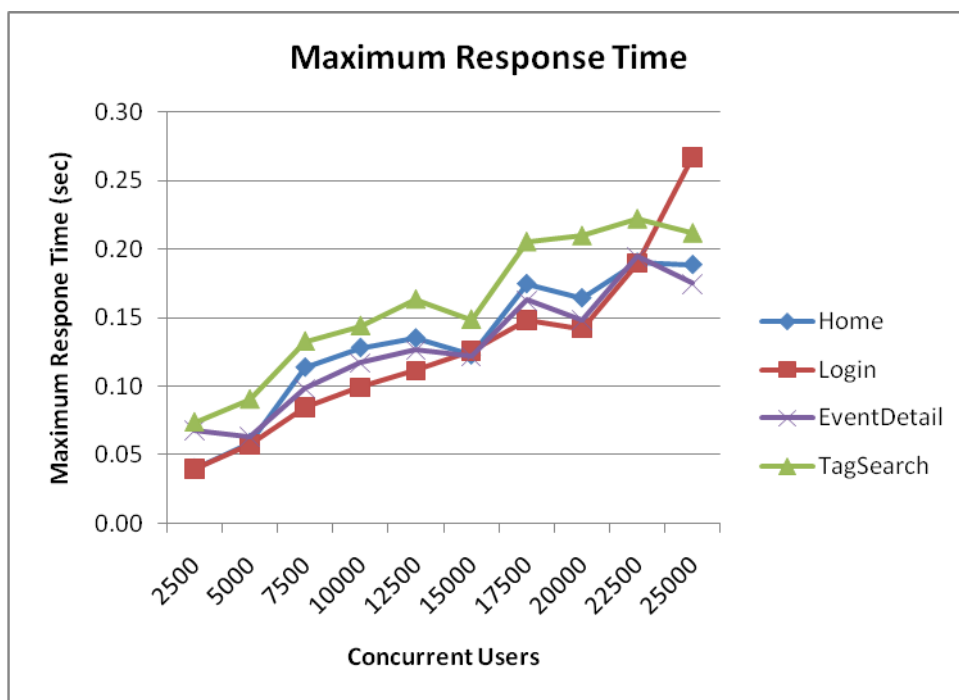
Το ακόλουθο πείραμα ακολουθεί τις παραμέτρους αναφοράς.



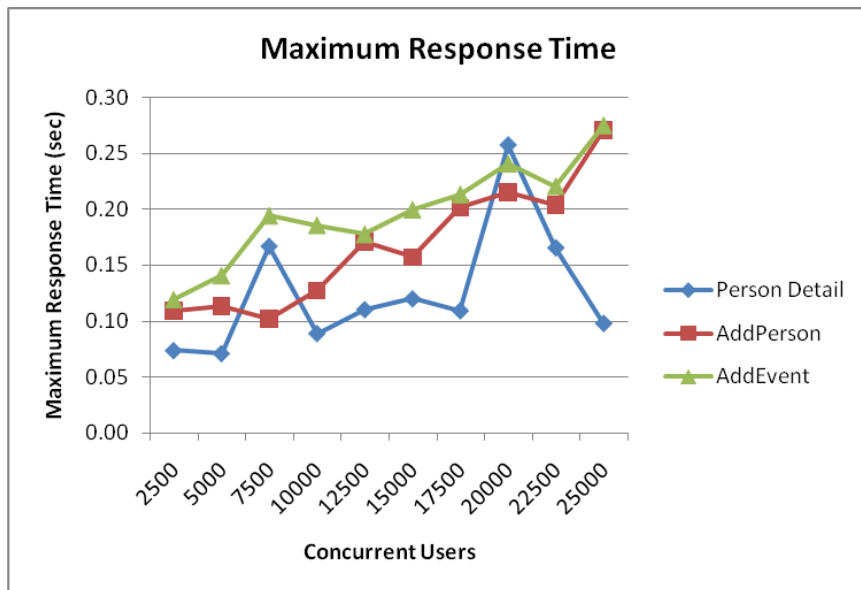
Γράφημα 4.1: Γραφική απεικόνιση του μέσου χρόνου ανταπόκρισης στα αιτήματα Home, Login, TagSearch, EventDetail και PersonDetail, ως προς τους ταυτόχρονους χρήστες.



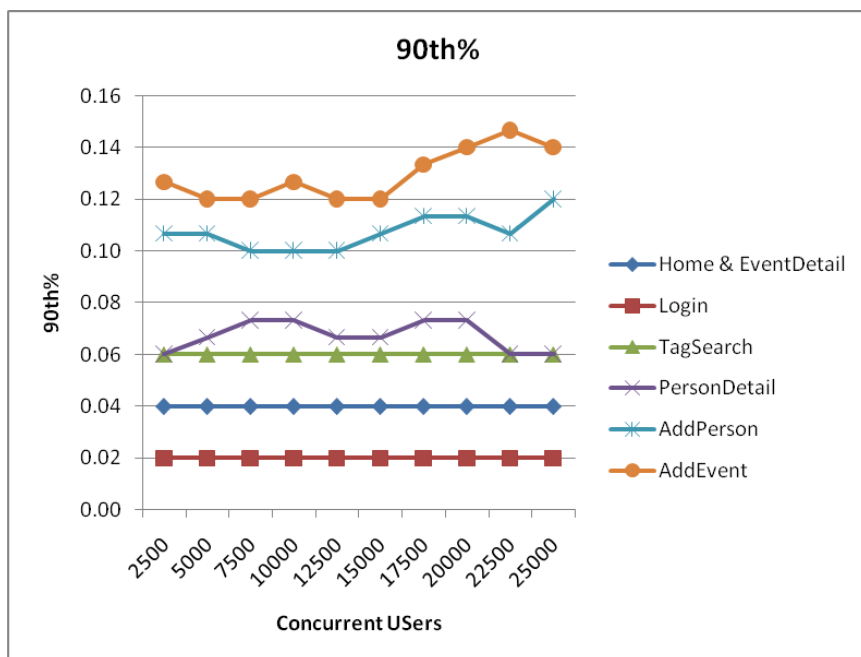
Γράφημα 4.2: Γραφική απεικόνιση του μέσου χρόνου ανταπόκρισης στα αιτήματα AddPerson και AddEvent ως προς τους ταυτόχρονους χρήστες. Παρατηρείται μία μικρή αύξηση του χρόνου με κάθε αύξηση των ταυτόχρονων χρηστών.



Γράφημα 4.3: Γραφική απεικόνιση του μέγιστου χρόνου ανταπόκρισης στα αιτήματα Home, Login, TagSearch, EventDetail και PersonDetail, ως προς τους ταυτόχρονους χρήστες. Παρατηρείται αύξηση του μέγιστου χρόνου με κάθε αύξηση των ταυτόχρονων χρηστών.



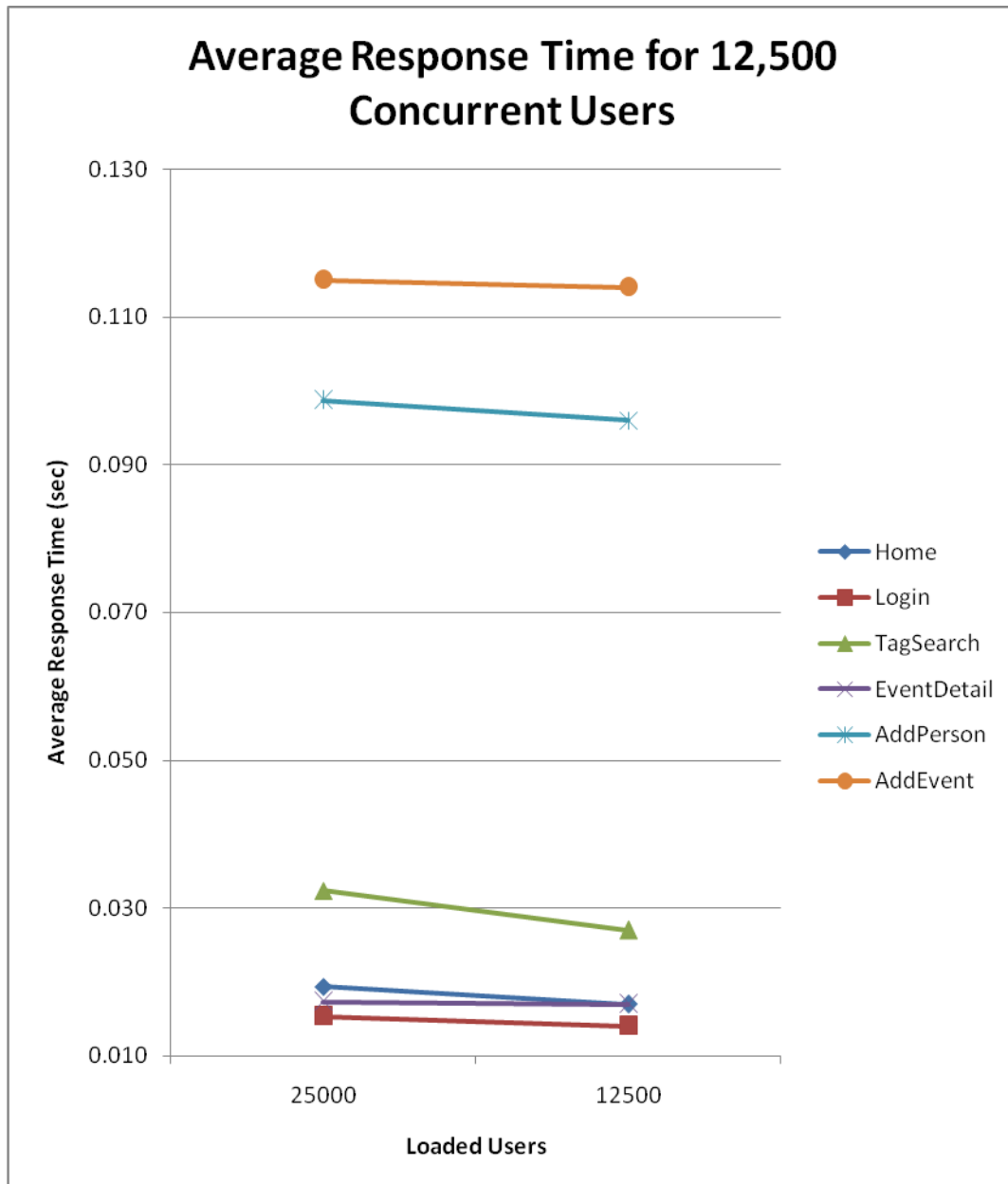
Γράφημα 4.4: Γραφική απεικόνιση του μέγιστου χρόνου ανταπόκρισης στα αιτήματα PersonDetail, AddPerson και AddEvent ως προς τους ταυτόχρονους χρήστες. Παρατηρείται αύξηση του μέγιστου χρόνου με κάθε αύξηση των ταυτόχρονων χρηστών.



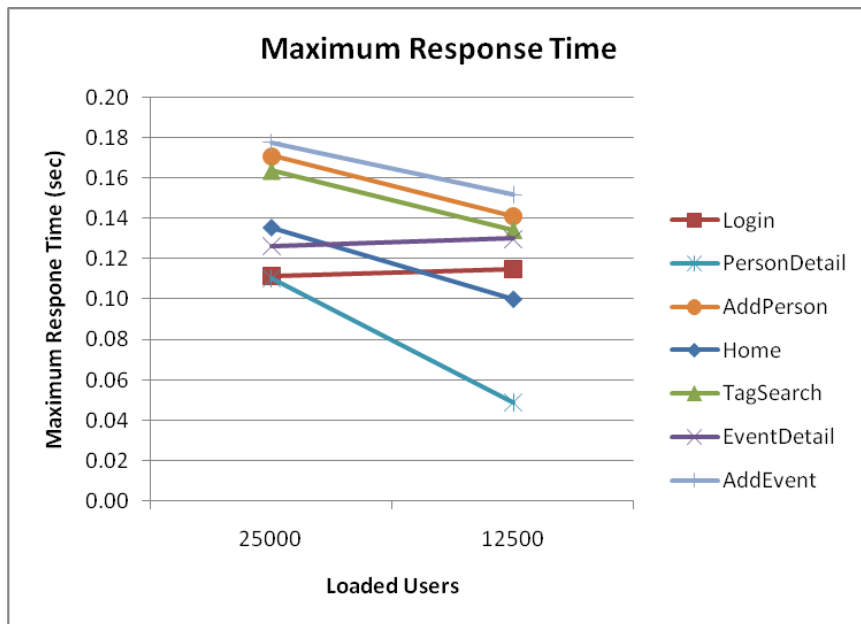
Γράφημα 4.5: Ανάλυση του 90th%, δηλαδή του χρόνου ανταπόκρισης εντός του οποίου διενεργήθηκαν το 90% των αιτημάτων, ως προς τους ταυτόχρονους χρήστες. Παρατηρείται ότι ο χρόνος αυτός δεν επηρεάζεται από τους ταυτόχρονους χρήστες για τα περισσότερα αιτήματα.

4.2.2 Πειραματική διαδικασία με διαφορετικό αριθμό αποθηκευμένων χρηστών

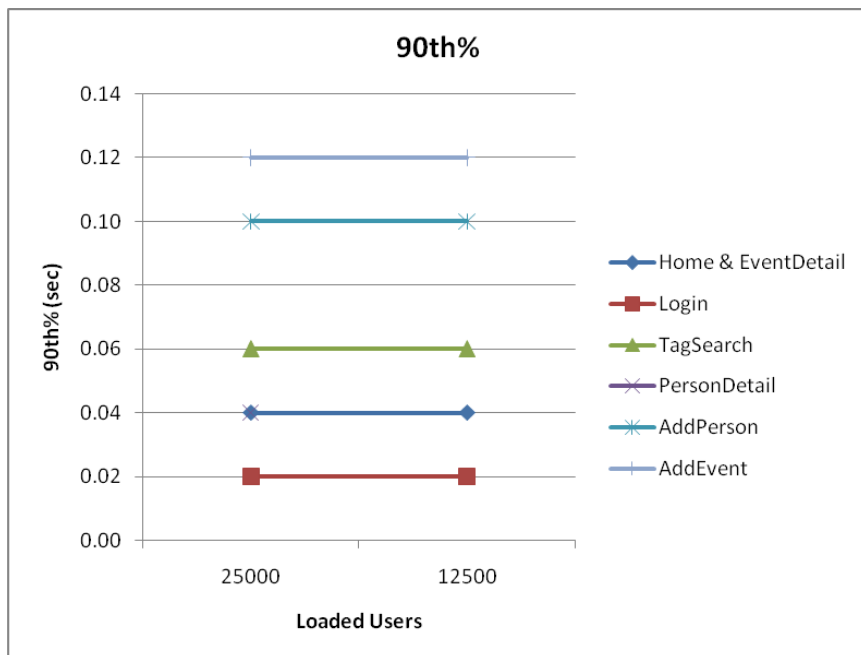
Τα ακόλουθα δύο πείραμα διενεργήθηκαν με 12,500 ταυτόχρονες χρήστες, και γίνεται σύγκριση των χρόνων ανταπόκρισης όταν οι αποθηκευμένοι χρήστες είναι 25,000 και 12,500 αντίστοιχα.



Γράφημα 4.6: Γραφική απεικόνιση του μέσου χρόνου ανταπόκρισης στα αιτήματα ως προς τους αποθηκευμένους χρήστες. Παρατηρείται μείωση στην ανταπόκριση για όλα τα αιτήματα, όταν το working set είναι το μισό.



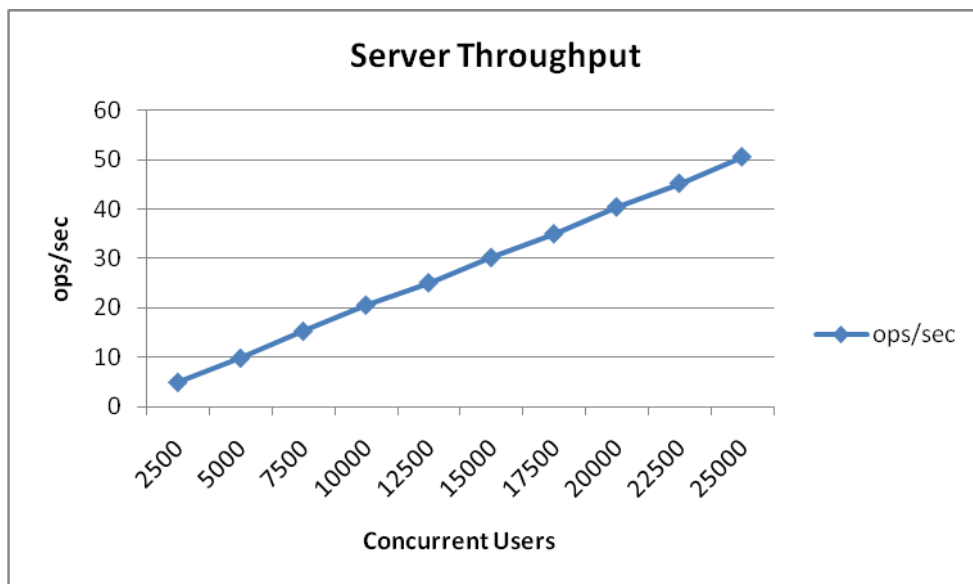
Γράφημα 4.7: Γραφική απεικόνιση του μέγιστου χρόνου ανταπόκρισης για όλα τα αιτήματα. Παρατηρείται μείωση του μέγιστου χρόνου όταν το working-set είναι το μισό, πλην των αιτημάτων EventDetail και Login, όπου παρατηρείται μικρή αύξηση.



Γράφημα 4.8: Ανάλυση του 90th%, δηλαδή του χρόνου ανταπόκρισης εντός του οποίου διενεργήθηκαν το 90% των αιτημάτων, ως προς τους ταυτόχρονους χρήστες. Παρατηρείται ότι ο χρόνος αυτός παραμένει σταθερός ανεξαρτήτως του working set.

4.3 Ανάλυση του throughput από τον εξυπηρετητή

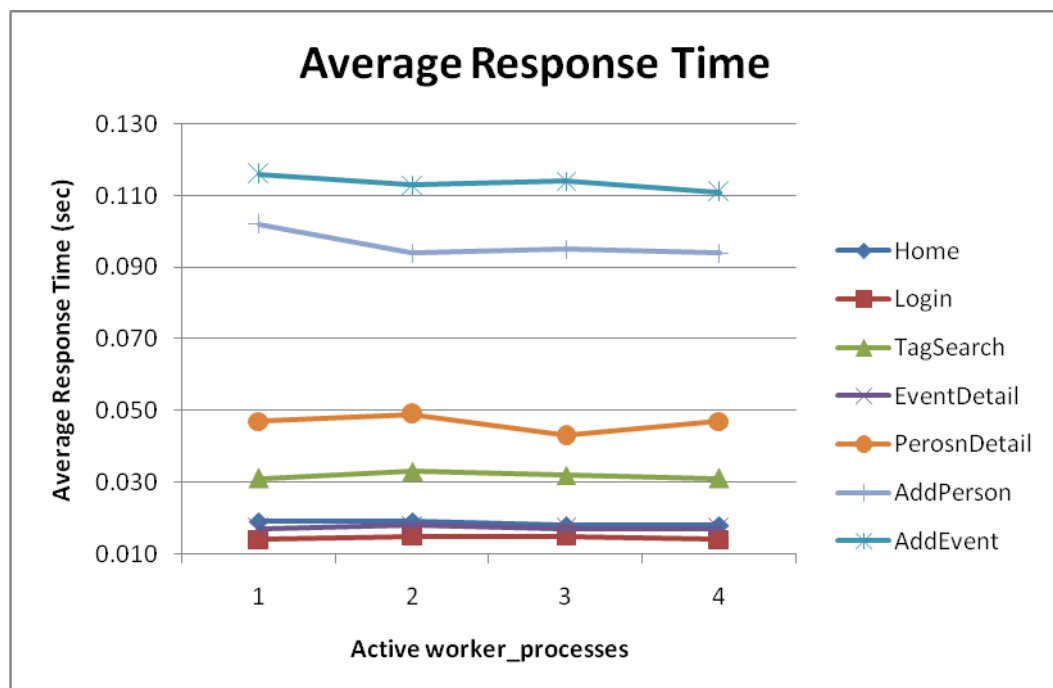
Ένα μετρικό το οποίο βοηθά στην ανάλυση της χρήσης του εξυπηρετητή, αλλά και του workload ολόκληρου, είναι το throughput από τον εξυπηρετητή, δηλαδή τα operations per second (ops/sec) τα οποία εκτελούνται. Ο αριθμός αυτός υπολογίζεται από τον συνολικό αριθμό των 7 αιτημάτων, δια τον χρόνο για τον οποίο το benchmark βρίσκεται σε steady state.



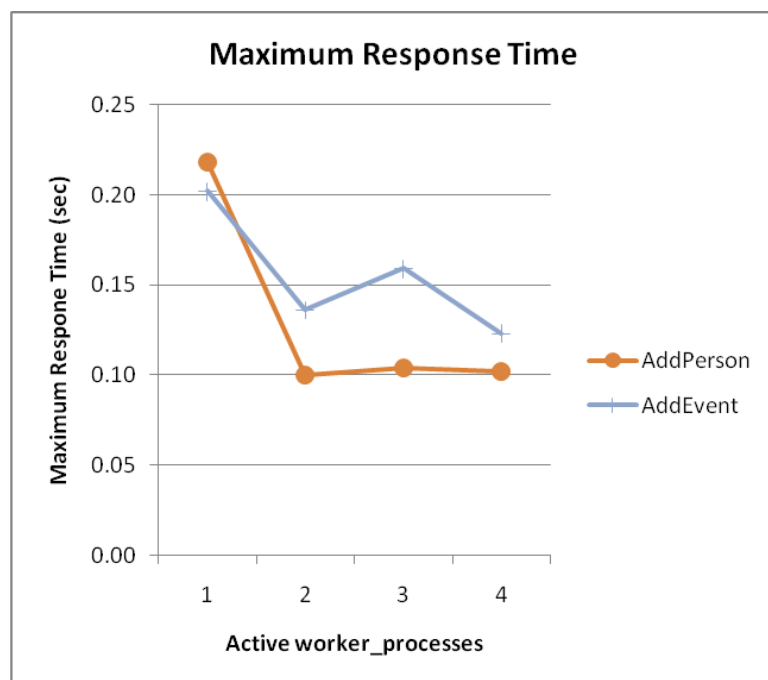
Γράφημα 4.9: Γραφική απεικόνιση του throughput του εξυπηρετητή, ως προς τους ταυτόχρονους χρήστες. Να σημειωθεί ότι η αναμενόμενη αναλογία (Concurrent Users / (ops/sec)) πρέπει να είναι μικρότερη του 500. Αυτό μεταφράζεται στο ότι τα αιτήματα που διενεργούν οι χρήστες και εκτελούνται από τον εξυπηρετητή ανά δευτερόλεπτο είναι περίπου το 1/500 των ταυτόχρονων χρηστών.

4.4 Ανάλυση των δυνατοτήτων του εξυπηρετητή

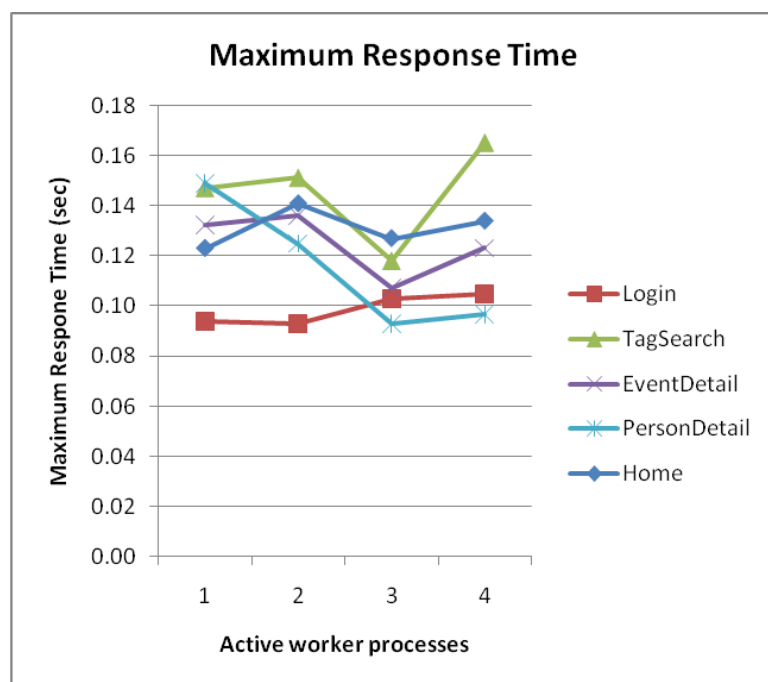
Στις ρυθμίσεις για την λειτουργία του nginx εξυπηρετητή, δίνεται η δυνατότητα για την επιλογή του αριθμού των `workers_processes` και `worker_connections`. Η παράμετρος `worker_processes` δηλώνει τον αριθμό των διεργασιών που δημιουργούνται με την έναρξη της λειτουργίας του εξυπηρετητή. Συνίσταται ο αριθμός αυτός να μην υπερβαίνει τον αριθμό των διαθέσιμων `cores`, λόγω του ότι κάποιες διεργασίες δεν θα έχουν διαθέσιμο `core` για να εκτελεστούν. Η παράμετρος `worker_connections` δηλώνει τον αριθμό των ταυτόχρονων συνδέσεων που δύναται κάθε `worker_process` να ανοίξει και να εξυπηρετήσει. Συμπερασματικά, ο ταυτόχρονος αριθμός των `clients` που μπορούν να εξυπηρετηθούν από τον εξυπηρετητή, υπολογίζεται από το γινόμενο `worker_processes` και `worker_connections`. Σε αυτή την πειραματική ενότητα, διενεργήθηκαν πειράματα με μεταβαλλόμενα `worker_processes`, ούτως ώστε να εξεταστεί η επίδραση στα αποτελέσματα των χρόνων ανταπόκρισης των αιτημάτων.



Γράφημα 4.10: Ανάλυση του μέσου χρόνου ανταπόκρισης στα αιτήματα ανάλογα με τα `worker_processes`. Να αναφερθεί ότι κάθε `worker_process` τρέχει σε ένα `core`.



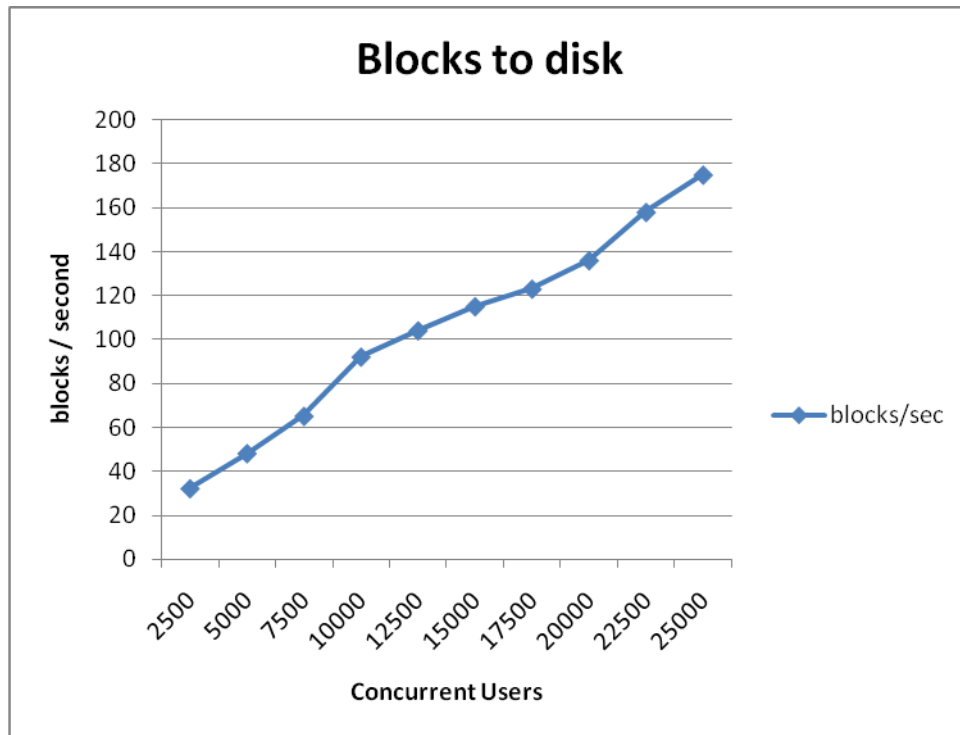
Γράφημα 4.11: Ανάλυση του μέγιστου χρόνου ανταπόκρισης στα αιτήματα ανάλογα με τα worker_processes. Είναι εμφανές ότι ο μέγιστος χρόνος ανταπόκρισης στα αιτήματα μειώνεται για τα συγκεκριμένα είδη αιτημάτων κατά την αύξηση των worker_processes από 1 σε 2, ενώ σε περαιτέρω αυξήσεις δεν παρατηρούνται σημαντικές βελτιώσεις.



Γράφημα 4.12: Ο ιδανικότερος αριθμός worker_process 3 για τα αιτήματα TagSearch, EventDetail και PersonDetail, 1 για το Home και 2 για το Login.

4.5 Ανάλυση της χρήσης του δίσκου

Έχουν χρησιμοποιηθεί το εργαλείο vmstat με σκοπό την ανάλυση της χρήσης του δίσκου. Τα στοιχεία λαμβάνονται από το συγκεκριμένο εργαλείο κάθε 10 δευτερόλεπτα όταν το πείραμα βρίσκεται σε steady state και τα αποτελέσματα που παρουσιάζονται σε αυτή την ενότητα αποτελούν τις υπολογισμένες μέσες τιμές.



Γράφημα 4.13: Ανάλυση των blocks / second τα οποία στέλνονται στον δίσκο από την εφαρμογή. Παρατηρείται ότι τα blocks αυξάνονται σχεδόν ανάλογα με την αύξηση των ταυτόχρονων χρηστών, λόγω των νέων καταχωρήσεων στην βάση δεδομένων αλλά και στο filestore που γίνονται από τα αιτήματα AddPerson και AddEvent.

Να σημειωθεί ότι δεν υπάρχουν blocks in από τον δίσκο, λόγω του ότι όλα τα διενεργηθέντα πειράματα είναι memory resident, και επομένως δεν γίνονται προσβάσεις στον δίσκο. Όλο το data set φορτώνεται στην μνήμη κατά το ramp up.

4.6 Ανάλυση του συστήματος σε διαφορετικές συχνότητες

Είναι πολύ σημαντικό για την αξιολόγηση του εξυπηρετητή, η διενέργεια πειραμάτων κάτω από διαφορετικές συχνότητες των CPU. Τα αποτελέσματα από αυτά τα πειράματα θα συντελέσουν στην εξαγωγή συμπερασμάτων ως προς την αποδοτικότητα του εξυπηρετητή, και κατά πόσον οι υψηλότερες συχνότητες συντελούν σε καλύτερη απόδοση στην ποιότητα εξυπηρέτησης. Αυτή η ενότητα είναι καθοριστικής σημασίας για την κατανάλωση ενέργειας και το απαιτούμενο υλικό.

4.5.1 Διενέργεια πειράματος

Για αυτή την ενότητα έχουν διενεργηθεί 3 πειράματα διαφορετικών συχνοτήτων. Στόχος αυτών των πειραμάτων είναι η αποσαφήνιση του κατά πόσον η συχνότητα (frequency) των κεντρικών μονάδων επεξεργασίας επηρεάζουν τον χρόνο ανταπόκρισης στα αιτήματα των χρηστών. Να σημειωθεί ότι οι αλλαγές που αναφέρονται έχουν επίδραση σε όλα τα cores της κεντρικής μονάδας επεξεργασίας.

ΠΕΙΡΑΜΑ	ΣΥΧΝΟΤΗΤΑ	ΣΥΧΝΟΤΗΤΑ ΒΑΣΗΣ
	ΕΞΥΠΗΡΕΤΗΤΗ	ΔΕΛΟΜΕΝΩΝ
1	1600 MHz	1600 MHz
2	2400 MHz	1600 MHz
3	2400 MHz	2400 MHz

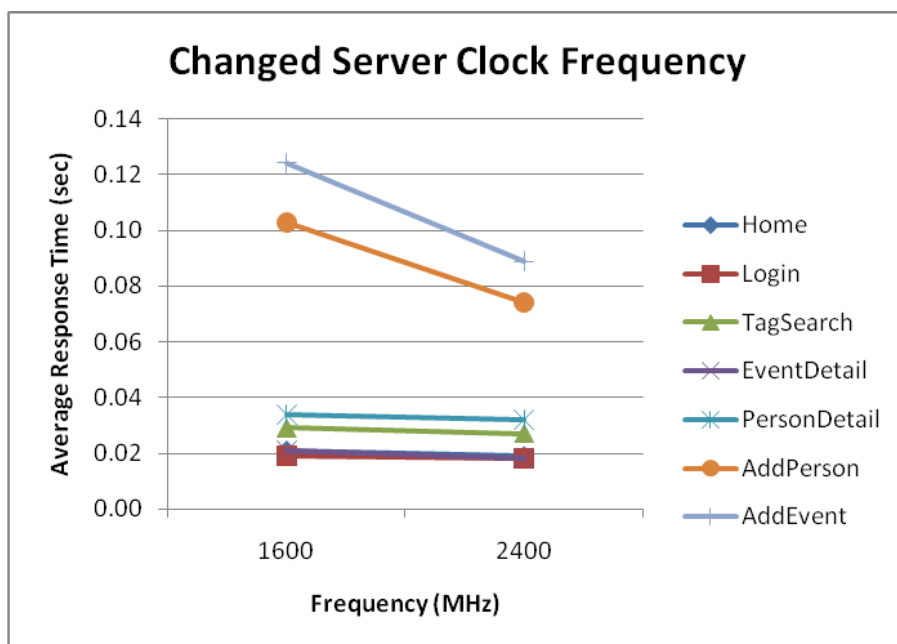
Πίνακας 4.4: Τα διενεργηθέντα πειράματα με τις αντίστοιχες συχνότητες.

4.5.2 Χαρακτηριστικά πειραμάτων

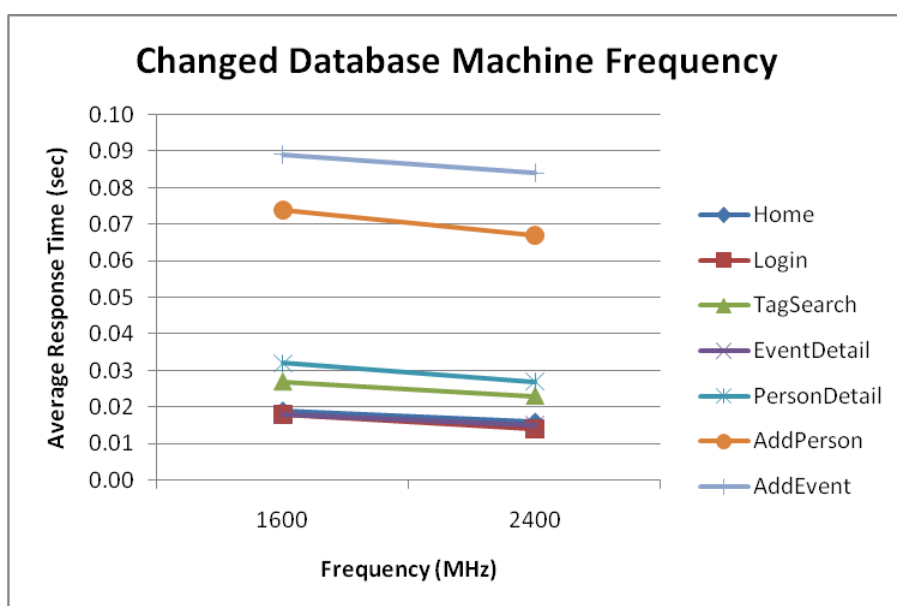
Τα χαρακτηριστικά των πειραμάτων αυτής της ενότητας ακολουθούν τα χαρακτηριστικά του baseline σεναρίου με τις εξής διαφοροποιήσεις:

Παράμετρος	Τιμή
Concurrent Users	25,000
Loaded Users	25,000
Ops/sec	~50
Steady state	120 seconds

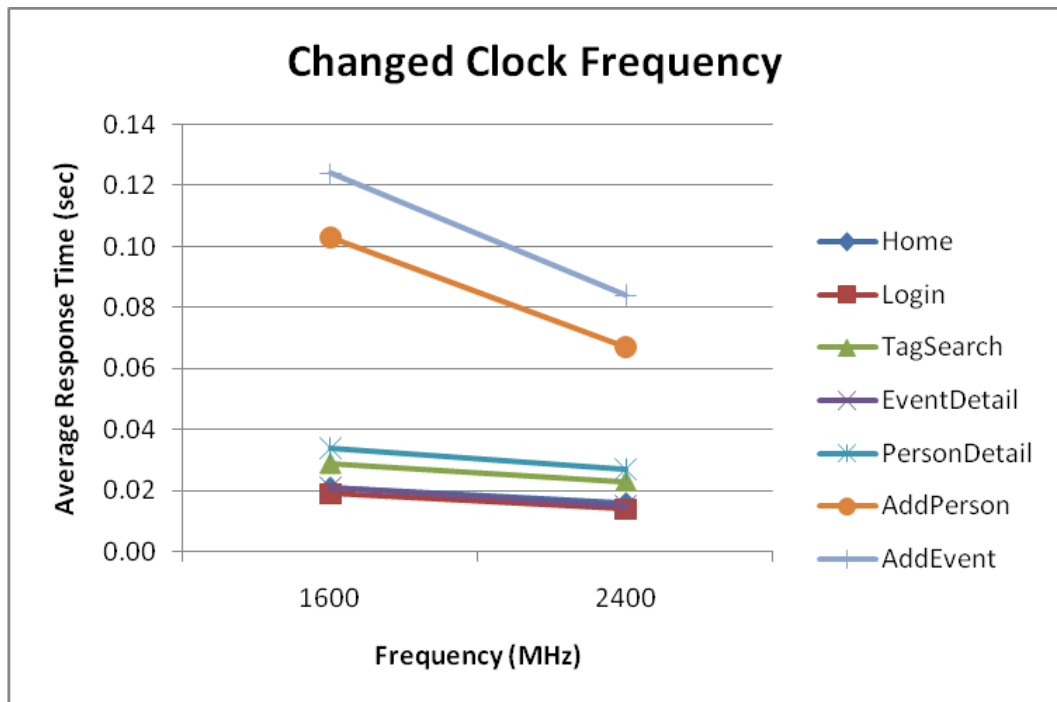
Πίνακας 4.5: Τα χαρακτηριστικά των διενεργηθέντων πειραμάτων αυτής της ενότητας.



Γράφημα 4.14: Ανάλυση του μέσου χρόνου ανταπόκρισης στα αιτήματα όπου η συχνότητα στην μηχανή όπου τρέχει η βάση δεδομένων είναι 1600 MHz και στα δύο πειράματα, ενώ η συχνότητα του εξυπηρετητή στο 1^ο πείραμα είναι 1600 MHz, ενώ στο 2^ο είναι 2400 MHz.



Γράφημα 4.15: Ανάλυση του μέσου χρόνου ανταπόκρισης στα αιτήματα όπου η συχνότητα στον εξυπηρετητή είναι 2400 MHz και στα δύο πειράματα, ενώ η συχνότητα της μηχανής όπου τρέχει η βάση δεδομένων στο 1^ο πείραμα είναι 1600 MHz, ενώ στο 2^ο είναι 2400 MHz.



Γράφημα 4.16: Ανάλυση του μέσου χρόνου ανταπόκρισης στα αιτήματα όπου η συχνότητα και στις 2 μηχανές (εξυπηρετητή και βάση δεδομένων) στο 1^ο πείραμα είναι 1600 MHz, ενώ στο 2^ο είναι 2400 MHz.

Από το γράφημα 4.14 διαφαίνεται το γεγονός ότι η αύξηση της συχνότητας της μηχανής του εξυπηρετητή, συνεπάγεται με ταυτόχρονη μείωση στον μέσο χρόνο ανταπόκρισης όλων των αιτημάτων. Σημαντική μείωση παρατηρείται στα 2 πιο χρονοβόρα αιτήματα, AddPerson και AddEvent. Αυτή η παρατήρηση οδηγεί στο συμπέρασμα ότι η συχνότητα της κεντρικής μονάδας επεξεργασίας στην οποία τρέχει ο εξυπηρετητής, συντελεί καθοριστικά στην απόδοση μειώνοντας τον χρόνο διεκπεραίωσης των αιτημάτων.

Από το γράφημα 4.15 παρατηρείται το γεγονός ότι η αύξηση της συχνότητας της μηχανής όπου τρέχει η βάση δεδομένων μαζί με την αύξηση της συχνότητας του εξυπηρετητή, συνεπάγεται με περαιτέρω μείωση στον μέσο χρόνο ανταπόκρισης σε όλα τα αιτήματα.

Το γράφημα 4.16 αποτελεί την συνολική βελτίωση που παρουσιάζεται στα γραφήματα 4.14 και 4.15, όπου η βελτίωση του χρόνου ανταπόκρισης σε όλα τα αιτήματα είναι εμφανή.

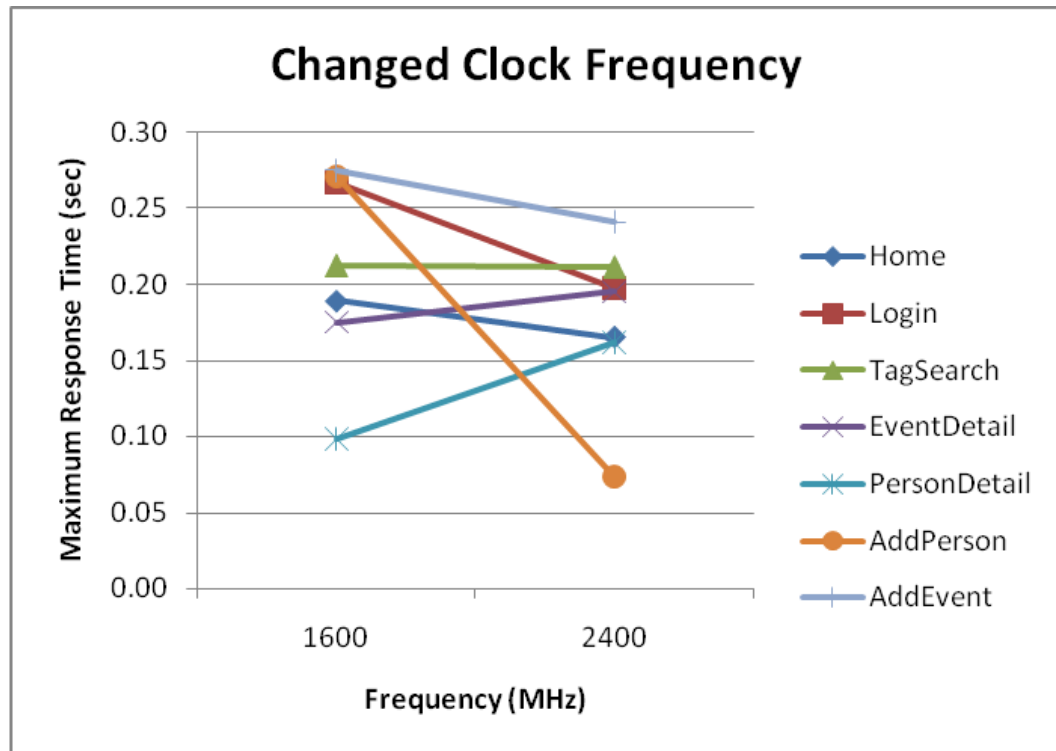
Η βελτίωση σε κάθε είδος αιτημάτων ξεχωριστά παρουσιάζεται στον πίνακα 4.6 που ακολουθεί.

	ΠΕΙΡΑΜΑ 1&2	ΠΕΙΡΑΜΑ 2&3	ΠΕΙΡΑΜΑ 1&3
Home	9,5%	15,8%	23,8%
Login	5,3%	22,2%	26,3%
TagSearch	6,9%	14,8%	20,7%
EventDetail	14,2%	16,7%	28,6%
PersonDetail	5,9%	15,6%	20,6%
AddPerson	28,2%	9,5%	34,9%
AddEvent	28,2%	5,7%	32,3%

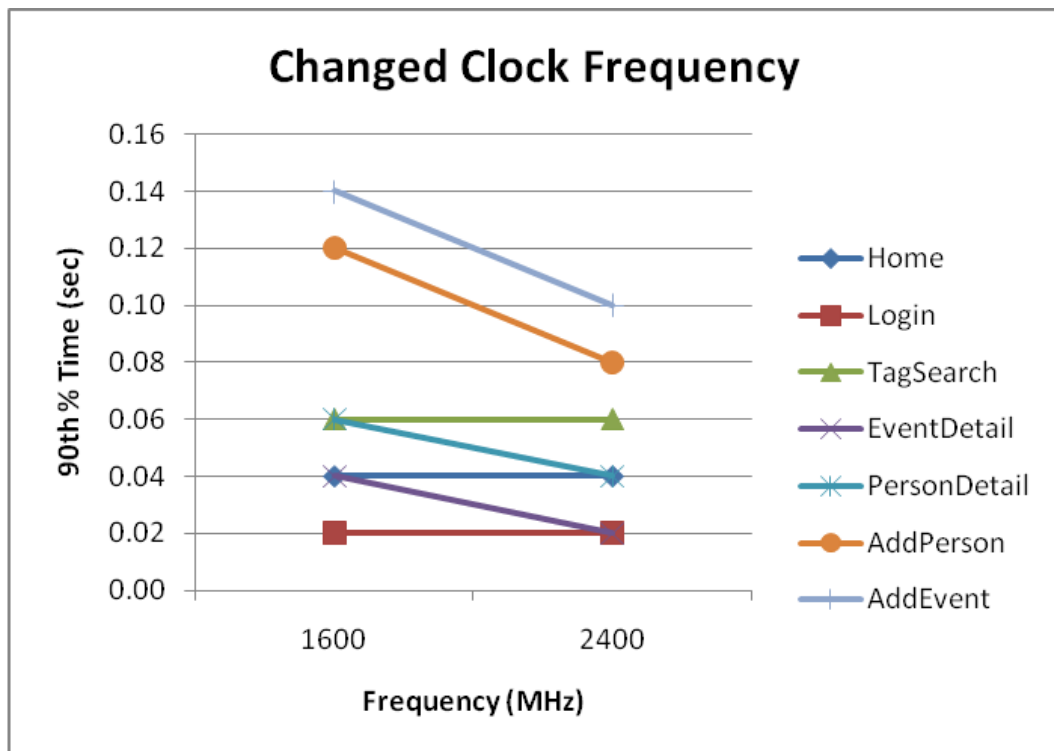
Πίνακας 4.6: Παρουσίαση των βελτιώσεων ανά αίτημα. Στην πρώτη στήλη των αποτελεσμάτων παρουσιάζεται η βελτίωση με αύξηση της συχνότητας του εξυπηρετητή, στην δεύτερη στήλη η επιπλέον βελτίωση με την αύξηση και της συχνότητας της μηχανής που τρέχει η βάση δεδομένων, ενώ στην 3^η στήλη παρουσιάζεται η βελτίωση του χρόνου και με τις 2 αυξήσεις συχνότητας.

Οι ποσοστιαίες βελτιώσεις που παρατηρούνται στον ανωτέρω πίνακα, επιβεβαιώνουν ότι με μεγαλύτερη συχνότητα ρολογιού όλα τα αιτήματα διενεργούνται από τον εξυπηρετητή σε σημαντικά μικρότερο χρόνο. Αυτό δεν σημαίνει απαραίτητα ότι οι υπηρεσίες που προσφέρουν παρόμοιο περιεχόμενο πρέπει απαραίτητα να χρησιμοποιήσουν επεξεργαστές με μεγαλύτερη συχνότητα. Όπως έχει παρατηρηθεί και από τα αποτελέσματα αυτών των πειραμάτων οι χρόνοι στους οποίους διεκπεραιώνονται όλα τα αιτήματα σε συχνότητα 1,6 GHz, είναι ήδη μικροί. Συμπερασματικά, για την επιλογή των δυνατοτήτων του επεξεργαστή, πρέπει να ληφθούν πολλοί παράγοντες υπ' όψη όπως το μέγεθος των αποθηκευμένων πληροφοριών και τον αριθμό των ταυτόχρονων χρηστών που δέχονται.

Εκτός του μέσου χρόνου ανταπόκρισης των αιτημάτων, 2 άλλα μετρικά που είναι ενδιαφέρον να μελετηθούν, είναι οι μέγιστοι χρόνοι ανταπόκρισης για κάθε αίτημα, αλλά και το 90th%.



Γράφημα 4.17: Ανάλυση του μέγιστου χρόνου ανταπόκρισης στα αιτήματα όπου η συχνότητα και στις 2 μηχανές (εξυπηρετητή και βάσης δεδομένων) στο 1ο πείραμα είναι 1600 MHz, ενώ στο 2ο είναι 2400 MHz. Είναι εμφανές ότι ο μέγιστος χρόνος ανταπόκρισης στα αιτήματα μειώνεται, εκτός το PersonDetail και EventDetail.



Γράφημα 4.18: Ανάλυση του 90th%, δηλαδή του χρόνου ανταπόκρισης εντός του οποίου διενεργήθηκαν το 90% των αιτημάτων όπου η συχνότητα και στις 2 μηχανές (εξυπηρετητή και βάσης δεδομένων) στο 1^ο πείραμα είναι 1600 MHz, ενώ στο 2^ο είναι 2400 MHz.

Από το γράφημα 4.18 γίνεται η παρατήρηση ότι στα αιτήματα EventDetail, PersonDetail, AddPerson και AddEvent ο χρόνος αυτός μειώνεται, ενώ για τα αιτήματα TagSearch, Home και Login παραμένει σταθερός.

Κεφάλαιο 5

Συμπεράσματα

5.1 Συγκεντρωτικά συμπεράσματα	37
5.2 Μελλοντική εργασία	38

5.1 Συγκεντρωτικά συμπεράσματα

Τα συμπεράσματα που έχουν εξαχθεί από την παρούσα εργασία είναι ποικίλα. Δίνεται μία σφαιρική εικόνα των αναγκών της συγκεκριμένης εφαρμογής.

Έχει παρατηρηθεί ότι με την αύξηση των ταυτόχρονων χρηστών στο σύστημα αυξάνεται ο χρόνος διεκπεραίωσης των αιτημάτων από τον εξυπηρετητή. Η μεταβολή όμως δεν είναι σταθερή για όλα τα αιτήματα, αλλά όμως είναι δυνατόν να γίνει μία ανάλυση αυτών των χρόνων. Αυτό προσδίδει περισσότερη σημασία στην έννοια του ότι πρέπει να γίνεται μία ανάλυση των υπηρεσιών που θα τρέχουν σε ένα data center, για να είναι δυνατή και η διαχείριση σε περιόδους μεγάλου φόρτου, ούτως ώστε τα διενεργηθέντα αιτήματα να εξυπηρετούνται εντός των καθορισμένων από την υπηρεσία ορίων.

Ένας άλλος σημαντικός παράγοντας είναι το τι υλικό θα επιλεγεί για να εξυπηρετήσει τις ανάγκες μίας συγκεκριμένης υπηρεσίας. Από τα διενεργηθέντα πειράματα έχει παρατηρηθεί ότι σε αυξημένες συχνότητες ρολογιού οι χρόνοι ανταπόκρισης των αιτημάτων μειώνεται. Ταυτόχρονα όμως, πρέπει να γίνει μία ανάλυση για το κατά πόσον αυτή η βελτίωση που επιτυγχάνεται αποσβένει το κόστος για την αγορά του υλικού, αλλά και για την διανομή και κατανάλωση της ενέργειας που επιφέρει η αυξημένη συχνότητα ρολογιού.

Επιπλέον έχει παρατηρηθεί ότι η αύξηση των διαθέσιμων cores δεν επέφερε βελτιώσεις στον μέσο χρόνο ανταπόκρισης, σε αντίθεση με τον μέγιστο χρόνο όπου υπήρξαν κάποιες βελτιώσεις.

Σύμφωνα με όλα τα πιο πάνω, η διαχείριση μίας τέτοιας εφαρμογής είναι ιδιαίτερα πολύπλοκη. Πρέπει να ληφθούν πολλοί παράγοντες υπ' όψη, όπως το working-set και οι ταυτόχρονοι χρήστες που αναμένεται να έχουν πρόσβαση στο σύστημα ταυτόχρονα. Με βάση αυτά θα μπορέσει να γίνει και μία εκτίμηση όσον αφορά το υλικό που θα εγκατασταθεί αλλά και τον τρόπο που αυτό θα χρησιμοποιηθεί. Λόγω του ότι οι ανάγκες μιας τέτοιας εφαρμογής αλλάζουν με τον χρόνο, είναι σημαντικό να γίνονται και οι ανάλογες τροποποιήσεις ούτως ώστε να επιτυγχάνεται όσον το δυνατόν καλύτερος χρόνος ανταπόκρισης στα αιτήματα των χρηστών, με όσο το δυνατόν χαμηλότερο κόστος υλικού αλλά και καταναλωμένης ενέργειας.

5.2 Μελλοντική εργασία

Αυτή η εργασία αποτελεί μία πολύπλευρη ανάλυση του web serving benchmark, και τα εξαγόμενα αποτελέσματα και συμπεράσματα δίνουν ώθηση για μελλοντική έρευνα. Μπορεί να γίνει ανάλυση του χρόνου που αναλώνεται σε κάθε επιμέρους κομμάτι του benchmark, ούτως ώστε να γίνουν πιο ενδελεχείς αναλύσεις και πιο συγκεκριμένες βελτιστοποιήσεις. Επίσης, μπορεί να διενεργηθούν παρόμοια πειράματα με μεγαλύτερα working sets. Επιπρόσθετα μπορεί να γίνει μία ανάλυση του κατά πόσον συγκεκριμένα αιτήματα που απαιτούν μεγαλύτερο χρόνο να διεκπεραιωθούν, να εξυπηρετούνται από εξυπηρετητή με μεγαλύτερη συχνότητα ρολογιού, με σκοπό την αύξηση της απόδοσης του χρόνου ανταπόκρισης των συγκεκριμένων αιτημάτων. Επιπλέον μπορεί να διενεργηθεί ανάλυση TCO.

Βιβλιογραφία

- [1] Boris Grot, Damien Hardy, Pejman Lotfi-Kamran, Babak Falsafi, Chrysostomos Nicopoulos, Yiannakis Sazeides, “Optimizing Data-center TCO with scale-out processors” ,2012.
- [2] Stijn Polfliet, Frederick Ryckbosch, Lieven Eeckhout, “Studying Hardware and Software Trade-Offs for a Real-Life Web 2.0 Workload”, 2012
- [3] Daniel Wong, Murali Annavaram, “KnightShift: Scaling the Energy Proportionality Wall Through Server-Level Heterogeneity”, 2012
- [4] Pejman Lotfi-Kamran, Boris Grot, Michael Ferdman, Stavros Volos, Onur Kocberber, Javier Picorel, Almutaz Adileh, Djordje Jevdjic, Sachin Idgunji, Emre Ozer, and Babak Falsafi, “Scale-Out Processors”, 2012.
- [5] Michael Ferdman, Almutaz Adileh, Onur Kocberber, Stavros Volos, Mohammad Alisafae, Djordje Jevdjic, Cansu Kaynak, Adrian Daniel Popescu, Anastasia Ailamaki, and Babak Falsafi, “Clearing the Clouds”, 2012.
- [6] Stijn Polfliet, Frederick Ryckbosch, Lieven Eeckhout, “Optimizing the Datacenter for Data-Centric Workloads”, 2011
- [7] Krishna T. Malladi, Frank A. Nothaft, Karthika Periyathambi, Benjamin C. Lee, Christos Kozyrakis, Mark Horowitz, “Towards Energy-Proportional Datacenter Memory with Mobile DRAM”, 2012
- [8] Jeffrey S. Chase, Darrell C. Anderson, Prachi N. Thakar, Amin M. Vahdat, “Managing Energy and Server Resources in Hosting Centers”, 2011
- [9] Luiz André Barroso and Urs Hölzle “The Datacenter as a Computer: An Introduction to the Design of Warehouse-Scale Machines”, 2009

- [10] Jason Mars, Lingjia Tang, Robert Hundt, Kevin Skadron, Mary Lou Soffa
“Bubble-Up: Increasing Utilization in Modern Warehouse Scale Computers via
Sensible Co-location”, 2011