ΕΘΝΙΚΟ ΚΑΙ ΚΑΠΟΔΙΣΤΡΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ

ΔΙΔΑΚΤΟΡΙΚΗ ΔΙΑΤΡΙΒΗ

Αποθηκευτικά συστήματα με δυνατότητα κλιμάκωσης σε eXascale περιβάλλοντα

Χρίστος Π. Φιλιππίδης

ΑΘΗΝΑ

ΙΟΥΛΙΟΣ 2016

NATIONAL AND KAPODISTRIAN UNIVERSITY OF ATHENS

SCHOOL OF SCIENCES
DEPARTMENT OF INFORMATICS AND TELECOMMUNICATIONS

PROGRAM OF POSTGRADUATE STUDIES

PhD THESIS

Scaling storage systems for future eXascale environments

Christos P. Filippidis

ATHENS

JULY 2016

ΔΙΔΑΚΤΟΡΙΚΗ ΔΙΑΤΡΙΒΗ

Αποθηκευτικά συστήματα με δυνατότητα κλιμάκωσης σε eXascale περιβάλλοντα.

Χρίστος Π. Φιλιππίδης

ΕΠΙΒΛΕΠΩΝ ΚΑΘΗΓΗΤΗΣ: Ιωάννης Κοτρώνης, Αναπληρωτής Καθηγητής ΕΚΠΑ

ΤΡΙΜΕΛΗΣ ΕΠΙΤΡΟΠΗ ΠΑΡΑΚΟΛΟΥΘΗΣΗΣ:
Παναγιώτης Τσανάκας, Καθηγητής ΕΜΠ
Ιωάννης Κοτρώνης, Αναπληρωτής Καθηγητής ΕΚΠΑ
Ευστάθιος Χατζηευθυμιάδης, Αναπληρωτής Καθηγητής ΕΚΠΑ

ΕΠΤΑΜΕΛΗΣ ΕΞΕΤΑΣΤΙΚΗ ΕΠΙΤΡΟΠΗ

Παναγιώτης Τσανάκας,
Καθηγητής ΕΜΠ

Στάθης Χατζηευθυμιάδης,
Αναπληρωτής Καθηγητής ΕΚΠΑ

Ιωάννης Κοτρώνης,
Αναπληρωτής Καθηγητής ΕΚΠΑ

Λάζαρος Μεράκος,
Καθηγητής ΕΚΠΑ

Ηλίας Μανωλάκος,
Καθηγητής ΕΚΠΑ

Νεκτάριος Κοζύρης,
Καθηγητής ΕΜΠ

Παναγιώτης Λουρίδας,
Αναπληρωτής Καθηγητής ΟΠΑ

Ημερομηνία εξέτασης 6/07/2016

PhD THESIS

Scaling storage systems for future eXascale environments

Christos P. Filippidis

SUPERVISOR: Yiannis Cotronis, Associate Professor UoA

THREE-MEMBER ADVISORY COMMITTEE:
Panayiotis Tsanakas, Professor NTUA
Yiannis Cotronis, Associate professor UoA
Stathes Hadjiefthymiades, Associate professor UoA

SEVEN-MEMBER EXAMINATION COMMITTEE

Panayiotis Tsanakas,
Professor NTUA

Stathes Hadjiefthymiades,
Associate Professor UoA

Yiannis Cotronis,
Associate Professor UoA

Lazaros Merakos,
Professor UoA

Elias Manolakos,
Professor UoA

Nectarios Koziris,
Professor NTUA)

Panagiotis Louridas,
Associate Professor AUEB

Examination Date 6/07/2016

ΠΕΡΙΛΗΨΗ

Οι επιστημονικοί υπολογισμοί μεγάλης κλίμακας είναι εξαιρετικά απαιτητικοί με αποτέλεσμα να έχουν μεγάλες ανάγκες σε υπολογιστική ισχύ. Οι παράλληλοι υπολογισμοί και τα παράλληλα συστήματα αρχείων αναγνωρίζονται ως η μόνη εφικτή λύση σε αυτού του είδους τα προβλήματα, ενώ οι διεργασίες εισόδου/εξόδου (I/O) αποτελούν το σημαντικότερο σημείο συμφόρησης στην απόδοση των εφαρμογών. Τα προβλήματα εντείνονται καθώς οι επιδόσεις των επεξεργαστών αυξάνονται ενώ το λογισμικό και το υλικό που δομεί τις αποθηκευτικές διατάξεις δεν εξελίσσεται με αντίστοιχους ρυθμούς. Οι σημαντικότεροι παράγοντες που επηρεάζουν την απόδοση είναι ο αριθμός των παράλληλων διεργασιών που συμμετέχουν στις μεταφορές των δεδομένων, το μέγεθος της κάθε μεταφοράς καθώς και τα διάφορα I/O μοτίβα πρόσβασης.

Ένας επιπλέον σημαντικός παράγοντας που επηρεάζει την απόδοση είναι η γενικότερη αρχιτεκτονική της αποθηκευτικής υποδομής. Μια τυπική High Performance Computing (HPC) υποδομή χρησιμοποιεί ένα μικρό μέρος των διαθέσιμων κόμβων για αποθηκευτικούς σκοπούς (κόμβοι I/O). Κάθε ένας από τους κόμβους I/O διαθέτει έναν μεγάλο αριθμό σκληρών δίσκων και εφαρμόζει ένα Redundant Array of Independent Disks (RAID) σχηματισμό για την οργάνωση των αποθηκευτικών μέσων. Τα διαμοιραζόμενα συστήματα αρχείων έχουν σημαντικούς περιορισμούς όταν εφαρμόζονται σε μεγάλης κλίμακας συστήματα, επειδή το εύρος ζώνης δεν κλιμακώνει οικονομικά αλλά και γιατί η I/O κίνηση στην δικτυακή υποδομή και στους αποθηκευτικούς κόμβους μπορεί να επηρεαστεί από άλλες ξένες διεργασίες ή με την σειρά της να επηρεάσει άλλες διεργασίες.

Στοχεύοντας στην επίλυση των πιο πάνω περιορισμών αναπτύχθηκε το πλαίσιο ΙΚΑΡΟΣ ως ένας μηχανισμός που επιτρέπει να δομούνται αποθηκευτικοί σχηματισμοί on demand. Το ΙΚΑΡΟΣ επιτυγχάνει καλύτερο συντονισμό μεταξύ των πολλαπλών στρωμάτων λογισμικού στην συνολική ροή των δεδομένων (τοπική-απομακρυσμένη πρόσβαση), διατηρώντας ταυτόχρονα την αυτονομία των επιπέδων. Επιτρέπει την κλιμάκωση του διαθέσιμου εύρους ζώνης (I/O και δίκτυο) με κόστος ανάλογο με αυτό της κλιμάκωσης της χωρητικότητας των αποθηκευτικών συστημάτων. Στοχεύει στη δημιουργία υποδομών που θα απαιτούν εξαιρετικά μικρότερα ποσά ηλεκτρικής ενέργειας καθώς και στην αντιμετώπιση των προβλημάτων κλιμάκωσης των μηχανισμών μεταδεδομένων.

Το ΙΚΑΡΟΣ έχει δομηθεί ως ένα “λεπτό” στρώμα (thin layer) που έχει τη δυνατότητα να προσφέρει υπηρεσίες σε πολλαπλά επίπεδα. Μπορεί να χρησιμοποιήσει ένα πολύ μεγάλο αριθμό αποθηκευτικών κόμβων και επιτρέπει την απομόνωση των λειτουργιών I/O μίας διεργασίας από τις αντίστοιχες των άλλων διεργασιών, στοχεύοντας στην μέγιστη αξιοποίηση των διαθέσιμων πόρων. Προσφέρει άμεση πρόσβαση σε κάθε αποθηκευτικό I/O κόμβο, ανεξάρτητα από την βαθμίδα (Tier) στην οποία ενεργεί. Κάθε βαθμίδα παρέχει πρόσβαση σε πολλαπλά υπολογιστικά κέντρα και υποστηρίζει μια συγκεκριμένη ομάδα υπηρεσιών. Κατά αυτόν τον τρόπο επιτυγχάνεται η διαχείριση της συνολικής ροής των δεδομένων (τοπική και απομακρυσμένη πρόσβαση) στο επίπεδο του δικτύου ελαχιστοποιώντας την χρήση του λειτουργικού συστήματος.

Η προσέγγιση που ακολουθεί το ΙΚΑΡΟΣ επιτρέπει σημαντική μείωση του συνολικού κόστους παρέχοντας ταυτόχρονα μεγαλύτερη ευελιξία στις υποδομές δημιουργώντας στο, επίπεδο του χρήστη, έναν ενιαίο αποθηκευτικό σχηματισμό που μπορεί να αποτελείται από όλες τις διαφορετικές υποδομές (Grids, Clouds, HPCs, Data Centers

και τοπικές συστοιχίες υπολογιστών) που χρησιμοποιεί το εκάστοτε υπολογιστικό μοντέλο.

Με τη χρήση του ΙΚΑΡΟΣ επιτυγχάνεται η μείωση του ανταγωνισμού, για ανεύρεση πόρων, μεταξύ δικτύου και αποθηκευτικών μέσων εφαρμόζοντας συντονισμένες παράλληλες μεταφορές δεδομένων στην συνολική ροή της μεταφοράς (τοπική-απομακρυσμένη πρόσβαση), καθώς και η βελτίωση της I/O απόδοσης κατά 33% με το 1/3 των διαθέσιμων σκληρών δίσκων.

Τέλος, το ΙΚΑΡΟΣ παρέχει τη δυνατότητα δημιουργίας συνεργιών μεταξύ ευρύτερων κοινοτήτων, χρησιμοποιώντας ευρέως διαδεδομένα πρότυπα και πρωτόκολλα. Αυτή η λογική δίνει την δυνατότητα να δημιουργηθούν υποδομές στις οποίες οι χρήστες θα έχουν μεγαλύτερη επιρροή στην διακυβέρνηση τους κάτι που θα επιτρέψει να τοποθετηθεί η επιστήμη των υπολογιστών και η εκμετάλλευσή των “Big Data” στο κέντρο της επιστημονικής ανακάλυψης, στοχεύοντας παράλληλα στην ανάπτυξη αποθηκευτικών συστημάτων νέας γενιάς που θα έχουν δυνατότητα κλιμάκωσης σε eXascale περιβάλλοντα.

ΘΕΜΑΤΙΚΗ ΠΕΡΙΟΧΗ: Κατανεμημένα Συστήματα

ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ: αποθηκευτικά συστήματα, συμφόρηση εισόδου/εξόδου, eXascale, συντονισμός υλικού-λογισμικού, συσκευές χαμηλής κατανάλωσης ενέργειας.

ABSTRACT

Large-scale scientific computations tend to stretch the limits of computational power with parallel computing generally recognized as the only viable solution to high performance computing problems. I/O has become a bottleneck in application performance as processor speed skyrockets, leaving storage hardware and software struggling to keep up. Parallel file systems have be developed in order to allow applications to make optimum use of available processor parallelism. The most important factors affecting performance are the number of parallel processes participating in the transfers, the size of the individual transfers and of course the access patterns.

An other important factor affecting performance is the storage architecture on which we apply the file system. A typical HPC facility uses a small portion of the available nodes for storage purposes (I/O nodes: acting as storage servers) and normally each storage server provides a huge number of hard disks through a RAID system. Current globally shared file systems, being deployed at the aforementioned facilities using current storage architectures, have several performance limitations when used with large-scale systems, because bandwidth does not scale economically to large-scale systems, I/O traffic on the high speed network can impact on and be influenced by other unrelated jobs and I/O traffic on the storage server can impact on and be influenced by other unrelated jobs.

To avoid those limitations, one approach is to configure multiple instances of smaller capacity, higher bandwidth storage closer to the compute nodes (nearby storage). The multiple instances can provide exascale size bandwidth and capacity in aggregate and can avoid much of the impact on other jobs. This approach does not provide the same file system semantics as a globally shared file system. In particular, it does not provide file cache coherency or distributed locking, but there are many use cases where those semantics are not required.

Other globally shared file system semantics are required, such as a consistent file name space, and must be provided by a nearby storage infrastructure. In cases where the usage or lifetime of the application data is constrained, a globally shared file system provides more functionality than the application requirements while at the same time limits the bandwidth which the application can use. Nearby storage as described above reverses that, providing more bandwidth but not providing globally shared file system behavior.

In the context of addressing the above limitations, IKAROS framework has been developed as a utility which enables us to create scalable storage formations on-demand. It unifies remote and local access in the overall data flow, by permitting direct access to each I/O node. In this way IKAROS manages the overall data flow at the network layer avoiding the extensive use of the operating system.

IKAROS is a data centric, energy aware platform seeking synergies between wider communities, permits ad-hoc nearby storage formations and is able to use a huge number of low spec, low power consumption I/O nodes in order to increase the available bandwidth (I/O and network) and decrease the overall power consumption. Additionally, it enables users and applications to create on demand dedicated or semi-dedicating clusters of HDDs per job to isolate the resources (storage media and I/O nodes) used by the specific job in order to increase I/O performance. By using IKAROS

we improve I/O performance by 33% with the 1/3 of the available hard disks, and we minimaze disk and network contention, by providing coordinated parallel data transfers on the overall data flow.

IKAROS enables us to connect, at the user level, all the differed storage infrastructures being used within a specific computing model (Grids, Clouds, HPCs, Data Centers and Local computing Clusters). This approach allows the creation of more user-driven computing facilities with application users and owners playing a decisive role in governance and focusing on placing computer science and the harvesting of ‘big data’ at the center of scientific discovery. In the same time, we are able to contribute towards developing the next generation storage systems capable to scale in eXascale environments.

SUBJECT AREA: Distributed Computing

KEYWORDS: storage Systems, I/O bottleneck, eXascale, hardware-software coordination, low power consumption devices.

ΕΥΧΑΡΙΣΤΙΕΣ

Θα ήθελα να ευχαριστώ τα μέλη της τριμελούς επιτροπής παρακολούθησης για την επιμονή τους να αποτυπωθεί η αλήθεια της παρούσας διατριβής.

ΠΕΡΙΕΧΟΜΕΝΑ

ΠΡΟΛΟΓΟΣ..... 27

1. ΕΙΣΑΓΩΓΗ..... 29

1.1 Big Data και το Τέταρτο Παράδειγμα..... 29

1.2 Υφιστάμενες κατανεμημένες υπολογιστικές υποδομές, παγκόσμιας κλίμακας.....30

1.3 Κίνητρο και ερευνητικές προκλήσεις.....31

1.4 Συνεισφορά διατριβής.....34

1.5 Οργάνωση διατριβής.....36

2. ΠΡΟΚΛΗΣΕΙΣ ΚΑΙ ΕΥΚΑΙΡΙΕΣ ΣΕ EXASCALE ΠΕΡΙΒΑΛΛΟΝΤΑ..... 39

2.1 Προκλήσεις στην κατανάλωση ενέργειας.....39

2.2 Ιεραρχίες αποθήκευσης και διαχείρισης δεδομένων..... 39

2.3 Ανταλλαγή δεδομένων και διαχείριση του κύκλου ζωής..... 40

2.4 Μεταδεδομένα.....41

2.5 Ευρύτερες Συνέργειες.....41

3. ΤΟ ΠΛΑΙΣΙΟ ΙΚΑΡΟΣ..... 43

3.1 Στόχοι του ΙΚΑΡΟΣ.....46

3.2 Αρχιτεκτονική του ΙΚΑΡΟΣ..... 50

4. ΟΝΤΟΤΗΤΑ ΜΕΤΑΔΕΔΟΜΕΝΩΝ ΤΟΥ ΙΚΑΡΟΣ.....55

4.1 Αρχιτεκτονική της οντότητας μεταδεδομένων..... 56

4.2 Περιπτώσεις χρήσης της οντότητας μεταδεδομένων..... 59

5. ΕΙΣΟΔΟΣ/ΕΞΟΔΟΣ ΣΤΟ ΙΚΑΡΟΣ.....63

5.1 Μηχανισμοί εισόδου/εξόδου του ΙΚΑΡΟΣ.....63

5.2 Παραμετροποίηση των αιτημάτων.....66

5.3 Ροή διεργασίας των λειτουργιών του ΙΚΑΡΟΣ Apache module.....67

5.4 Μηχανισμοί συστήματος αρχείου του ΙΚΑΡΟΣ (POSIX - Συμβατότητα).....71

5.5 Περιβάλλον εφαρμογών και μετρήσεων.....74

5.6 Πειραματικά αποτελέσματα σύγκρισης των ΙΚΑΡΟΣ, NFS, HDFS και PVFS2 (χρήση υποδομής τύπου: soho-NAS).....77

5.7 Πειραματικά αποτελέσματα σύγκρισης των ΙΚΑΡΟΣ και PVFS2 (commodity hardware)..... 85

5.8 Επικοινωνία των μηχανισμών εισόδου/εξόδου με υπηρεσίες κοινωνικής δικτύωσης, για την διαχείριση των μεταδεδομένων στο ΙΚΑΡΟΣ.....88

6. ΜΕΤΑΦΟΡΑ ΔΕΔΟΜΕΝΩΝ ΣΕ ΔΙΚΤΥΑ ΕΥΡΕΙΑΣ ΠΕΡΙΟΧΗΣ (WAN)..... 91

6.1 Πειραματικά αποτελέσματα σύγκρισης του ΙΚΑΡΟΣ με το GridFTP..... 93

6.2 Πειραματικά αποτελέσματα από την χρήση των μηχανισμών μεταφοράς δεδομένων στο δίκτυο κατανομής δεδομένων του Πειράματος CMS του LHC στο CERN.....96

7. ΙΚΑΡΟΣ, ΕΝΑ ΠΛΑΙΣΙΟ ΔΗΜΙΟΥΡΓΙΑΣ ΔΥΝΑΜΙΚΩΝ ΑΠΟΘΗΚΕΥΤΙΚΩΝ ΣΧΗΜΑΤΙΣΜΩΝ..... 99

7.1 Τεχνικά χαρακτηριστικά του ΙΚΑΡΟΣ.....99

7.2 Ανάλυση τεχνικών χαρατηριστικών της Cytera HPC υποδομής.....100

7.3 Πειραματικές μετρήσεις στη Cytera HPC υποδομή.....102

8. MOBILE GRID, ΜΙΑ ΠΛΑΤΦΟΡΜΑ ΔΗΜΙΟΥΡΓΙΑΣ ΕΥΡΥΤΕΡΩΝ ΣΥΝΕΡΓΙΩΝ.. 105

8.1 Μηχανισμοί Διαχείρισης Πόρων στο ΙΚΑΡΟΣ.....107

8.2 Πιλοτική εφαρμογή IkarosM..... 110

9. ΣΥΜΠΕΡΑΣΜΑΤΑ / ΜΕΛΛΟΝΤΙΚΗ ΕΡΓΑΣΙΑ..... 115

ΠΙΝΑΚΑΣ ΟΡΟΛΟΓΙΑΣ..... 119

ΣΥΝΤΜΗΣΕΙΣ – ΑΡΚΤΙΚΟΛΕΞΑ – ΑΚΡΩΝΥΜΙΑ..... 123

ΠΑΡΑΡΤΗΜΑ Ι (HTTP και WebDav μέθοδοι)..... 127

ΠΑΡΑΡΤΗΜΑ ΙΙ (Επεκτάσεις του FTP)..... 129

ΑΝΑΦΟΡΕΣ..... 131

ΚΑΤΑΛΟΓΟΣ ΣΧΗΜΑΤΩΝ

Σχήμα 1: Τυπικός κύκλος δημιουργίας νέας γνώσης..... 30

Σχήμα 2: Το Πλαίσιο ΙΚΑΡΟΣ.....44

Σχήμα 3: Αρχιτεκτονική ΙΚΑΡΟΣ.....51

Σχήμα 4: Ροή δεδομένων.....52

Σχήμα 5: Ιεραρχία οντότητας μεταδεδομένων ΙΚΑΡΟΣ.....57

Σχήμα 6: Επίπεδα λειτουργιών ΙΚΑΡΟΣ σε σύγκριση με ένα τυπικό σύστημα.....58

Σχήμα 7: Αυτόνομη χρήση..... 59

Σχήμα 8: ΙΚΑΡΟΣ MDS, Τοπική χρήση..... 60

Σχήμα 9: ΙΚΑΡΟΣ MDS, υβριδική χρήση.....61

Σχήμα 10: Μηχανισμοί Ανάγνωσής..... 64

Σχήμα 11: Μηχανισμοί εγγραφής, ΙΚΑΡΟΣ.....65

Σχήμα 12: ΙΚΑΡΟΣ module, διάγραμμα ροής περίπτωση 1, (ανάσυρση τμήματος αρχείου)..... 67

Σχήμα 13: ΙΚΑΡΟΣ module, διάγραμμα ροής περίπτωση 2, (διεργασία ανάγνωσης)..... 68

Σχήμα 14: ΙΚΑΡΟΣ module, διάγραμμα ροής περίπτωση 3, (διεργασίες εγγραφής).....69

Σχήμα 15: ΙΚΑΡΟΣ module, διάγραμμα ροής περίπτωση 4, (διεργασίες εγγραφής- reversed read).....70

Σχήμα 16: Μηχανισμοί συστήματος αρχείου του ΙΚΑΡΟΣ (POSIX - συμβατότητα)..... 74

Σχήμα 17: Αλυσίδα Ανάλυσης seatray.....78

Σχήμα 18: Υπολογιστικό μοντέλο KM3NeT.....78

Σχήμα 19: ΙΚΑΡΟΣ module, διάγραμμα ροής περίπτωση 4, (διεργασίες εγγραφής- reversed read)..... 91

Σχήμα 20: CMS Event Data Model..... 92

Σχήμα 21: Ανάλυση συνολικής ροής δεδομένων.....98

Σχήμα 22: Το πλαίσιο ΙΚΑΡΟΣ..... 100

Σχήμα 23: Condor, μηχανισμοί δρομολόγησης..... 106

Σχήμα 24: Διάγραμμα ροής εφαρμογής ikarosM..... 110

ΚΑΤΑΛΟΓΟΣ ΕΙΚΟΝΩΝ

Εικόνα 1: Διαθέσιμοι κόμβοι..... 58

Εικόνα 2: Κατανομή του αρχείου στους κόμβους.....59

Εικόνα 3:Παραδείγματα χρήσης του WebDav.....73

Εικόνα 4: Υπολογιστική συστοιχία Zeus.....75

Εικόνα 5: Συγκριτικές μετρήσεις ΙΚΑΡΟΣ, HDFS, PVFS2, NFS (με την χρήση 1 I/O κόμβου), πείραμα:KM3NeT, Διεργασίες εγγραφής: case 4, υποδομή: soho-NAS..... 79

Εικόνα 6: Συγκριτικές μετρήσεις ΙΚΑΡΟΣ, NFS, HDFS, PVFS2, (με την χρήση 2 I/O κόμβων), πείραμα:KM3NeT, Διεργασίες εγγραφής: case 4, υποδομή: soho-NAS..... 80

Εικόνα 7: Συγκριτικές μετρήσεις ΙΚΑΡΟΣ, NFS, HDFS, PVFS2(με την χρήση 3 I/O κόμβων), πείραμα:KM3NeT, Διεργασίες εγγραφής: case 4, υποδομή: soho-NAS..... 80

Εικόνα 8: Συγκριτικές μετρήσεις ΙΚΑΡΟΣ, NFS, HDFS PVFS2 (με την χρήση 4 I/O κόμβων), πείραμα:KM3NeT, Διεργασίες εγγραφής: case 4, υποδομή: soho-NAS..... 81

Εικόνα 9: Συγκεντρωτικές μετρήσεις ΙΚΑΡΟΣ, NFS, HDFS PVFS2, πείραμα:KM3NeT Διεργασίες εγγραφής: case 4, υποδομή: soho-NAS... 82

Εικόνα 10: Συγκριτικές μετρήσεις ΙΚΑΡΟΣ, NFS, HDFS PVFS2 (με την χρήση 4 I/O κόμβων), πείραμα:KM3NeT, Διεργασίες ανάγνωσης: case 2, υποδομή: soho-NAS..... 83

Εικόνα 11: Συγκριτικές μετρήσεις ΙΚΑΡΟΣ, NFS, HDFS PVFS2 (με την χρήση 1 I/O κόμβου, μέγεθος αρχείου 6 GB, 2 και 4 ταυτόχρονες εγγραφές), πείραμα:KM3NeT, Διεργασίες εγγραφής: case 4, υποδομή: soho-NAS.. 84

Εικόνα 12: Συγκριτικές μετρήσεις ΙΚΑΡΟΣ, NFS, HDFS PVFS2 (με την χρήση 2 I/O κόμβων, μέγεθος αρχείου 6 GB, 2 και 4 ταυτόχρονες εγγραφές), πείραμα:KM3NeT, Διεργασίες εγγραφής: case 4, υποδομή: soho-NAS.. 84

Εικόνα 13: Συγκριτικές μετρήσεις ΙΚΑΡΟΣ, NFS, HDFS PVFS2 (με την χρήση 3 I/O κόμβων, μέγεθος αρχείου 6 GB, 2 και 4 ταυτόχρονες εγγραφές), πείραμα:KM3NeT, Διεργασίες εγγραφής: case 4, υποδομή: soho-NAS.. 85

Εικόνα 14: Συγκριτικές μετρήσεις ΙΚΑΡΟΣ, NFS, HDFS PVFS2 (με την χρήση 4 I/O κόμβων, μέγεθος αρχείου 6 GB, 2 και 4 ταυτόχρονες εγγραφές), πείραμα:KM3NeT, Διεργασίες εγγραφής: case 4, υποδομή: soho-NAS.. 85

Εικόνα 15: Συγκριτικές μετρήσεις του ΙΚΑΡΟΣ με το PVFS2, στις διεργασίες ανάγνωσης (Blocksize: 100MB) πείραμα: IOR-HPC, Διεργασίες ανάγνωσης: case 2, υποδομή: commodity hardware..... 86

Εικόνα 16: Συγκριτικές μετρήσεις του ΙΚΑΡΟΣ με το PVFS2, στις διεργασίες ανάγνωσης (Blocksize: 1GB) πείραμα: IOR-HPC, Διεργασίες ανάγνωσης: case 2, υποδομή: commodity hardware..... 87

Εικόνα 17: Συγκριτικές μετρήσεις του ΙΚΑΡΟΣ με το PVFS2, στις διεργασίες εγγραφής (Blocksize: 100MB) πείραμα: IOR-HPC, Διεργασίες εγγραφής: case 4, υποδομή: commodity hardware..... 87

Εικόνα 18: Συγκριτικές μετρήσεις του ΙΚΑΡΟΣ με το PVFS2, στις διεργασίες εγγραφής (Blocksize: 1GB) πείραμα: IOR-HPC, Διεργασίες εγγραφής: case 4, υποδομή: commodity hardware..... 88

Εικόνα 19: Μεταδεδομένα ενός αρχείου του πειράματος KM3NeT, σε JSON μορφή, ως ένα Facebook note..... 88

Εικόνα 20: ΙΚΑΡΟΣ JSON meta-data object ώς Facebook note, διαθέσιμοι κόμβοι.....89

Εικόνα 21: ΙΚΑΡΟΣ, WAN Testbed..... 93

Εικόνα 22: Σύγκριση του ΙΚΑΡΟΣ με το GridFTP, HellasGrid.....94

Εικόνα 23: Σύγκριση του ΙΚΑΡΟΣ με το GridFTP, HellasGrid (4 παράλληλα κανάλια μεταφοράς)..... 94

Εικόνα 24: Σύγκριση του ΙΚΑΡΟΣ με το GridFTP, HellasGrid (8 παράλληλα κανάλια μεταφοράς)..... 95

Εικόνα 25: Σύγκριση του ΙΚΑΡΟΣ με το GridFTP, σε παγκόσμια κλίμακα.....95

Εικόνα 26: Δίκτυο κατανομής δεδομένων του Πειράματος CMS του LHC στο CERN..... 96

Εικόνα 27: Όγκος δεδομένων του πειράματος, τελικές δοκιμαστικές μετρήσεις.....97

Εικόνα 28: Ρυθμός μεταφοράς δεδομένων σε πραγματικές συνθήκες (2013) 98

Εικόνα 29: Απόδοση GPFS σε σχέση με τον αριθμό των πελατών..... 101

Εικόνα 30: Απόδοση τοπικού δίσκου σε σχέση με τα αιτήματα ανάγνωσης/εγγραφής.....102

Εικόνα 31: ΙΚΑΡΟΣ vs GPFS.....103

Εικόνα 32: Η έννοια του Mobile Grid..... 108

Εικόνα 33: IkarosM, σύνδεση στο Mobile Grid..... 111

Εικόνα 34: IkarosM, ανάσυρση διεργασίας..... 111

Εικόνα 35: IkarosM, λήψη αδιαμόρφωτων μεταδεδομένων..... 112

Εικόνα 36: IkarosM, επιστροφή αποτελεσμάτων στο ΙΚΑΡΟΣ..... 112

Εικόνα 37: XML DOM δέντρο..... 113

ΚΑΤΑΛΟΓΟΣ ΠΙΝΑΚΩΝ

Πίνακας 1: Χαρακτηριστικά λειτουργίας: ΙΚΑΡΟΣ, PVFS2,HDFS..... 45

Πίνακας 2: Τεχνικές προδιαγραφές αποθηκευτικών συστημάτων 75

Πίνακας 3: Συνολική ροή δεδομένων (WAN-LAN)..... 97

Πίνακας 4: HTTP και WebDav μέθοδοι..... 127

Πίνακας 5: Υποστήριξη πελατών σε λειτουργίες του HTTP.....128

Πίνακας 6: Υποστήριξη πελατών σε λειτουργίες του WebDav..... 128

Πίνακας 7: Επεκτάσεις του FTP, που χρησιμοποιούνται για την υλοποίηση του GridFTP, προσαρτήθηκε από [36].....129

ΠΡΟΛΟΓΟΣ

-το εύδαιμον το ελεύθερον, το δ’ ελεύθερον το εύψυχον (Θουκυδίδης: Κεφ. Β’ 43,4)

-ευδαιμονία: η γνώση της αρετής σου η καλλιέργεια και η αποδοχή της (αριστεία), συνδυαζόμενη με τη μάχη για την απόκτηση της γνώσης αυτής.

1. ΕΙΣΑΓΩΓΗ

1.1 Big Data και το Τέταρτο Παράδειγμα

Οι δυο κυρίαρχες δομές/μεθοδολογίες (paradigms) για την επιστημονική ανακάλυψη είναι ιστορικά η “θεωρία” και το “πείραμα”, με τις υπολογιστικές προσομοιώσεις μεγάλης κλίμακας να αναδύονται ως η τρίτη μεθοδολογία κατά την διάρκεια του εικοστού αιώνα. Σε πολλές περιπτώσεις, οι υπολογιστικές προσομοιώσεις μεγάλης κλίμακας εξαρτώνται σε μεγάλο βαθμό από τα δεδομένα και τον χειρισμό αυτών (data-intensive computing). Οι προσεγγίσεις που ακολουθούνται για την διευθέτηση των συγκεκριμένων υπολογιστικών προκλήσεων περιλαμβάνουν την:

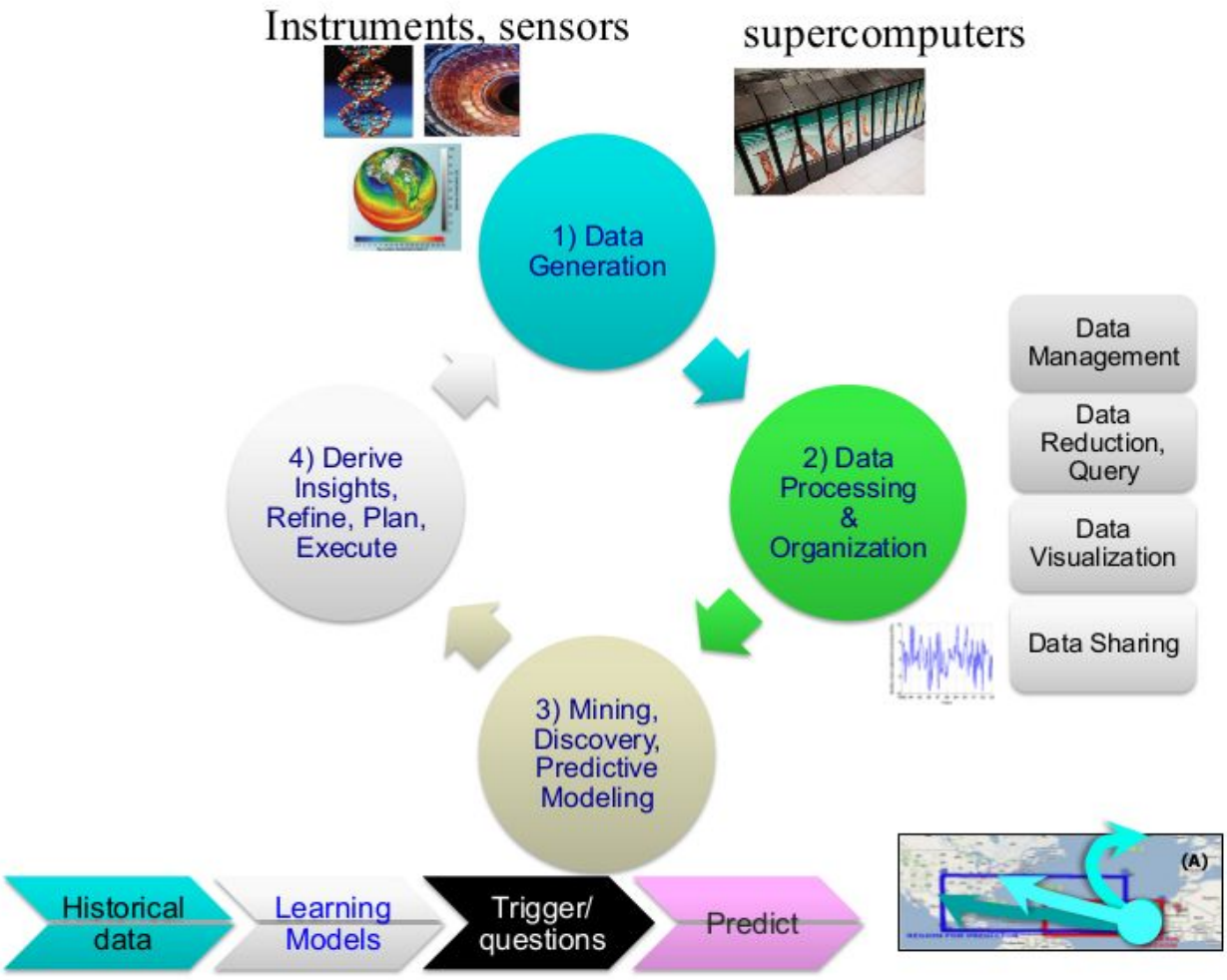
1. Όσο το δυνατόν ταχύτερη αποθήκευση των δεδομένων μιας διαδικασίας προσομοίωσης, στοχεύοντας στην μελλοντική επεξεργασία ή αρχειοθέτηση.
2. Ελαχιστοποίηση της μετακίνησης των δεδομένων.
3. Βελτιστοποίηση της επικοινωνίας μεταξύ υπολογιστικών και αποθηκευτικών κόμβων.
4. Αποτελεσματική, συντονισμένη σχεδίαση, χρήση και βελτιστοποίηση όλων των επιμέρους τμημάτων του υπολογιστικού συστήματος, από την αρχιτεκτονική ως το λογισμικό.

Ο αυξανόμενος ρυθμός παραγωγής του όγκου των δεδομένων που εξάγονται από επιστημονικές διατάξεις/όργανα όπως τα τηλεσκόπια, οι επιταχυντές, οι διατάξεις σύγκρουσης σωματιδίων, οι πηγές φωτός καθώς και ο πολλαπλασιασμός των αισθητήρων εισάγουν μια νέα μεθοδολογία για την επιστημονική ανακάλυψη [50]. Αυτή η τάση συνήθως αναφέρεται ως “Big Data”. Παρόλα αυτά, η απλή παραγωγή δεδομένων δεν μπορεί να έχει ιδιαίτερη αξία αν δεν οδηγεί στην γνώση και στην δημιουργία νέων ιδεών. Έτσι η τέταρτη μεθοδολογία με την οποία επιδιώκεται η εκμετάλλευση των πληροφοριών που μπορούν να εξαχθούν από τα τεράστια σύνολα δεδομένων, με σκοπό την δημιουργία νέας επιστημονικής γνώσης, έχει αναδειχθεί ως ένα απαραίτητο συμπλήρωμα των τριών υφισταμένων δομών.

Η πολυπλοκότητα και οι προκλήσεις της τέταρτης μεθοδολογίας προκύπτουν από την αυξανόμενη ταχύτητα, ετερογένεια καθώς και τον όγκο παραγωγής των δεδομένων. Για παράδειγμα, τα πειράματα που χρησιμοποιούν το μεγάλο ανδρονικό επιταχυντή Large Hadron Collider (LHC) παράγουν δεκάδες petabytes δεδομένων ανά έτος, οι παρατηρήσεις και οι προσομοιώσεις των δεδομένων στον τομέα των κλιματολογικών επιστημών αναμένεται να αγγίξουν το επίπεδο των exabytes μέχρι το 2021, ενώ τα πειράματα που βασίζονται σε πηγές φωτός αναμένεται να παράγουν εκατοντάδες terabytes ανά μέρα [51]. Η ετερογένεια των δεδομένων προσθέτει επιπλέον πολυπλοκότητα, καθώς τα δεδομένα μπορεί να είναι προσωρινά, διαθέσιμα για συγκεκριμένα διαστήματα, μη δομημένα, ή μπορεί να προέρχονται από πολύπλοκες μορφές.

Η ανάλυση του τεράστιου όγκου πολύπλοκων δεδομένων με σκοπό την εξαγωγή νέας γνώσης απαιτεί υπολογιστικά συστήματα που να ακολουθούν τις απαιτήσεις των δεδομένων. Οι υπολογισμοί καθώς και ο έλεγχος αυτών περιλαμβάνει πολύπλοκα ερωτήματα, ανάλυση, στατιστική μεθοδολογία, διατύπωση υποθέσεων και επικύρωση αυτών, καθώς και μεθόδους εξόρυξης δεδομένων. Ένας τυπικός κύκλος δημιουργίας νέας γνώσης (σχήμα 1 [45]) από πειραματικές διατάξεις παγκόσμιας κλίμακας αποτελείται από τα ακόλουθα στάδια:

1. Παραγωγή δεδομένων. Τα δεδομένα μπορεί να παράγονται από όργανα, πειράματα, ανιχνευτές ή υπερυπολογιστές.
2. Επεξεργασία δεδομένων και οργάνωσή τους. Σε αυτό το στάδιο επιχειρείται η αναδιοργάνωση, η μείωση του όγκου, η δημιουργία υποσυνόλων, η οπτικοποίηση, η διενέργεια ερωτημάτων με σκοπό την ανάλυση, διανομή καθώς και άλλες γενικές διαδικασίες επεξεργασίας δεδομένων. Αυτό μπορεί να περιλαμβάνει και την σύνθεση των δεδομένων με άλλα εξωτερικά δεδομένα ή δεδομένα με ιστορικό υπόβαθρο.
3. Ανάλυση δεδομένων, εξόρυξη και ανακάλυψη της γνώσης. Οι αλγόριθμοι επεξεργασίας και το αντίστοιχο λογισμικό ακολουθούν το μέγεθος και την πολυπλοκότητα των δεδομένων.
4. Ενέργειες,σχόλια/ανατροφοδότηση και βελτιστοποίηση.



Σχήμα 1: Τυπικός κύκλος δημιουργίας νέας γνώσης

1.2 Υφιστάμενες κατανεμημένες υπολογιστικές υποδομές, παγκόσμιας κλίμακας

Η ιδέα ενός υπολογιστικού πλέγματος αρχικά εμφανίστηκε στα μέσα της δεκαετίας του 1990 και προτάθηκε ως μία υποδομή για προωθημένη επιστημονική και τεχνολογική έρευνα. Στις αρχές τις δεκαετίας του 2000 οι τεράστιες ανάγκες σε υπολογιστική και

αποθηκευτική ισχύ που δημιουργήθηκαν από πειράματα όπως αυτό του Μεγάλου Επιταχυντή Αδρονίων (LHC) του Ευρωπαϊκού Κέντρου Πυρηνικών Ερευνών (CERN) της Γενεύης, έκαναν επιτακτική την ανάγκη δημιουργίας μιας υπολογιστικής υποδομής παγκόσμιας κλίμακας, όπως τα υπολογιστικά Πλέγματα (Grid Computing Infrastructures).

Παρά το γεγονός ότι τα περισσότερα ερευνητικά ιδρύματα διέθεταν κεντρικοποιημένες υπολογιστικές υποδομές, όπως συστοιχίες υπολογιστών ή υπερυπολογιστές, διαπιστώθηκε ότι υπήρχαν κάποιες χρονικές περίοδοι όπου οι ανάγκες σε υπολογιστική ισχύ ξεπερνούσαν τις μεμονωμένες δυνατότητες τους. Ήταν επίσης φανερό, ότι θα ήταν ασύμφορο να αποκτηθεί επιπλέον εξοπλισμός που θα χρησιμοποιούνταν μόνο για ένα μικρό χρονικό διάστημα. Έτσι δημιουργήθηκε η έννοια του διαμοιρασμού των πόρων μεταξύ μεγάλων υπολογιστικών υποδομών.

Στην προσπάθεια να αυτοματοποιηθούν οι διαδικασίες για τον διαμοιρασμό των πόρων και τον ορισμό των κανόνων διαχείρισης και χρήσης, έγινε αναγκαία η εισαγωγή της έννοιας του Ιδεατού Οργανισμού (Virtual Organization - VO) [1]. Τα μέλη ενός Ιδεατού Οργανισμού έχουν μια περισσότερο χαλαρή σχέση μεταξύ τους, καθώς μπορεί να είναι μέλη ενός επιστημονικού πειράματος, που υποστηρίζεται από πολλά πανεπιστημιακά ιδρύματα ή ερευνητικά κέντρα. Ο Ιδεατός Οργανισμός έχει πεπερασμένο χρόνο ζωής. Για τον λόγο αυτό χρειάστηκε να δομηθούν αυτοματοποιημένοι μηχανισμοί που να επιτρέπουν την συμμετοχή στον ιδεατό οργανισμό με δυναμικό τρόπο. Έπρεπε επίσης να οριστούν οι πολιτικές διαχείρισης των κοινόχρηστων πόρων καθώς και να εξασφαλιστεί ότι η αυθεντικοποίηση και η πιστοποίηση θα γίνεται σε ένα σημείο για ολόκληρη την υποδομή.

Γίνεται φανερό πως μία κατανεμημένη υπολογιστική υποδομή παγκόσμιας κλίμακας θα πρέπει να είναι ένα σύστημα [2] που θα :

1. συντονίζει πόρους που δεν θα υπόκεινται σε κεντρικοποιημένο έλεγχο.
2. χρησιμοποιεί πρότυπα, πρωτόκολλα και διεπαφές γενικού σκοπού και ανοιχτού κώδικα.
3. παρέχει υψηλού επιπέδου ποιότητα υπηρεσίας.

Η λειτουργία του υπολογιστικού αυτού πλέγματος είχε ως αποτέλεσμα ένα ευρύ φάσμα πλεονεκτημάτων, όπως :

1. πρόσβαση στις υποδομές με διαφανή τρόπο.
2. καλύτερη αξιοποίηση των πόρων.
3. πολύ μεγάλη υπολογιστική και αποθηκευτική ικανότητα.
4. ευελιξία, προσαρμοστικότητα και αυτοματισμό μέσω της δυναμικής και συντονισμένης διαλειτουργικότητας των πόρων [3].
5. μείωση του συνολικού κόστους, λόγω της χρήσης κοινών πρακτικών και προτύπων.
6. πρόσβαση σε υπερυπολογιστικούς πόρους για οργανισμούς αλλά και για άτομα, κάτι που θα ήταν αδύνατο να επιτύχουν με ίδια μέσα.

1.3 Κίνητρο και ερευνητικές προκλήσεις

Σε ένα τόσο δυναμικά εξελισσόμενο περιβάλλον όπου οι υπολογιστικοί, αποθηκευτικοί και δικτυακοί πόροι είναι εγγενώς ετερογενείς, διαμοιράζονται και μεταβάλλουν συνεχώς την κατάστασή τους, οι ερευνητικές προκλήσεις αναδύονται με ταχύτατους ρυθμούς.

Ταυτόχρονα, οι απαιτήσεις των εφαρμογών, όπως αυτές παρουσιάζονται από τις αναφορές των επιστημονικών ομάδων που δραστηριοποιούνται στους τομείς της κλιματολογικής αλλαγής της φυσικής υψηλών ενεργειών και άλλων τομέων, που θα υλοποιηθούν με χρονικό ορίζοντα το 2022 [51] κατατείνουν στο ότι οι υπάρχουσες υποδομές δεν θα μπορούν να ανταποκριθούν αν δεν ανασχεδιαστούν. Τεχνικά, αυτό συνεπάγεται ότι θα πρέπει τα σημερινά συστήματα Petascale να εξελιχθούν σε eXascale. Η ευρύτερη επιστημονική κοινότητα, όπως αυτή εκφράζεται από αναφορές επιστημονικών ομάδων και τεχνικές μελέτες κρατών, συγκλίνει στο ότι είναι αναγκαίο να τοποθετηθεί η διαλειτουργικότητα μεταξύ ετερογενών συστημάτων σε ένα ευρύτερο πλαίσιο που θα επιτρέπει να δομηθούν συνέργειες μεταξύ ευρύτερων κοινοτήτων.

Η πρακτική στα Petascale συστήματα υπαγορεύει την εγγραφή των μη επεξεργασμένων δεδομένων, που εξάγονται από τις προσομοιώσεις, σε αποθηκευτικά μέσα και δίσκους με μόνιμο χαρακτήρα, ενώ σε δεύτερο στάδιο πραγματοποιείται η ανάγνωση και η περαιτέρω ανάλυση τους. Αυτή η προσέγγιση είναι απίθανο να κλιμακώνει σε eXascale περιβάλλοντα, εξαιτίας των διαφορών ανάμεσα σε υπολογιστική ισχύ, τη μνήμη του μηχανήματος και του διαθέσιμου bandwidth (I/O και δίκτυο) [41]. Ως εκ τούτου, κατά την διάρκεια των προσομοιώσεων σε eXascale περιβάλλοντα δεν θα είναι εφικτό να εγγραφούν αρκετά δεδομένα στη μόνιμη αποθήκευση ώστε να εξασφαλιστεί μια αξιόπιστη ανάλυση. Αναμένεται ότι μεγάλο μέρος της τρέχουσας ανάλυσης και οπτικοποίησης θα πρέπει να ενσωματωθούν με την προσομοίωση σε μια ολοκληρωμένη συνολική ροή εργασίας [45].

Πιο συγκεκριμένα, τα συνεργατικά πειράματα παγκόσμιας κλίμακας παράγουν δεδομένα που συνεχώς αυξάνονται σε μέγεθος αλλά και πολυπλοκότητα, με αποτέλεσμα να καθιστούν την ανάλυση, αρχειοθέτηση αλλά και το διαμοιρασμό τους ως μία από τις μεγαλύτερες προκλήσεις του 21ου αιώνα. Αυτού του τύπου τα πειράματα, στην πλειοψηφία τους, υιοθετούν υπολογιστικά μοντέλα που αποτελούνται από βαθμίδες/ιεραρχίες (Tiers), όπου κάθε βαθμίδα παρέχει πρόσβαση σε πολλαπλά υπολογιστικά κέντρα και υποστηρίζει μια συγκεκριμένη ομάδα υπηρεσιών.

Για τα διαφορετικά βήματα επεξεργασίας των δεδομένων χρησιμοποιούνται πολλαπλά πακέτα λογισμικού. Οι υπολογιστικές απαιτήσεις αυτών των πειραμάτων είναι εξαιρετικά υψηλές και περιλαμβάνουν την χρήση πολλαπλών υπολογιστικών τεχνικών και υποδομών. Ταυτόχρονα, η συνεργατική τους φύση, συνεργασία μεταξύ κρατών, ερευνητικών ιδρυμάτων και ομάδων, απαιτεί πολύ συχνές μεταφορές δεδομένων σε δίκτυα ευρείας ζώνης (WAN) και των διαμοιρασμό των δεδομένων μεταξύ ατόμων και ομάδων. Συνήθως ένα τέτοιο υπολογιστικό μοντέλο χρησιμοποιεί πολλαπλές υπολογιστικές υποδομές όπως Grids, Clouds, HPCs, Data Centers και τοπικές συστοιχίες υπολογιστών.

Ο κύκλος εργασίας ενός τέτοιου πειράματος είναι εξαιρετικά πολύπλοκος και υπαγορεύει την μετακίνηση πολλαπλών ομάδων δεδομένων από κάποιο Data Center στην τοπική συστοιχία υπολογιστών ώστε να διεξαχθεί μια περιορισμένη ανάλυση τοπικά πριν μεταφερθεί η διεργασία με τα δεδομένα που την αποτελούν σε κάποια υπολογιστική υποδομή μεγάλης κλίμακας (Grid, HPC). Τελικά, θα πρέπει να ανακτηθούν τα δεδομένα εξόδου αλλά και να τα αντιγραφούν σε κάποιο Data Center, κάνοντας χρήση πολλαπλών διεπαφών και εργαλείων.

Ο συγκεκριμένος κύκλος εργασίας υπαγορεύεται από πολλούς και διαφορετικούς λόγους καθώς οι υπολογιστικοί πόροι που χρησιμοποιεί ένα τέτοιο πείραμα μπορεί να εφαρμόζουν αντιδιαμετρικά αντίθετες τεχνικές υλοποιήσεις, να ανήκουν σε διαφορετικές ομάδες καθώς και να προσπαθούν να επιτύχουν πολλαπλούς διαφορετικούς στόχους.

Είναι φανερό πως, τα υπάρχοντα εργαλεία και υποδομές δεν μπορούν να παρέχουν την ευελιξία που απαιτούν τα συνεργατικά πειράματα επόμενης γενιάς, με

χαρακτηριστικό παράδειγμα, τις υποδομές Πλέγματος (Grids) που δομήθηκαν την προηγούμενη δεκαετία και οι οποίες εισήγαγαν την λογική της πρόσβασης σε πόρους με την μορφή της υπηρεσίας.

Μεγαλύτερη κριτική προς τα Grids αποτελεί το γεγονός ότι οι υπάρχουσες υλοποιήσεις της αρχιτεκτονικής μετατρέπουν τα υπολογιστικά Πλέγματα σε κεντρικοποιημένες υποδομές, κάτι που παραβιάζει το μοντέλο λειτουργίας τους. Οι συγκεκριμένες υλοποιήσεις των υποδομών Πλέγματος παρέχουν απο-κεντρικοποιημένη διαχείριση των αποθηκευτικών και υπολογιστικών πόρων, αλλά στο επίπεδο των εφαρμογών οι περισσότεροι ιδεατοί οργανισμοί (VO) χειρίζονται τα δεδομένα μόνο σε πολύ συγκεκριμένες υποδομές ανά βαθμίδα. Με αποτέλεσμα να εφαρμόζουν μηχανισμούς κεντρικού ελέγχου, απεμπολώντας σε μεγάλο βαθμό τα πλεονεκτήματα μίας αρχιτεκτονικής Πλέγματος.

Αντίστοιχα, οι τεχνολογίες Data as a Service (DaaS) παρουσιάζουν άλλου είδους μειονεκτήματα καθώς δεν επιτρέπουν στους χρήστες να παρεμβαίνουν και να διαμορφώνουν τα μοντέλα δεδομένων, έτσι ώστε να “εμπλουτίζονται” με νέες λειτουργίες [78]. Οι πάροχοι αυτών των υπηρεσιών διαθέτουν μια πολύ περιορισμένη ομάδα λειτουργιών που βασίζεται στις εντολές Create, Read, Update, Delete (CRUD) ή απλά επιτρέπουν την μεταφορά των δεδομένων στην πλευρά του χρήστη/πελάτη χωρίς να υπάρχει η δυνατότητα της επιτόπου περαιτέρω επεξεργασίας. Έτσι η συνεργασία μεταξύ των ομάδων περιορίζεται σε μεγάλο βαθμό ή δεν είναι τόσο αποδοτική όσο θα έπρεπε, καθώς στο επίπεδο του χρήστη όλες οι παραπάνω υποδομές αποτελούν απομονωμένες υπολογιστικές και αποθηκευτικές “νησίδες” και όχι μια ενιαία υποδομή.

Το πρόβλημα διαχείρισης και μεταφοράς των δεδομένων γίνεται πιο επιτακτικό αν συνυπολογιστεί η απαιτούμενη κλιμάκωση των υποδομών στα μελλοντικά eXascale περιβάλλοντα. Η διαχείριση των δεδομένων και η μεταφορά τους μεταξύ υπολογιστικών και αποθηκευτικών πόρων δημιουργεί τεράστιους περιορισμούς στις εφαρμογές μεγάλης κλίμακας. Αυτό οφείλεται στο ότι τα αποθηκευτικά συστήματα παρουσιάζουν τεράστια αναντιστοιχία μεταξύ της χωρητικότητας και του διαθέσιμου εύρους ζώνης (I/O και δίκτυο). Τα προβλήματα για την ανάπτυξη των eXascale συστημάτων είναι τεράστια από οικονομικής, περιβαλλοντολογικής αλλά και τεχνικής άποψης. Προβλήματα όπως η θέση και η ψύξη των συγκεκριμένων υποδομών θα έχουν πρωτεύοντα ρόλο για την υλοποίησή τους. Έτσι, θα πρέπει να αντιμετωπιστούν τεράστιες ερευνητικές προκλήσεις καθώς είναι απαραίτητο να αυξηθούν αισθητά οι διαθέσιμοι κόμβοι αποθήκευσης, για να καλυφθούν οι απαιτήσεις σε διαθέσιμο εύρος ζώνης (I/O και δίκτυο). Κάτι που όμως θα εκτινάξει την κατανάλωση ενέργειας σε απαγορευτικά επίπεδα [41].

Είναι πλέον φανερό ότι οι επιστημονικοί υπολογισμοί μεγάλης κλίμακας είναι εξαιρετικά απαιτητικοί με αποτέλεσμα να έχουν τεράστιες ανάγκες σε υπολογιστική ισχύ. Οι παράλληλοι υπολογισμοί και τα παράλληλα συστήματα αρχείων αναγνωρίζονται ως η μόνη εφικτή λύση καθώς οι διεργασίες εισόδου/εξόδου (I/O) αποτελούν το σημαντικότερο σημείο συμφόρησης στην απόδοση των εφαρμογών. Τα προβλήματα εντείνονται καθώς οι επιδόσεις των επεξεργαστών αυξάνονται θεαματικά ενώ το λογισμικό και το υλικό που δομεί τις αποθηκευτικές διατάξεις δεν εξελίσσεται με αντίστοιχους ρυθμούς.

Οι σημαντικότεροι παράγοντες που επηρεάζουν την απόδοση είναι ο αριθμός των παράλληλων διεργασιών που συμμετέχουν στις μεταφορές των δεδομένων, το μέγεθος της κάθε μεταφοράς καθώς και τα διάφορα I/O μοτίβα πρόσβασης. Τα μοτίβα πρόσβασης έχουν ως ακολούθως:

1.Υποχρεωτικά (Compulsory), αποτελείται από διεργασίες I/O που πρέπει να διεξαχθούν έτσι ώστε να διαβαστεί η αρχική κατάσταση του προγράμματος από τον σκληρό δίσκο καθώς και να εγγραφεί η τελική του κατάσταση στο τέλος της εκτέλεσης του.

2.Checkpoint/restart, χρησιμοποιείται ώστε να διατηρηθεί η κατάσταση του υπολογισμού σε περίπτωση αστοχίας του υλικού ή του λογισμικού με αποτέλεσμα να απαιτείται η επανεκκίνηση του.

3.Στιγμιότυπα της υπολογιστικής προόδου.

4.Αναγνώσεις/εγγραφές που υπερβαίνουν την διαθέσιμη RAM.

5.Συνεχής έξοδος δεδομένων οπτικοποίησης και άλλων λειτουργιών μετα-επεξεργασίας.

Στις εφαρμογές που εξετάζονται στην παρούσα εργασία οι διεργασίες εγγραφής είναι περισσότερο συχνές από τις αντίστοιχες της ανάγνωσης και οι κατηγορίες 2 και 4 κυριαρχούν. Οι συγκεκριμένες εφαρμογές χρησιμοποιούνται σε πειράματα Σωματιδιακής Φυσικής και Αστροφυσικής αλλά και σε τομείς όπως η ρευστοδυναμική και η βιοπληροφορική.

Ένας επιπλέον σημαντικός παράγοντας που επηρεάζει την απόδοση είναι η γενικότερη αρχιτεκτονική της αποθηκευτικής υποδομής. Μια τυπική High Performance Computing (HPC) υποδομή χρησιμοποιεί ένα μικρό μέρος των διαθέσιμων κόμβων για αποθηκευτικούς σκοπούς (κόμβοι I/O). Κάθε ένας από τους κόμβους I/O διαθέτει έναν μεγάλο αριθμό σκληρών δίσκων και εφαρμόζει ένα Redundant Array of Independent Disks (RAID) σχηματισμό για την οργάνωση των αποθηκευτικών μέσων. Τα διαμοιραζόμενα συστήματα αρχείων έχουν σημαντικούς περιορισμούς όταν εφαρμόζονται σε μεγάλης κλίμακας συστήματα, επειδή:

1. Το εύρος ζώνης (I/O και δίκτυο) δεν κλιμακώνει οικονομικά σε αντιστοιχία με την διαθέσιμη χωρητικότητα που παρέχουν οι σκληροί δίσκοι.

2. Η I/O κίνηση στην δικτυακή υποδομή και στους αποθηκευτικούς κόμβους μπορεί να επηρεαστεί από άλλες διεργασίες ή να με την σειρά της να επηρεάσει άλλες διεργασίες.

1.4 Συνεισφορά διατριβής

Στη παρούσα διατριβή αναπτύχθηκε το πλαίσιο ΙΚΑΡΟΣ στοχεύοντας στην διευθέτηση των ερευνητικών προκλήσεων της διαχείρισης και μεταφοράς των δεδομένων από τα συνεργατικά επιστημονικά πειράματα επόμενης γενιάς. Το πλαίσιο ΙΚΑΡΟΣ δομήθηκε στοχεύοντας στην επίτευξη των πιο κάτω χαρακτηριστικών/λειτουργιών :

1. <u>Καλύτερο συντονισμό μεταξύ των πολλαπλών στρωμάτων λογισμικού στην συνολική ροή των δεδομένων (τοπική-απομακρυσμένη πρόσβαση).</u>

Το ΙΚΑΡΟΣ έχει δομηθεί ως ένα “λεπτό” στρώμα (thin layer) που έχει την δυνατότητα να προσφέρει υπηρεσίες σε πολλαπλά επίπεδα και επιτρέπει την απομόνωση των λειτουργιών I/O μίας διεργασίας από τις αντίστοιχες των άλλων διεργασιών, στοχεύοντας στην μέγιστη αξιοποίηση των διαθέσιμων πόρων. Σε τεχνικό επίπεδο επιτρέπει την απευθείας πρόσβαση σε κάθε αποθηκευτικό κόμβο (I/O κόμβο) από οποιαδήποτε επίπεδο και αν ενεργεί. Με αυτόν τον τρόπο επιτυγχάνεται η ροή των

δεδομένων με χρήση παράλληλων τεχνικών στην συνολική διαδρομή χωρίς να απαιτούνται ενδιάμεσα στάδια συγχρονισμού. Ταυτόχρονα επιτυγχάνεται καλύτερη συνεργασία μεταξύ υλικού και λογισμικού κάτι που είναι απαραίτητο για την κλιμάκωση σε eXascale περιβάλλοντα [79].

2. *Δυνατότητα δημιουργίας συνεργιών μεταξύ ευρύτερων κοινοτήτων.*

Το ΙΚΑΡΟΣ προσπαθεί να αναγνωρίσει τις πιθανές ομοιότητες μεταξύ των λειτουργιών στα διαφορετικά επίπεδα. Σκοπός είναι να διατηρηθεί η λογική της διαλειτουργικότητας μεταξύ των ετερογενών υποδομών, κάτι που ουσιαστικά παρέχεται από την αυτονομία δράσης των υπηρεσιών μεταξύ των διαφορετικών επιπέδων, και να επεκταθεί σε συνέργεια μεταξύ κοινοτήτων. Γίνεται προσπάθεια, σε αυτόν τον κατακερματισμό, να αναγνωριστούν κοινές λειτουργίες και συμπεριφορές που θα οδηγήσουν στην δημιουργία υποδομών που θα διατρέχονται από κοινές πρακτικές και θα μπορούν να υλοποιηθούν με μειωμένο συνολικό κόστος.

Το πρωτόκολλο HTTP έχει επιλεγεί ως ο βασικός μηχανισμό πάνω στον οποίο δομήθηκε το όλο πλαίσιο. Η επιλογή του HTTP επιτρέπει την δημιουργία συνεργιών μεταξύ ευρύτερων κοινοτήτων, στο επίπεδο του λογισμικού, ώστε να λειτουργεί με ομοιόμορφο τρόπο σε όλα τα επίπεδα. Ενώ ταυτόχρονα διατηρείται η αυτονομία υλοποίησης των υπηρεσιών των διαφόρων επιπέδων. Η επιλογή των τεχνοοικονομικών παραμέτρων, σε αυτήν την περίπτωση η επιλογή του HTTP, διαδραματίζει κυρίαρχο ρόλο στο κατά πόσο η υποδομή επιτρέπει την δημιουργία ευρύτερων συνεργιών ή όχι.

3. *Κλιμάκωση του διαθέσιμου εύρους ζώνης (I/O και δίκτυο) με κόστος ανάλογο με αυτό της κλιμάκωσης της χωρητικότητας των αποθηκευτικών συστημάτων. Δημιουργία υποδομών που θα απαιτούν εξαιρετικά μικρότερα ποσά ηλεκτρικής ενέργειας.*

Το ΙΚΑΡΟΣ επιτρέπει την χρήση αποθηκευτικών συσκευών, χαμηλών τεχνικών προδιαγραφών και χαμηλής κατανάλωσης ενέργειας, για την δημιουργία υψηλής απόδοσης αποθηκευτικών σχηματισμών on demand. Οι αναφερόμενες συσκευές χαμηλών τεχνικών προδιαγραφών (πίνακας 2) είναι συσκευές που απευθύνονται σε χρήσεις οικιακές ή γραφείου, γνωστές ως Small Office/Home Office Network Attach Storage (SOHO-NAS). Τα υφιστάμενα συστήματα αρχείου ακόμα και όταν αναφέρουν ότι υποστηρίζουν την χρήση κοινών εμπορικών προϊόντων, δεν αναφέρονται σε τέτοιου είδους συσκευές, αλλά σε συστήματα μέτριας και υψηλής απόδοσης που είναι γενικότερα διαθέσιμα στην αγορά (commodity).

Η χρήση ενός μεγάλου αριθμού SOHO-NAS συσκευών για την δημιουργία ενός αποθηκευτικού συστήματος υψηλής απόδοσης μπορεί να επιφέρει τα επιθυμητά αποτελέσματα. Οι συγκεκριμένες συσκευές έχουν πολύ μικρό κόστος κτήσης, πολύ χαμηλή κατανάλωση ενέργειας, θεωρούνται plug and play και η χρήση τους μπορεί να αυξήσει με οικονομικό τρόπο το συνολικό διαθέσιμο εύρος ζώνης (I/O και δίκτυο).

Για παράδειγμα, η δημιουργία ενός αποθηκευτικού συστήματος με την χρήση συσκευών τύπου SOHO-NAS θα κοστίσει κατά προσέγγιση το 1/5 από ένα τυπικό αποθηκευτικό σύστημα ενώ η κατανάλωση ενέργειας δεν θα υπερβαίνει το 1/3. Με αυτόν τον τρόπο γίνεται, καταρχήν, εφικτό το να δημιουργηθούν υποδομές που θα αποτελούνται από έναν αριθμό κόμβων στην κλίμακα του εκατομμυρίου. Αφού η κατανάλωση σε ηλεκτρική ενέργεια για το συνολικό σύστημα μειώνεται σε σημαντικό βαθμό και το διαθέσιμο εύρος ζώνης μπορεί να είναι ικανοποιητικό.

4. *Αντιμετώπιση των προβλημάτων κλιμάκωσης των μηχανισμών μεταδεδομένων.*

Οι μηχανισμοί μεταδεδομένων του ΙΚΑΡΟΣ έχουν κυρίαρχο ρόλο στην όλη αρχιτεκτονική. Οι συγκεκριμένοι μηχανισμοί επιτρέπουν να δημιουργηθούν κοινές μεθοδολογίες αντιμετώπισής των προβλημάτων, στην γενικότερη ροή των δεδομένων (τοπική-απομακρυσμένη πρόσβαση), διασφαλίζοντας παράλληλα την αυτονομία υλοποίησης των επιμέρους υπηρεσιών. Το ΙΚΑΡΟΣ παρέχει μία οντότητα μεταδεδομένων για την συνολική ροή. Η οντότητα αυτή ενεργεί ως utility και είναι δομημένη με ιεραρχική δομή που αποκρίνεται με διαφορετικό τρόπο στα αιτήματα των χρηστών ή των εφαρμογών.

Το ΙΚΑΡΟΣ στοχεύει στο να λειτουργεί ως ένα πλαίσιο που θα επιτρέπει την δημιουργία, στο επίπεδο του χρήστη, ενός ενιαίου αποθηκευτικού σχηματισμού που θα μπορεί να αποτελείται από όλες τις διαφορετικές υποδομές (Grids, Clouds, HPCs, Data Centers και τοπικές συστοιχίες υπολογιστών) που χρησιμοποιεί το εκάστοτε υπολογιστικό μοντέλο. Έτσι θα υπάρχει η δυνατότητα αντιμετώπισης των ζητημάτων της κλιμάκωσης, της απόδοσης και της κατανάλωσης ενέργειας.

Ο στόχος αυτός μπορεί να επιτευχθεί δομώντας κατάλληλους τεχνοοικονομικούς μηχανισμούς που θα επιτρέπουν την συνεργασία μεταξύ ευρύτερων κοινοτήτων. Αυτή η λογική επιτρέπει τη δημιουργία υποδομών στις οποίες οι χρήστες θα έχουν μεγαλύτερη επιρροή στην διακυβέρνηση τους κάτι που θα επιστρέψει να τοποθετηθεί η επιστήμη των υπολογιστών και η εκμετάλλευσή των “Big Data” στο κέντρο της επιστημονικής ανακάλυψης.

1.5 Οργάνωση διατριβής

Η διατριβή οργανώνεται ως ακολούθως:

- Κεφάλαιο 2, εξετάζονται οι προκλήσεις για την ανάπτυξη των eXascale υποδομών.
- Κεφάλαιο 3, παρουσιάζεται το πλαίσιο ΙΚΑΡΟΣ και η αρχιτεκτονική του.
- Κεφάλαιο 4, παρουσιάζεται η οντότητα μεταδεδομένων του ΙΚΑΡΟΣ καθώς και η λογική που αυτή εισάγει στην συνολική ροή των δεδομένων (τοπική-απομακρυσμένη πρόσβαση).
- Κεφάλαιο 5, παρουσιάζονται οι μηχανισμοί εισόδου/εξόδου (I/O) οι τεχνικές που επιτρέπουν στο ΙΚΑΡΟΣ να υπερέχει (κυρίως κατά την εκτέλεση των διεργασιών εγγραφής) και διενεργούνται δύο ομάδες μετρήσεων. Στην πρώτη ομάδα μετρήσεων εκτελούνται δοκιμές απόδοσης και φόρτου σε πραγματικό περιβάλλον παραγωγής χρησιμοποιώντας δεδομένα του πειράματος “Κυβικού Χιλιομέτρου Τηλεσκοπίου Νετρίνων” (KM3NeT) και οι εφαρμογές SeaTray [52] και ROOT [53]. Στην δεύτερη ομάδα μετρήσεων χρησιμοποιείται το benchmark tool IOR-HPC που διενεργεί αιτήματα τυχαίας προσπέλασης σε παράλληλα προγραμματιστικά περιβάλλοντα, όπως το MPICH.
- Κεφάλαιο 6, παρουσιάζονται οι μηχανισμοί μεταφοράς δεδομένων σε δίκτυα ευρείας περιοχής καθώς και η ενσωμάτωση τεχνικών παράλληλων καναλιών μεταφοράς στο HTTP. Επιπλέον, διενεργούνται μετρήσεις σε πραγματικό περιβάλλον παραγωγής στα πλαίσια του πειράματος CMS-LHC του CERN. Αυτή η ομάδα των

μετρήσεων καταδεικνύει την υπεροχή του ΙΚΑΡΟΣ στην συνολική ροή των δεδομένων (τοπική-απομακρυσμένη πρόσβαση).

- Κεφάλαιο 7, παρουσιάζεται το ΙΚΑΡΟΣ ως ένα πλαίσιο δημιουργίας on demand αποθηκευτικών σχηματισμών που επιτρέπει την απομόνωσή των λειτουργιών I/O μίας διεργασίας από τις αντίστοιχες των άλλων διεργασιών, στοχεύοντας στην μέγιστη αξιοποίηση των διαθέσιμων πόρων.

- Κεφάλαιο 8, αναπτύσσονται οι πιθανές συνέργειες των δικτύων δεδομένων με τηλεπικοινωνιακά δίκτυα, υπό το πρίσμα του ΙΚΑΡΟΣ. Αυτή η προοπτική δυνητικά θα μπορούσε να ενισχύσει τις προσπάθειες υλοποίησης των eXascale αποθηκευτικών υποδομών.

- Κεφάλαιο 9, παρουσιάζονται τα συμπεράσματα καθώς και το πλαίσιο της μελλοντικής εργασίας.

2. ΠΡΟΚΛΗΣΕΙΣ ΚΑΙ ΕΥΚΑΙΡΙΕΣ ΣΕ EXASCALE ΠΕΡΙΒΑΛΛΟΝΤΑ

Τα τελευταία χρόνια, οι ερευνητικές δραστηριότητες στον τομέα των υπολογιστικών υποδομών υψηλής απόδοσης έχουν επικεντρωθεί στις προκλήσεις που συνδέονται με την δημιουργία eXascale υπολογιστικών συστημάτων. Τα εν λόγω συστήματα θα είναι 10^3 φόρες ταχύτερα από τα σημερινά, με δυνατότητα 10^{18} πράξεων ανά δευτερόλεπτο. Συστήματα με την προαναφερόμενη υπολογιστική ισχύ αναμένεται να διαδραματίσουν καθοριστικό ρόλο σε ένα ευρύ φάσμα επιστημονικών εφαρμογών, επιχειρηματικών δραστηριοτήτων καθώς και σε θέματα εθνικής ασφάλειας. Η πορεία για την δημιουργία eXascale υπολογιστικών συστημάτων απαιτεί σημαντικές προόδους σε ένα ευρύ φάσμα τεχνολογιών. Θεωρείται βέβαιο πως για την επίτευξη του συγκεκριμένου στόχου θα πρέπει να αναδιοργανώθει ριζικά ο τρόπος με τον οποίο δομούνται και χρησιμοποιούνται οι υπολογιστές [45].

Η πρόοδος προς την κατεύθυνση των eXascale συστημάτων απαιτεί συνεργατικές ενέργειες προς το ευρύτερο φάσμα του οικοσυστήματος των υπολογιστών. Ο τομέας των υπολογιστικών συστημάτων υψηλής απόδοσης αποτελεί μικρό κομμάτι της εν γένει επιχειρηματικότητας και θα πρέπει να μοχλεύσει τεχνολογίες που αναπτύσσονται για μεγαλύτερα τμήματα της αγοράς. Στην συνέχεια παρουσιάζονται οι σημαντικότερες προκλήσεις που έχει να αντιμετωπίσει η eXascale κοινότητα, και οι οποίες είναι δυνατόν να ευθυγραμμιστούν με την ευρύτερη αγορά. Πιθανότερη κοινότητα για την ανάπτυξη συνεργατικών δράσεων αποτελεί αυτή των εμπορικών εφαρμογών που επικεντρώνονται στα δεδομένα καθώς και στην οικονομική δραστηριότητα που σχετίζεται με το Διαδίκτυο (internet economy).

2.1 Προκλήσεις στην κατανάλωση ενέργειας

Πιθανότατα, το μεγαλύτερο εμπόδιο για την κατασκευή και λειτουργία ενός eXascale υπολογιστή είναι οι απαιτήσεις σε ηλεκτρική ενέργεια. Υπολογίζεται πως ένα τέτοιο μηχάνημα θα καταναλώνει αρκετές εκατοντάδες megawatts ανά έτος [46]. Η κατηγοριοποίηση των λειτουργιών με βάση την κατανάλωση μπορεί να οδηγήσει σε κάποια συμπεράσματα. Για παράδειγμα, υπολογίζεται ότι η κατανάλωση ενέργειας κατά την μεταφορά των δεδομένων είναι σημαντικά αυξημένη σε σχέση με τα αντίστοιχα ποσά ενέργειας που απαιτούνται για την εκτέλεση υπολογισμών. Πολλές αναδυόμενες τεχνολογίες έχουν την δυνατότητα να βοηθήσουν προς την κατεύθυνση της ανάπτυξης eXascale υποδομών και παρέχουν πολλές δυνατότητες διακοινοτικών συνεργασιών.

Ένα παράδειγμα αποτελεί η τεχνολογία through-silicon vias (TSV) που αναπτύσσεται στις μνήμες και επιτρέπει τη διατήρηση περισσότερων δεδομένων πλησιέστερα στους επεξεργαστές, μειώνοντας έτσι την συνολική μετακίνηση των δεδομένων. Η κοινοπραξία Hybrid Memory Cube (HMC) αναπτύσσει μνήμες οι οποίες θα διαθέτουν επεξεργαστικές ιδιότητες. Σκοπός αυτής της τεχνολογίας είναι να διενεργούνται ορισμένες επεξεργαστικές διαδικασίες πολύ κοντά στην μνήμη ώστε να αποφευχθεί η μεταφορά των δεδομένων στην CPU. Αυτές οι τεχνικές μπορούν να μειώσουν σημαντικά την κατανάλωση ενέργειας και ταυτόχρονα να βελτιώσουν σε σημαντικό βαθμό την απόδοση. Η κοινοπραξία HMC αναφέρει ότι μπορεί να βελτιώσει την απόδοση κατά 15% και να μειώσει την κατανάλωση ενέργειας κατά 70%, σε σχέση με την τεχνολογία DDR3 [45].

2.2 Ιεραρχίες αποθήκευσης και διαχείρισης δεδομένων

Μία τυπική επιστημονική εφαρμογή υπολογιστικής έντασης (CPU-intensive) περιλαμβάνει έναν σχετικά μικρό όγκο δεδομένων εισόδου και ένα, πιθανά, αρκετά μεγάλο όγκο δεδομένων εξόδου. Αντίθετα, οι εφαρμογές που επικεντρώνονται στα

δεδομένα έχουν ακριβώς την αντίθετη συμπεριφορά. Μεγάλες ποσότητες δεδομένων διαβάζονται με σκοπό την περαιτέρω ανάλυση και την εξαγωγή ενός μικρού υποσυνόλου ως έξοδο.

Μια σημαντική αναδυόμενη τεχνολογία στον τομέα της αποθήκευσης είναι οι μνήμες στερεάς κατάστασης [51]. Η NAND Flash χρησιμοποιείται ως εναλλακτική λύση παρέχοντας υψηλότερη απόδοση από τους σκληρούς δίσκους. Υποστηρίζει λειτουργίες τυχαίας προσπέλασης με γρήγορη απόκριση. Ωστόσο, παραμένει πιο ακριβή από τους δίσκους σε ανά-bit σύγκριση και έτσι φαίνεται πιο χρήσιμη ως ενδιάμεσο επίπεδο στην ιεραρχία αποθήκευσης. Ένα σημαντικό μειονέκτημα των σημερινών NAND Flash τεχνολογιών είναι η περιορισμένη αντοχή τους. Μια πολλά υποσχόμενη εναλλακτική λύση είναι οι μνήμες αλλαγής φάσης η οποίες αναμένεται να παρέχουν μεγαλύτερη αντοχή. Η εμφάνιση των τεχνολογιών solid-state για αποθήκευση και η εκμετάλλευσή τους από επεξεργαστές χαμηλής κατανάλωσης ενέργειας όπως οι ARM επιτρέπει τη κατασκευή νέων ιεραρχιών και μηχανισμών αποθήκευσης.

2.3 Ανταλλαγή δεδομένων και διαχείριση του κύκλου ζωής

Οι οικονομικές και τεχνικές προκλήσεις που δημιουργούν οι data-intensive εφαρμογές στα συστήματα αποθήκευσης και πρόσβασης δεδομένων μπορούν ακόμα και να ξεπεράσουν τις προκλήσεις που πρέπει να αντιμετωπίσουν τα αντίστοιχα υπολογιστικά συστήματα. Το LHC στο CERN έχοντας έναν ρυθμό παραγωγής δεδομένων που προσεγγίζει το petabyte ανά δευτερόλεπτο είναι ένα αντιπροσωπευτικό παράδειγμα. Όλα αυτά τα δεδομένα εκτός από το 0,001% πρέπει να εξεταστούν σε πραγματικό χρόνο και να απορριφθούν ή να προκύψουν δεδομένα κατάλληλα για προσομοιώσεις. Διαδικασία που επαναλαμβάνεται καθ όλη την διάρκεια των ετών που πραγματοποιείται η ανάλυση.

Η διαδικασία της διατήρησης των δεδομένων απαιτεί δυο βασικές ενέργειες/αποφάσεις, πρέπει να αποφασιστεί ποια δεδομένα θα διατηρηθούν (data retention) και σε ποια μορφή (data preservation). Τα δεδομένα μπορούν να κατηγοριοποιηθούν [45] ως εξής:

- Μοναδικά δεδομένα. Δεδομένα που εξάγονται από παρατηρήσεις κοσμικών φαινομένων και συστημάτων παρατήρησης της γης όπως δεδομένα από αισθητήρες που παρακολουθούν το κλίμα ή κάποιο σουπερνόβα. Τέτοιου είδους δεδομένα δεν μπορούν να αναπαραχθούν και ως εκ τούτου θα πρέπει να διατηρούνται επ 'αόριστον.
- Δύσκολο να αναπαραχθούν. Τα δεδομένα από πειράματα σωματιδιακής φυσικής είναι κατά αρχήν δυνατό να αναπαραχθούν, με στατιστικές μεθόδους. Όπως όμως συμβαίνει με τις εξερευνήσεις στο φεγγάρι, που πραγματοποιήθηκαν την δεκαετία του 1970, δεν αναμένεται να αναπαραχθούν στο εγγύς μέλλον. Περιστασιακά, παλιότερα δεδομένα σωματιδιακής φυσικής χάνουν την αξία τους καθώς τα πειράματα επαναλαμβάνονται, “εργαστηριακά” με καλύτερους ανιχνευτές. Σε αντίθεση με την προηγούμενη κατάσταση, τα δεδομένα που είναι δύσκολο να αναπαραχθούν θα πρέπει να διατηρηθούν για το ορατό μέλλον.
- Δεδομένα με δυνατότητα αναπαραγωγής. Τα δεδομένα που εξάγονται από πηγές φωτός και νετρονίων είναι αρκετά εύκολο να αναπαραχθούν. Έτσι, οποιαδήποτε χρονική στιγμή είναι δυνατόν να εκτιμηθεί το κόστος για την αναπαραγωγή σε σχέση με το αντίστοιχο κόστος για την διατήρηση, για μελλοντική χρήση. Για πολλές δεκαετίες, η μείωση του κόστους ανά μονάδα αποθήκευσης δίνει την δυνατότητα διατήρησης ακόμα και αυτής της κατηγορίας των δεδομένων ακόμα και επ 'αόριστον.

- Τα δεδομένα που έχουν εξαχθεί από προσομοιώσεις ή αποτελούν τα παράγωγα τους. Τα δεδομένα που είναι αποτέλεσμα της προσομοίωσης ή προέρχονται από πειραματικά ή παρατηρησιακά δεδομένα μπορούν εύκολα να ανασυνταχθούν. Τις περισσότερες φορές είναι δύσκολο να αποφασιστεί αν είναι απαραίτητο να διατηρηθούν η όχι. Έτσι το πιο αποδοτικό είναι να μεταφερθεί αυτή η δικαιοδοσία απευθείας στους επιστήμονες που τα παράγουν, οι οποίοι με την σειρά τους και θα μπορούν κατά περίπτωση να κρίνουν και να αποφασίσουν.

Η διατήρηση των δεδομένων που δεν μπορούν να αναπαραχθούν είτε διότι είναι φυσικά αδύνατο είτε οικονομικά ανέφικτο δημιουργεί νέες προκλήσεις. Οι σημειώσεις του Gallileo είναι εύκολο να κατανοηθούν 400 χρόνια μετά τις πειραματικές μετρήσεις. Τα ψηφιακά δεδομένα (bits) που προκύπτουν από σύγχρονα συστήματα συλλογής δεδομένων δεν έχουν προφανή ερμηνεία και συνήθως απαιτείται η συνδυασμένη γνώση πολλών επιστημόνων για την εξαγωγή συμπερασμάτων [41]. Η διατήρηση των δεδομένων απαιτεί την ταυτόχρονη διατήρηση δεδομένων, μεταδεδομένων, της προέλευσης τους, των αλγορίθμων καθώς και των λογισμικών με την βοήθεια των οποίων εξάγονται η πληροφορίες που οδηγούν στην γνώση.

2.4 Μεταδεδομένα

Τα μεταδεδομένα είναι είτε περιγραφικά είτε ακολουθούν μια συγκεκριμένη δομή, σε αυτήν την εργασία γίνεται χρήση λειτουργικών μεταδεδομένων. Αντίστοιχα οι κατάλογοι δεδομένων επιτρέπουν στους χρήστες να αναγνωρίζουν και να εντοπίζουν τα σύνολα των δεδομένων που τους ενδιαφέρουν ανάμεσα σε πολλαπλά αρχεία διασκορπισμένα σε πολλαπλά κέντρα δεδομένων (data centers).

θα πρέπει λοιπόν να υπάρχει η δυνατότητα να αποσυνδέονται τα δεδομένα από τα μεταδεδομένα με τρόπο ώστε να επιτυγχάνεται ο αποτελεσματικός εντοπισμός και διαμοιρασμός των δεδομένων. Πιο συγκεκριμένα, είναι απαραίτητο να υπάρχουν αρχιτεκτονικές που να μπορούν να κλιμακώνουν τους μηχανισμούς διαχείρισης και μεταφοράς δεδομένων ανεξάρτητα από τους αντίστοιχους μηχανισμούς των λειτουργικών μεταδεδομένων.

2.5 Ευρύτερες Συνέργειες

Με τις υπάρχουσες κατανεμημένες υπολογιστικές υποδομές παγκόσμιας κλίμακας, όπως αυτές υλοποιούνται με τις υφιστάμενες αρχιτεκτονικές πλέγματος, επιτυγχάνεται η διαλειτουργικότητα μεταξύ ετερογενών συστημάτων γεωγραφικά κατανεμημένων. Έτσι παρέχονται κοινές υπηρεσίες, με διάφανο τρόπο, που βασίζονται σε μια παγκόσμια υπερυπολογιστική υποδομή. Παρότι πολλοί από τους στόχους των τεχνολογιών Πλέγματος πραγματοποιήθηκαν, καταλήγοντας στην ουσιαστική στήριξη της επιστημονικής κοινότητας, ο κυρίαρχος στόχος της ενσωμάτωσης του συγκεκριμένου μοντέλου στην καθημερινότητα ενός οικιακού χρήστη δεν επιτεύχθηκε. Ένας από τους σημαντικότερους λόγους που συντέλεσε σε αυτό ήταν το γεγονός ότι δημιουργήθηκαν πολλοί νέοι μηχανισμοί και πρωτόκολλα, κυρίως ειδικού σκοπού, που δεν ήταν δυνατόν να χρησιμοποιηθούν από την ευρύτερη κοινότητα.

Το όραμα για την υλοποίηση των νέων κατανεμημένων υπολογιστικών υποδομών στην eXascale εποχή απαιτεί ευρύτερες συνέργειες με επιστημονικές και επιχειρηματικές κοινότητες. Στις τεχνολογίες Πλέγματος αυτές οι συνέργειες εντοπίζονταν κυρίως στο επίπεδο της εφαρμογής μέσω της διαλειτουργικότητας των ετερογενών υποδομών. Για την κατασκευή των υποδομών επόμενης γενιάς μπορούν να αναζητηθούν συνέργειες με τους επιχειρηματικούς κύκλους που δραστηριοποιούνται οικονομικά στον τομέα του Διαδικτύου και τις εφαρμογές που εξαρτώνται από τα “Big Data”. Αυτό, κυρίως, εκφράζεται με την αναγνώριση κοινών προβλημάτων μεταξύ των δυο κοινοτήτων.

Αντιπροσωπευτικό παράδειγμα είναι η προσπάθεια για μείωση στην κατανάλωση ενέργειας μέσω της δημιουργίας νέου τύπου υλικού (hardware), όπως παρουσιάστηκε στις παραγράφους 2.1-2.3. Οι συνέργειες αυτές θα πρέπει να υλοποιηθούν με τέτοιο τρόπο ώστε να μην δημιουργήσουν αρχιτεκτονικές και hardware ειδικού σκοπού όπως έγινε με τα πρωτόκολλα και τις εφαρμογές που δημιουργήθηκαν για την επίτευξη της διαλειτουργικότητας των υποδομών Πλέγματος. Θα πρέπει να υπάρχουν διαδικασίες εξέλιξης και όχι διαδικασίες πλήρους αναδόμησης και επανεκκίνησης, όπως αναφέρεται σε αρκετές μελέτες μεγάλων κοινοπραξιών προς κρατικές και διακρατικές οντότητες [45]. Τέτοιες μεθοδολογίες σχεδόν ποτέ δεν πέτυχαν ευρύτερους στόχους αλλά μόνο συγκεκριμένους ειδικούς σκοπούς. Κάτι τέτοιο δεν θα είναι αποδεκτό στην eXascale εποχή, καθώς τέτοιας κλίμακας υποδομές μπορούν να υλοποιηθούν μόνο με ευρύτερες συνέργειες και κοινούς στόχους.

Το ΙΚΑΡΟΣ επιχειρεί να δομήσει μια πλατφόρμα συνεργιών, στο παραπάνω οικοσύστημα, λαμβάνοντας υπ όψιν τις τεχνολογικές εξελίξεις και κατευθύνσεις στο επίπεδο του υλικού. Λειτουργεί ως ένα στρώμα μεταξύ των εφαρμογών και του υλικού και προτείνει λύσεις στα βασικά προβλήματα που αφορούν την μετάβαση από τα petascale στα eXascale αποθηκευτικά συστήματα.

3. ΤΟ ΠΛΑΙΣΙΟ ΙΚΑΡΟΣ

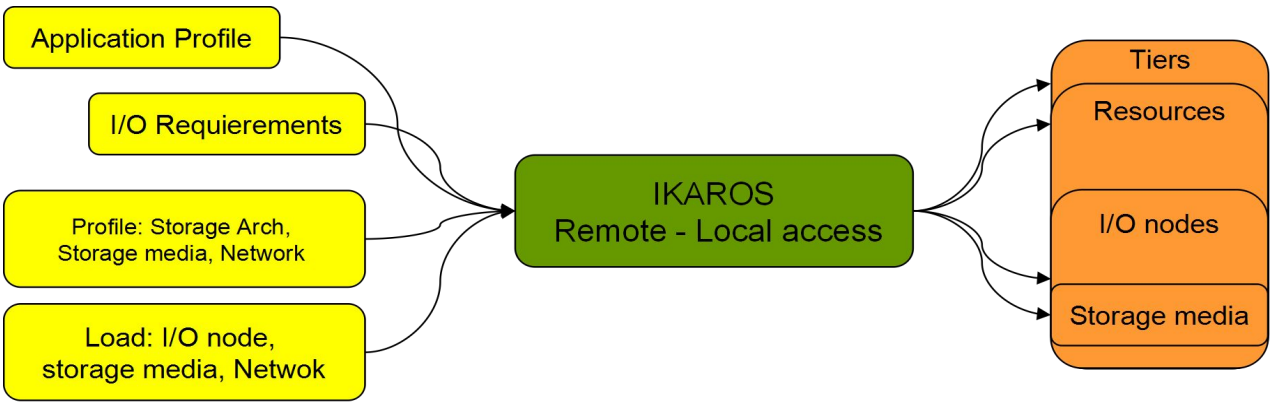
Το πλαίσιο ΙΚΑΡΟΣ αναπτύχθηκε στοχεύοντας:

1. Στην άρση των πολλαπλών επιπέδων συντονισμού στην συνολική ροή των δεδομένων (τοπική-απομακρυσμένη πρόσβαση), διατηρώντας ταυτόχρονα την αυτονομία των επιπέδων.
2. Στην δυνατότητα δημιουργίας συνεργιών μεταξύ ευρύτερων κοινοτήτων.
3. Στην κλιμάκωση του διαθέσιμου εύρους ζώνης (I/O και δίκτυο) με κόστος ανάλογο με αυτό της κλιμάκωσης της χωρητικότητας των αποθηκευτικών συστημάτων καθώς και στην δημιουργία υποδομών που θα απαιτούν εξαιρετικά μικρότερη ποσά ηλεκτρικής ενέργειας.
4. Στην αντιμετώπιση των προβλημάτων κλιμάκωσης των μηχανισμών μεταδεδομένων.

Για να επιτευχθούν τα παραπάνω το ΙΚΑΡΟΣ δομήθηκε ως ένα “λεπτό” στρώμα (thin layer) που έχει τη δυνατότητα να προσφέρει υπηρεσίες σε πολλαπλά επίπεδα, χρησιμοποιεί ένα μεγάλο αριθμό I/O κόμβων χαμηλής κατανάλωσης ενέργειας και που επιτρέπει την απομόνωση των λειτουργιών I/O μίας διεργασίας από τις αντίστοιχες των άλλων διεργασιών στοχεύοντας στην μέγιστη αξιοποίηση των διαθέσιμων πόρων.

Πιο συγκεκριμένα, το μοντέλο ανάπτυξης (Deployment model) του ΙΚΑΡΟΣ επιτρέπει το διαχωρισμό των υπολογιστικών από τους αποθηκευτικούς πόρους χωρίς όμως να αποτρέπει το αντίθετο, σε περίπτωση που αυτό μπορεί να θεωρηθεί επωφελές. Έχει τη δυνατότητα να εκτελεί διεργασίες εγγραφής σε διαφορετικές περιοχές του αρχείου, επιτρέποντας και τις ταυτόχρονες (διαμοιραζόμενες) εγγραφές (Concurrent-shared writes). Εκμεταλλεύεται πλήρως την βελτιστοποιημένη λειτουργία του HTTP για αρχεία της τάξεως των MBs και υλοποιεί τεχνικές caching και buffering στην πλευρά του πελάτη. Επιτρέπει την εγγραφή σε οποιοδήποτε σημείο του αρχείου (Append mode) σε αντίθεση με το Hadoop Distributed File System (HDFS).

Το ΙΚΑΡΟΣ επιτρέπει στους χρήστες και στις εφαρμογές να γνωρίζουν την ακριβή απεικόνιση των κομματιών που αποτελούν ένα αρχείο (Data layout) και παρέχει έναν πολύ μεγάλο αριθμό παραμετροποιήσεων με δυναμικό τρόπο. Η συγκεκριμένη λειτουργία αξιολογείται ως ένα από τα σημαντικότερα χαρακτηριστικά του ΙΚΑΡΟΣ καθώς ουσιαστικά παρέχει, μέσω των αποφάσεων για το εκάστοτε Data Layout, στο χρήστη την άμεση διαχείριση των πόρων (I/O κόμβοι, αποθηκευτικά μέσα) ανεξάρτητα από το επίπεδο που αυτοί ενεργούν. Αυτή η λογική οδηγεί στη βέλτιστη διαχείριση τους με σκοπό την επίτευξη της μέγιστης δυνατής I/O απόδοσης. Σε αυτό συντελεί και το γεγονός ότι παρέχεται η δυνατότητα εισαγωγής στο ΙΚΑΡΟΣ δεδομένων που αφορούν το προφίλ της εφαρμογής, τις I/O απαιτήσεις, το φορτίο του δικτύου και των κόμβων όπως και το προφίλ της αποθηκευτικής υποδομής (Σχήμα 2) [84]. Στη παρούσα φάση τα συγκεκριμένα δεδομένα παρέχονται στο σύστημα με μη αυτοματοποιημένο τρόπο.



Σχήμα 2: Το Πλαίσιο ΙΚΑΡΟΣ

Το ΙΚΑΡΟΣ είναι συμβατό (Compatibility) με τα περισσότερα λειτουργικά συστήματα, λόγω της χρήσης του HTTP ενώ λόγω της αρχιτεκτονικής του αλλά και της χρήσης του HTTP παρέχει μια non-blocking λειτουργία στην συνολική ροή των δεδομένων (τοπική-απομακρυσμένη πρόσβαση). Έτσι μπορεί να μειωθεί το συνολικό κόστος της όλης διαδικασίας καθώς αίρονται οι φραγμοί και τα πολλαπλά επίπεδα συντονισμού στην ροή των δεδομένων (WAN capabilities).

Η αρχιτεκτονική του ΙΚΑΡΟΣ παρέχει άμεση πρόσβαση σε κάθε I/O κόμβο, ανεξάρτητα από την ιεραρχία από την οποία αυτός ενεργεί, με αποτέλεσμα να μπορεί να διαχειρίζεται την συνολική ροή των δεδομένων (τοπική και απομακρυσμένη πρόσβαση) στο επίπεδο του δικτύου. Αντίθετα, τα συστήματα που πρέπει εναλλακτικά να συνδυαστούν ώστε να καλυφθεί η συνολική ροή των δεδομένων (π.χ PVFS2+GridFTP) είναι απομονωμένα μεταξύ τους και αναγκάζονται να εκτελούν όλες τις διεργασίες του μεταξύ τους συντονισμού στο λειτουργικό σύστημα. Αποτέλεσμα αυτής της έλλειψης συντονισμού είναι να μην μπορούν να επιτύχουν την απόδοση του ΙΚΑΡΟΣ στην συνολική ροή των δεδομένων. Η υπεροχή του ΙΚΑΡΟΣ σε αυτόν τον τομέα θα παρουσιαστεί αναλυτικά στο κεφάλαιο 6.

Τα παραπάνω χαρακτηριστικά του ΙΚΑΡΟΣ είναι πολύ σημαντικά καθώς σε eXascale περιβάλλοντα οι υποδομές θα είναι εξαιρετικά πολύπλοκες και είναι πολύ πιθανόν τα όρια των διαχειριστικών τομέων ή η εσωτερική τους οργάνωση να είναι δυσδιάκριτα από τις εφαρμογές. Γίνεται αντιληπτό ότι το ΙΚΑΡΟΣ παρουσιάζει σημαντικά πλεονεκτήματα καθώς συνδυάζει χαρακτηριστικά από τα δύο άλλα συστήματα ενώ ταυτόχρονα υιοθετεί μηχανισμούς και τεχνικές που του επιτρέπουν να λειτουργεί αποδοτικότερα σε περισσότερα επίπεδα.

Στον πίνακα 1 συνοψίζονται τα χαρακτηριστικά λειτουργίας του ΙΚΑΡΟΣ, σε σχέση με το Parallel Virtual File System (PVFS2) και το HDFS. Επιλέχθηκε η σύγκριση του ΙΚΑΡΟΣ με τα PVFS2 και HDFS καθώς και τα δυο αποτελούν εξαιρετικής ποιότητας συστήματα, αντιπροσωπευτικά της κατηγορίας τους, ενώ ταυτόχρονα λογίζονται ως ανοιχτά λογισμικά.

Πίνακας 1: Χαρακτηριστικά λειτουργίας: ΙΚΑΡΟΣ, PVFS2,HDFS

	Hadoop Distributed File System (HDFS)	**Parallel Virtual File System (PVFS2)**	**IKAROS File System (IFS)**
Deployment model	Co-locates compute and storage on the same node (beneficial to Hadoop/MapReduce model where computation is moved closer to data)	Separate compute and storage nodes	Separate compute and storage nodes
Concurrent (shared) writes	Not supported – allows only one write per file	Optimized writes to different regions of a file	Optimized writes to different regions of a file
Small file operations	Not optimized for small files; client-side buffering aggregates many small requests to one file into one large request	Use few optimizations for packing small files, but the lack of client-side buffering or caching may result in high I/O overhead	Fully exploitation of the HTTP optimization for small files (in scale of MBs), supporting client-side caching due to IKAROS architecture
Append mode	Write once semantics that allows file appends using a single writer only	Full write anywhere and rewrite support	Full write anywhere and rewrite support
Buffering	Client-side read-ahead and write-behind staging improves bandwidth, but reduces durability guarantees and limits support for consistency semantics	No client-side prefetching or caching, provides improved durability and consistency for write	Client-side caching improves bandwidth and guarantees durability and consistency for writes due to the session separation
Data layout	Exposes mapping of chunks to data-nodes to Hadoop applications	Maintains stripe layout information as extended attributes but not exposed to applications	Exposes mapping of chunks to applications and users
Compatibility	Custom API and semantics for specific users	UNIX FS	UNIX, WINDOWS and MAC FS (through DavFS). HTTP client access (browser, curl, wget....)
WAN capabilities	Can be exported through webdav	Can be exported through pNFS	Supports parallel channels WAN data transfers, stripping servers, third party data transfers requests by extending HTTP capabilities through the IKAROS module. Optionally, it can communicate with GridFTP servers, by using the GridFTP API, and can be exported through webdav

3.1 Στόχοι του ΙΚΑΡΟΣ

Στη συνέχεια αναφέρονται πιο αναλυτικά οι στόχοι του ΙΚΑΡΟΣ με βάση τις ερευνητικές προκλήσεις που αναφέρθηκαν στα προηγούμενα κεφάλαια. Το ΙΚΑΡΟΣ στο να επιτύχει:

Καλύτερο συντονισμό μεταξύ των πολλαπλών στρωμάτων λογισμικού στην συνολική ροή των δεδομένων (τοπική-απομακρυσμένη πρόσβαση), διατηρώντας ταυτόχρονα την αυτονομία των επιπέδων.

Οι λειτουργίες που απαιτούνται για την υλοποίηση των υπηρεσιών που δομούν κατανεμημένες υπολογιστικές υποδομές μεγάλη κλίμακας συνήθως αναπτύσσονται αποσπασματικά και ακολουθούν την λογική των διαφορετικών επιπέδων. Έτσι, οι λειτουργίες εξειδικεύονται και δεν επιτυγχάνεται ο ομοιόμορφος σχεδιασμός των μεθόδων που συνολικά διατρέχουν την υποδομή. Αναμφισβήτητα, η εξειδίκευση ανά κατηγορία και ανά περίπτωση έχει εξαιρετικά αποτελέσματα, σε μικρή κλίμακα, αλλά είναι φανερό πως η επέκταση αυτού του μοντέλου έχει φτάσει στα όρια του.

Αυτό γίνεται άμεσα αντιληπτό, αν αναλυθεί η εξέλιξη των τεχνολογιών πλέγματος (Grid). Το υπολογιστικό Πλέγμα φιλοδοξούσε την μετεξέλιξη του Διαδικτύου από μία υποδομή στην οποία απλά διαμοιράζεται πληροφορία σε μια υποδομή που με διάφανο τρόπο θα διαμοιράζεται υπολογιστική και αποθηκευτική ισχύ, δεδομένα και εφαρμογές. Για την ανάπτυξη των κατανεμημένων υποδομών μεγάλης ή και παγκόσμιας κλίμακας χρησιμοποιήθηκε η λογική της αποσπασματικής ανάπτυξης των υπηρεσιών και της πλήρους απομόνωσης τους. Τεχνικά, αυτό υλοποιήθηκε με τη διαλειτουργικότητα των επί μέρους ετερογενών υποδομών.

Σε περιβάλλοντα τοπικού δικτύου οι κατανεμημένες υποδομές αντιμετωπίζουν τις εφαρμογές, όσον αφορά τα δεδομένα, με όρους εισόδου/εξόδου στο επίπεδο του παράλληλου συστήματος αρχείων (τοπική πρόσβαση). Ενώ στα δίκτυα ευρείας περιοχής αναφέρονται στα δεδομένα με όρους μεταφοράς και όχι εισόδου/εξόδου (I/O). Με αυτήν την λογική δομούνται απομονωμένες “νησίδες” και απαιτούνται γέφυρες επικοινωνίας ή πολλαπλά επίπεδα συγχρονισμού. Όπως αναφέρεται στο [81], τα παράλληλα συστήματα I/O συνήθως χειρίζονται τα δεδομένα σε δύο φάσεις: τοπικό I/O και απομακρυσμένο I/O. Τα δεδομένα πρώτα διαβάζονται και στην συνέχεια αναδιοργανωνόνται πριν την μεταφορά.

Για την μεταφορά χρησιμοποιούνται τεχνικές παράλληλης μεταφοράς (παράλληλα κανάλια ή striping servers). Εδώ πρέπει να σημειωθεί ότι όπως αναφέρεται στο [80] με την χρήση παράλληλων καναλιών μεταφοράς μπορεί να επιτευχθεί πολύ υψηλή ρυθμοαπόδοση καθώς αυτά συμπεριφέρονται ως ένα μεγάλο κανάλι-stream. θα πρέπει όμως να ρυθμιστούν πολύ προσεκτικά οι διάφορες παράμετροι του TCP καθώς πολύ εύκολα μπορεί να κορεστεί το δίκτυο και τελικά να σημειωθεί σημαντική πτώση της απόδοσης. Η συγκεκριμένες τεχνικές ενδείκνυνται για μη κορεσμένα high speed δίκτυα. Ο αναφερόμενος όγκος δεδομένων είναι της τάξεως των GBs και TBs ανά μεταφορά και αφορά μεταφορές στο επίπεδο των αποθηκευτικών μέσων (HDDs, SSDs).

Έτσι για την WAN διαχείριση των δεδομένων επιτυγχάνονται πολύ υψηλοί ρυθμοί μεταφοράς λόγω της χρήσης παράλληλων τεχνικών μεταφοράς. Αλλά εξαιτίας της διαφορετικής λογικής που ακολουθείται στην σχεδίαση των υπηρεσιών δεν είναι εφικτό να συνεχιστεί η “παραλληλία” της μεταφοράς άμεσα στο ίδιο βήμα προς το τοπικό επίπεδο. Παρότι τα σύγχρονα συστήματα αρχείων λειτουργούν χρησιμοποιώντας, κατεξοχήν, τεχνικές παραλληλισμού. Αυτό που συμβαίνει είναι ότι, στην πραγματικότητα χρειάζονται δυο βήματα για την ολοκλήρωση της διαδικασίας, αφού οι λειτουργίες WAN και LAN είναι απομονωμένες και η μία λειτουργία δεν γνωρίζει την αρχιτεκτονική της άλλης.

Δυνατότητα δημιουργίας συνεργιών μεταξύ ευρύτερων κοινοτήτων.

Σε ένα τόσο πολύπλοκο περιβάλλον είναι σαφές πως οι υποδομές δεν μπορεί να είναι ομοιογενείς και οι υπηρεσίες που τις υλοποιούν θα πρέπει να μπορούν να λειτουργούν αυτόνομα και να μην εξαρτώνται από τις υλοποιήσεις άλλων επιπέδων. Αυτή η "απομόνωση" των επιπέδων δεν δημιουργήθηκε τυχαία, καθώς παλαιότερα μοντέλα που επιχειρούσαν κοινή αντιμετώπιση και εξαρτούσαν τις υπηρεσίες του ενός επιπέδου από τις υλοποιήσεις των υπηρεσιών των άλλων επιπέδων δεν είχαν τα επιθυμητά αποτελέσματα και οδηγούσαν σε μη επεκτάσιμες λύσεις. Η εξειδίκευση αυτή των υπηρεσιών οδήγησε σε βέλτιστες αποδόσεις στα επιμέρους ζητήματα και επέτρεψε την υλοποίηση των εφαρμογών μεγάλης κλίμακας που σχεδιάστηκαν τη προηγούμενη δεκαετία ώστε να λειτουργήσουν στη παρούσα.

Οι απαιτήσεις των εφαρμογών όπως αυτές παρουσιάζονται από τις αναφορές των επιστημονικών ομάδων που δραστηριοποιούνται στους τομείς της κλιματολογικής αλλαγής, της φυσικής υψηλών ενεργειών και άλλων και που θα υλοποιηθούν με χρονικό ορίζοντα το 2022 κατατείνουν στο ότι οι υπάρχουσες υποδομές δεν θα μπορούν να ανταποκριθούν αν δεν ανασχεδιαστούν. Τεχνικά, αυτό συνεπάγεται ότι θα πρέπει τα σημερινά Petascale συστήματα να εξελιχθούν σε eXascale. Η ευρύτερη επιστημονική κοινότητα, όπως αυτή εκφράζεται από αναφορές επιστημονικών ομάδων και τεχνικές μελέτες κρατών, συγκλίνει στο ότι είναι αναγκαίο να τοποθετηθεί η διαλειτουργικότητα μεταξύ ετερογενών υποδομών σε ένα ευρύτερο πλαίσιο που θα επιτρέπει να δομηθούν συνέργειες μεταξύ ευρύτερων κοινοτήτων [45].

Το πλαίσιο ΙΚΑΡΟΣ φιλοδοξεί να ανταποκριθεί στα παραπάνω ζητήματα ως μία πλατφόρμα που θα επιτρέπει ευρύτερες συνέργειες. Έτσι προσπαθεί να αναγνωρίσει πιθανές κοινές λειτουργίες και συμπεριφορές μεταξύ των υπηρεσιών στα διαφορετικά επίπεδα. Αυτή η προσέγγιση μπορεί δυνητικά να οδηγήσει στην δημιουργία υποδομών που θα διατρέχονται από κοινές πρακτικές και θα μπορούν να υλοποιηθούν με μειωμένο συνολικό κόστος.

Το πρωτόκολλο HTTP επιλέχθηκε ως ο βασικός μηχανισμός πάνω στον οποίο δομήθηκε το όλο πλαίσιο. Η επιλογή του HTTP επιτρέπει την επίτευξη συνεργασιών μεταξύ ευρύτερων κοινοτήτων, στο επίπεδο του λογισμικού, επιτυγχάνοντας την ομοιόμορφη λειτουργία στη συνολική ροή των δεδομένων. Ενώ ταυτόχρονα, διατηρείται η αυτονομία υλοποίησης των υπηρεσιών των διαφόρων επιπέδων. Γίνεται φανερό ότι η επιλογή των κατάλληλων τεχνοοικονομικών παραμέτρων, σε αυτή την περίπτωσή η επιλογή του HTTP, διαδραματίζει κυρίαρχο ρόλο στο κατά πόσο η υποδομή θα επιτρέπει την δημιουργία ευρύτερων συνεργιών ή όχι.

Οι τεχνοοικονομικοί παράγοντες συνήθως υποβαθμίζονται και τοποθετούνται ένα επίπεδο κάτω από την θεωρητική προσέγγιση των ζητημάτων. Όμως, μια λεπτομερής ανάλυση της επιστήμης της πληροφορικής θα δείξει ότι είναι αυτοί οι μηχανισμοί, οι τεχνοοικονομικοί, οι οποίοι καθορίζουν τις εξελίξεις σε επιχειρηματικό και επιστημονικό επίπεδο. Η δημιουργία απομονωμένων τεχνολογικών "νησίδων" που παρέχουν την ίδια πληροφορία με βάση την αξία που έχει για τον εκάστοτε χρήστη/πελάτη και όχι με βάση την αξία παραγωγής ή αναπαραγωγής του προϊόντος αποτελεί πρακτική που στηρίζεται σε διαχρονικές αξίες της οικονομικής θεωρίας.

Οι υλοποιήσεις αυτές οφείλονται σε μεγαλύτερο βαθμό στις θεωρίες του μονοπωλίου και του lock-in (το "κλείδωμα" του χρήστη/πελάτη σε μια συγκεκριμένη τεχνική υλοποίηση) [47] παρά σε θεωρητικές προσεγγίσεις που αφορούν την επιστήμη της πληροφορικής. Μπορεί με ασφάλεια να εξαχθεί το συμπέρασμα πως η επιλογή των

τεχνοοικονομικών μηχανισμών, που υλοποιούν μια υποδομή, στηρίζεται σε μεγάλο βαθμό στην οικονομική επιστήμη, η οποία αποτελεί μια αποτύπωση της κοινωνικής συμπεριφοράς. Στη δεδομένη περίπτωση αυτό έχει να κάνει με το ποιος έχει την δυνατότητα να χειριστεί τα δεδομένα και κατά πόσο ο ιδιοκτήτης (καθημερινός χρήστης/ερευνητής) είναι ο αυτός που έχει τον απόλυτο έλεγχο σε αυτά.

Η συγκεκριμένη λογική έχει σαφείς προεκτάσεις και στον τομέα της ασφάλειας των πληροφοριακών συστημάτων. Συνήθως οι παραβιάσεις ασφαλείας δεν οφείλονται σε λανθασμένη επιλογή πρωτοκόλλου ασφαλείας, αλλά στο γεγονός ότι αυτοί που είναι επιφορτισμένοι με την φύλαξη των υποδομών ή των δεδομένων δεν είναι και αυτοί που θα υποστούν τις συνέπειες της παραβίασης. Αυτό συνεπάγεται ότι, θα πρέπει να δομηθούν μηχανισμοί που θα επιτρέπουν την διαχείριση των δεδομένων και των υποδομών απευθείας από τους χρήστες. Το ΙΚΑΡΟΣ ενστερνίζεται απόλυτα αυτή την λογική και επιτρέπει την δημιουργία υβριδικών υποδομών που συνδυάζουν την ποιότητα υπηρεσίας των υφιστάμενων επιστημονικών υπολογιστικών υποδομών και την ευχρηστία που που παρέχουν οι Web 2.0 τεχνολογίες.

Κλιμάκωση του διαθέσιμου εύρους ζώνης (I/O και δίκτυο) με κόστος ανάλογο με αυτό της κλιμάκωσης της χωρητικότητας των αποθηκευτικών συστημάτων. Δημιουργία υποδομών που θα απαιτούν εξαιρετικά μικρότερη κατανάλωση ηλεκτρικής ενέργειας.

Με την ανάπτυξη των επιστημονικών πειραμάτων παγκόσμιας κλίμακας, όπως του LHC, παρατηρείται ότι οι διεργασίες που υποβάλλονται στις υπολογιστικές υποδομές προς επίλυση εξαρτώνται όλο και περισσότερο από τα δεδομένα, που απαιτούν, για την διεκπεραίωση της διεργασίας. Συνήθως τα δεδομένα που απαιτούνται για την εκτέλεση μιας διεργασίας μπορεί να περιλαμβάνουν συλλογές δεδομένων που συνολικά είναι της τάξεως του TB. Έτσι γίνεται αντιληπτό ότι οι προς εκτέλεση διεργασίες θα πρέπει να μεταφερθούν και να εκτελεστούν στα υπολογιστικά κέντρα που φιλοξενούν τις απαιτούμενες συλλογές δεδομένων, καθώς το αντίστροφο δεν θα ήταν αποδοτικό.

Η περαιτέρω ανάλυση αυτού του μοντέλου λειτουργίας κάνει φανερό ότι οι υφιστάμενες κατανεμημένες υπολογιστικές υποδομές μετατρέπονται από μη κεντρικοποιημένες υποδομές παγκόσμιας κλίμακας, με γεωγραφικά κατανεμημένους πόρους, σε μια κεντρικοποιημένη εγκατάσταση. Έτσι περιορίζεται στην ουσιαστική συμμετοχή μόνο πολύ λίγων υπολογιστικών κέντρων. Για να μπορέσει μια υποδομή να φιλοξενήσει αυτόν τον όγκο των δεδομένων, αλλά κυρίως για να μπορέσει να ανταποκριθεί στον ρυθμό αύξησης τους θα πρέπει να διαθέτει εξειδικευμένες αποθηκευτικές υποδομές καθώς και εξειδικευμένο προσωπικό για να τις διαχειριστεί. Τις προϋποθέσεις αυτές δεν μπορούν να καλύψουν οργανισμοί μικρού ή μεσαίου επιπέδου, με αποτέλεσμα να παρουσιάζονται τα συγκεκριμένα φαινόμενα.

Το πρόβλημα διαχείρισης και μεταφοράς των δεδομένων γίνεται πιο επιτακτικό αν συνυπολογιστεί η απαιτούμενη μελλοντική κλιμάκωση των υποδομών σε eXascale περιβάλλοντα. Η διαχείριση των δεδομένων και η ροή τους, μεταξύ υπολογιστικών και αποθηκευτικών πόρων, δημιουργεί τεράστιους περιορισμούς στις εφαρμογές μεγάλης κλίμακας. Αυτό ουσιαστικά οφείλεται στο ότι στα αποθηκευτικά συστήματα υπάρχει αναντιστοιχία μεταξύ της κλιμάκωσης της διαθέσιμης χωρητικότητας και του διαθέσιμου εύρους ζώνης (I/O και δίκτυο). Οι νέες υποδομές θα πρέπει να έχουν την δυνατότητα κλιμάκωσης σε κόμβους της τάξεως του εκατομμυρίου, ενώ ταυτόχρονα είναι πιθανόν να πρέπει να ανταποκριθούν σε αιτήματα εισόδου/εξόδου της τάξεως του δισεκατομμυρίου/ δευτερόλεπτο.

Γίνεται ακόμα αντιληπτό πως η ηλεκτρική τροφοδοσία υποδομών που διαθέτουν αριθμό κόμβων της τάξεως του εκατομμυρίου με τις υφιστάμενες επιδόσεις στη κατανάλωση ηλεκτρικής ενέργειας πιθανότατα θα οδηγήσει σε αδυναμία υλοποίησης τους. Τα

ζητήματα που δημιουργούνται είναι τεράστια από οικονομικής, περιβαλλοντολογικής αλλά και τεχνικής απόψεως. Ζητήματα όπως η θέση και η ψύξη των συγκεκριμένων υποδομών θα έχουν πρωτεύοντα ρόλο για την υλοποίησή τους. Οι ερευνητικές προκλήσεις που αναδύονται είναι εξαιρετικά σημαντικές καθώς θα πρέπει να αυξηθούν, αισθητά, οι διαθέσιμοι κόμβοι αποθήκευσης για να ξεπεραστούν τα προβλήματα απόδοσης των I/O λειτουργιών κάτι που όμως θα εκτινάξει τα επίπεδα κατανάλωσης ενέργειας σε απαγορευτικά επίπεδα.

Είναι προφανές ότι τα αποθηκευτικά συστήματα θα αποτελέσουν την Αχίλλειο πτέρνα των μελλοντικών eXascale υποδομών, αν αυτά δεν αναδιοργανωθούν πλήρως. Όπως αναφέρεται στο [48], μπορεί πολύ απλά να αναδειχθεί το πρόβλημα αν και μόνο γίνει προσπάθεια επέκτασης μιας υπάρχουσας υπερυπολογιστικής υποδομής σε έναν αριθμό κόμβων της τάξεως του εκατομμυρίου. Αρκεί και μόνο να μελετηθεί η παράμετρος της αρχικοποίησης ενός τέτοιου συστήματος (boot time, ο χρόνο που απαιτείται έτσι ώστε όλοι οι κόμβοι του συστήματος να βρεθούν σε κατάσταση λειτουργίας και οι κόμβοι διαχείρισης του υπερυπολογιστή να πραγματοποιήσουν τους απαραίτητους ελέγχους σχετικά με την διαθεσιμότητα των υποσυστημάτων, ώστε το όλο σύστημα να είναι διαθέσιμο στους χρήστες). Σύμφωνα με την μελέτη που έχει διεξαχθεί στο [48] σε ένα BlueGene/P υπερυπολογιστικό σύστημα ο χρόνος που απαιτείται για την αρχικοποίηση αυτών των συστημάτων αυξάνεται γραμμικά σε σχέση με τον αριθμό των κόμβων. Κάτι που σημαίνει, ότι σε μία υλοποίηση με ένα εκατομμύριο κόμβους ο χρόνος και μόνο για την αρχικοποίηση του συστήματος μπορεί να ξεπεράσει τα 25 χιλιάδες δευτερόλεπτα (7 + ώρες) [40, 48].

Το ΙΚΑΡΟΣ απαντώντας σε αυτές τις προκλήσεις επιτρέπει τη χρήση ενός μεγάλου αριθμού SOHO-NAS συσκευών με σκοπό τη δημιουργία ενός αποθηκευτικού συστήματος υψηλής απόδοσης. Οι συγκεκριμένες συσκευές έχουν πολύ μικρό κόστος κτήσης, πολύ χαμηλή κατανάλωση ενέργειας, θεωρούνται plug and play συσκευές και μπορεί να αυξήσουν το συνολικό διαθέσιμο εύρος ζώνης (I/O και δίκτυο) εξαιρετικά οικονομικά. Σκοπός του πλαισίου είναι η χρήση αποθηκευτικών συσκευών χαμηλών τεχνικών προδιαγραφών και χαμηλής κατανάλωσης ενέργειας για την δημιουργία υψηλής απόδοσης αποθηκευτικών σχηματισμών on demand. Η συγκεκριμένη λογική θα μπορούσε να επιτρέψει την απεμπλοκή των λειτουργιών καθώς και των αστοχιών που αυτές παρουσιάζουν από την κλίμακα στην οποία ενεργούν οι επιμέρους συσκευές.

Με αυτόν τον τρόπο γίνεται, καταρχήν, εφικτό να δημιουργηθούν υποδομές που θα αποτελούνται από έναν αριθμό κόμβων στην κλίμακα του εκατομμυρίου, η κατανάλωση σε ηλεκτρική ενέργεια για το συνολικό σύστημα μπορεί να μειωθεί σε σημαντικό βαθμό ενώ το διαθέσιμο εύρος ζώνης μπορεί να είναι ικανοποιητικό.

Αντιμετώπιση των προβλημάτων κλιμάκωσης των μηχανισμών μεταδεδομένων.

Τα παράλληλα συστήματα αρχείου όπως το Parallel Virtual File System (PVFS2) [19], το Lustre [20] και το General Parallel File System (GPFS) [18, 21] προσφέρουν εξαιρετικά επεκτάσιμες λύσεις σε ένα τόσο ανταγωνιστικό περιβάλλον. Τα υψηλής απόδοσης συστήματα αρχείου όπως τα προαναφερόμενα είναι αρκετά εξειδικευμένα και πολύ συχνά στοχεύουν σε συγκεκριμένες πλατφόρμες υλικού. Παρόλα αυτά, τα λογισμικά που είδη υπάρχουν καθώς και άλλοι παράγοντες που συνεισφέρουν στην ετερογενή φύση των πελατών δημιουργούν ένα σχίσμα μεταξύ συστημάτων αρχείου και χρηστών [22]. Ως πελάτης ορίζεται ο κεντρικός κόμβος μιας συστοιχίας υπολογιστών, ένας εξυπηρετητής που παρέχει υπηρεσίες πρόσβασης στο αποθηκευτικό σύστημα, μια εφαρμογή που χρησιμοποιεί εργαλεία που δρουν ως πελάτες σε HTTP εξυπηρετητές, ή ακόμα και ένας χρήστης που χρησιμοποιεί τον φυλομετρητή του ή εργαλεία πελάτη όπως το wget [25] ή το curl [26].

Οι σύγχρονες εφαρμογές απαιτούν απομακρυσμένη πρόσβαση αρχείου και μια συνολική αντιμετώπιση στη ροή των δεδομένων (τοπική-απομακρυσμένη πρόσβαση). Τα εργαλεία που προσφέρουν απομακρυσμένη πρόσβαση δεδομένων όπως το NFS και το GridFTP [4] ξεπερνούν αυτούς τους περιορισμούς αλλά αποτυγχάνουν στο να παρέχουν καθολική διαφανή (transparent) και επεκτάσιμη απομακρυσμένη πρόσβαση δεδομένων [23]. Είναι προφανές ότι τα συγκεκριμένα συστήματα δεν έχουν δυνατότητες κλιμάκωσης σε eXascale περιβάλλοντα. Ένας λόγος είναι η αδυναμία κλιμάκωσης της οντότητας των μεταδεδομένων αρχείου, γίνεται αναφορά σε λειτουργικά μεταδεδομένα (file system metadata) .

Οι υφιστάμενοι μηχανισμοί μεταδεδομένων λειτουργούν στατικά χωρίς να αντιλαμβάνονται την δυναμική της εφαρμογής. Οι εφαρμογές που ενεργούν σε ένα τόσο πολύπλοκο περιβάλλον απαιτούν δυναμική διαχείριση της υποδομής από τους ίδιους τους χρήστες άλλα και τις εφαρμογές με ελάχιστη “παρεμβολή” από τους διαχειριστές. Για παράδειγμα, on the fly πρόσθεση και αφαίρεση κόμβων αποθήκευσης στο επίπεδο της υποδομής ή της εκάστοτε διεργασίας. Τα υφιστάμενα συστήματα μεταδεδομένων δεν έχουν την δυνατότητα να παρέχουν μια τέτοια λειτουργία. Η οντότητα μεταδεδομένων του ΙΚΑΡΟΣ επιτρέπει να δημιουργούνται υποδομές που θα κλιμακώνουν τα υποσυστήματα τους ανεξάρτητα ανάλογα με τις ανάγκες.

Οι μηχανισμοί μεταδεδομένων έχουν κυρίαρχο ρόλο στην όλη αρχιτεκτονική, αφού είναι αυτοί που επιτρέπουν να δημιουργούνται κοινές μεθοδολογίες αντιμετώπισης των προβλημάτων στην γενικότερη ροή των δεδομένων, διασφαλίζοντας παράλληλα την αυτονομία υλοποίησης των επιμέρους υπηρεσιών. Η λογική που ακολουθεί η οντότητα μεταδεδομένων του ΙΚΑΡΟΣ οδηγεί στην δημιουργία υποδομών που μειώνουν το κενό μεταξύ της διαθέσιμης χωρητικότητας και του διαθέσιμου εύρους ζώνης ενώ ταυτόχρονα επιτρέπεται την απομόνωση των λειτουργιών I/O μίας διεργασίας από τις αντίστοιχες των άλλων διεργασιών, στοχεύοντας στην μέγιστη αξιοποίηση των διαθέσιμων πόρων και του εύρους ζώνης.

Το ΙΚΑΡΟΣ στοχεύει στο να λειτουργεί ως ένα πλαίσιο που θα επιτρέπει την δημιουργία αποθηκευτικών συστημάτων νέας γενιάς, τα οποία θα μπορούν να ανταποκριθούν στις παραπάνω απαιτήσεις από πλευράς κλιμάκωσης, απόδοσης και κατανάλωσης ενέργειας. Αυτό μπορεί να επιτευχθεί δημιουργώντας κατάλληλους τεχνοοικονομικούς μηχανισμούς που θα επιτρέπουν την συνεργασία μεταξύ ευρύτερων κοινοτήτων.

3.2 Αρχιτεκτονική του ΙΚΑΡΟΣ

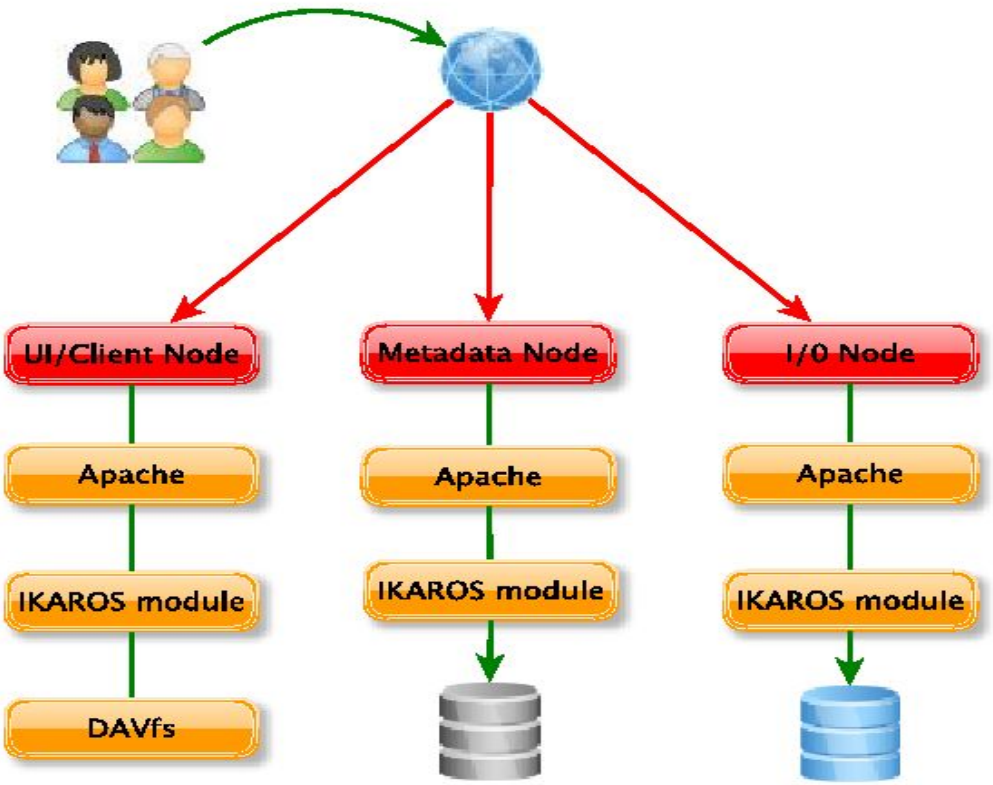
Το ΙΚΑΡΟΣ έχει σχεδιαστεί ως ένα Apache [10] module [11] που λειτουργεί και συνεργάζεται διάφανα με τις υπόλοιπες υπηρεσίες μιας κατανεμημένης υπολογιστικής υποδομής. Στις νεότερες εκδόσεις του παρέχεται ο πηγαίος κώδικας σε nodeJS, γεγονός που επιτρέπει μεγαλύτερες συνέργειες με τις πλατφόρμες που υλοποιούν web 2.0 τεχνολογίες.

Το ΙΚΑΡΟΣ επιτρέπει στα δεδομένα που συγκροτούν ένα αρχείο να είναι διασκορπισμένα μεταξύ πολλαπλών δίσκων σε πολλαπλούς ετερογενείς κόμβους, σε φυσικό επίπεδο. Ταυτόχρονα παρέχει την δυνατότητα στο αποθηκευτικό σύστημα να έχει πρόσβαση σε αυτά σαν ενιαίο αρχείο ή εν μέρει, ανάλογα με τα αιτήματα των χρηστών/εφαρμογών. Στο σχήμα 3 διακρίνονται οι τρεις διαφορετικοί τύποι κόμβων. Οι κόμβοι τύπου εισόδου/εξόδου (χειρίζονται τα αιτήματα εισόδου/εξόδου), οι κόμβοι τύπου πελάτη και τέλος ο κόμβος των μεταδεδομένων. Ο κόμβος των μεταδεδομένων συντονίζει τις αλληλεπιδράσεις μεταξύ των κόμβων τύπου πελάτη και των κόμβων που είναι επιφορτισμένοι με την εξυπηρέτηση των αιτημάτων εισόδου/εξόδου.

Το ΙΚΑΡΟΣ Apache module είναι εγκατεστημένο, ταυτόχρονα, στον πελάτη στους κόμβους εισόδου/εξόδου αλλά και στους κόμβους μεταδεδομένων. Με αυτήν την ρύθμιση χρησιμοποιείται μόνο η λειτουργία HTTP GET, για όλες τις λειτουργίες της μονάδας, και αποφεύγεται η χρήση της HTTP PUT. Η συγκεκριμένη ρύθμιση παρέχει μέγιστη ευελιξία κυρίως όταν απαιτείται παραμετροποίηση για την επίτευξη μέγιστης απόδοσης. Με την επιλογή της συγκεκριμένης παραμετροποίησης σε συνδυασμό με τη χρήση τεχνικών reverse HTTP γίνεται εφικτό σε τεχνικό επίπεδο να επιτευχθεί η άμεση πρόσβαση σε οποιουδήποτε αποθηκευτικό κόμβο I/O, ανεξάρτητα από την βαθμίδα στην οποία ενεργεί και διασφαλίζοντας παράλληλα την αυτονομία υλοποίησης των επιμέρους υπηρεσιών σε κάθε επίπεδο. Η συγκεκριμένη παραμετροποίηση έχει επιλεγεί για τις μετρήσεις που θα ακολουθήσουν.

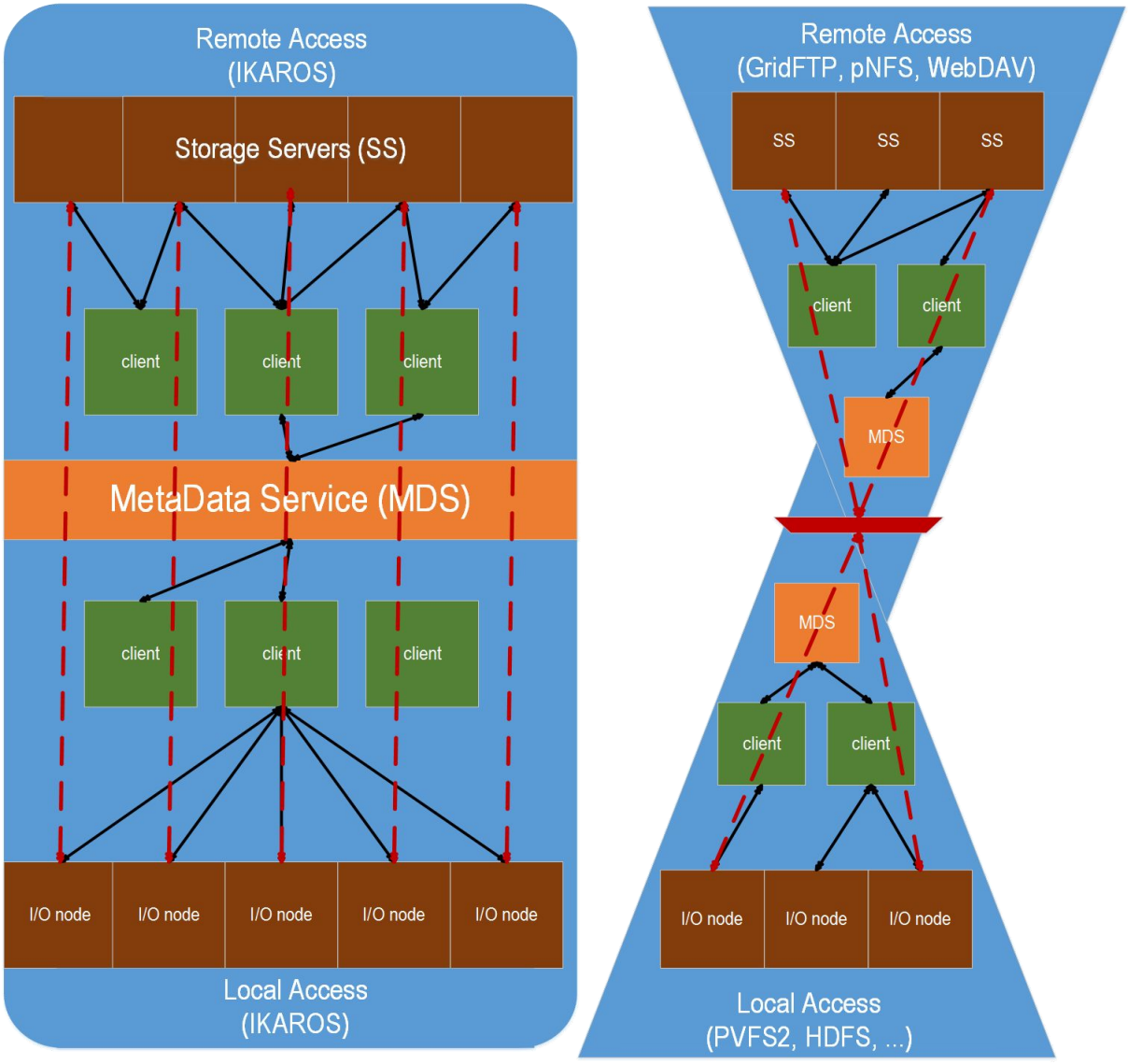
Εναλλακτικά, το ΙΚΑΡΟΣ Apache module εγκαθίσταται μόνο στην πλευρά του πελάτη. Οι κόμβους I/O και μεταδεδομένων ορίζονται ως τυπικοί HTTP εξυπηρετητές. Κατά αυτόν τον τρόπο επιτυγχάνεται η ελάχιστη πολυπλοκότητα στους κόμβους I/O. Η συγκεκριμένη ρύθμιση παρέχει σημαντικά πλεονεκτήματα στον τομέα της εύχρηστης διαχείρισης των υποδομών.

Τέλος, το ΙΚΑΡΟΣ Apache module μπορεί να εγκατασταθεί μόνο στην πλευρά του πελάτη και να χρησιμοποιήθούν τα προ-εγκατεστημένα πρωτόκολλα απομακρυσμένης πρόσβασης που πιθανόν να διαθέτουν οι SOHO-NAS συσκευές, όπως το Network File System (NFS)[27], PNFS (NFSV4.1), PVFS2, Common Internet File System (CIFS) [28], File Transfer Protocol (FTP) [29], Secure Shell (SSH) [30] και άλλα. Κατά αυτόν τον τρόπο επιτυγχάνεται η ελάχιστη δυνατή παραμετροποίηση στους κόμβους, που σε συγκεκριμένες περιπτώσεις μπορεί να είναι επιθυμητή.



Σχήμα 3: Αρχιτεκτονική ΙΚΑΡΟΣ

Το ΙΚΑΡΟΣ χρησιμοποιεί την λογική που εισάγει το σχήμα 3 με ενιαίο τρόπο σε σχέση με τα διάφορα επίπεδα ροής των δεδομένων. Στον αντίποδα, τα υφιστάμενα συστήματα επαναλαμβάνουν αυτήν την κατηγοριοποίηση (ή μέρος της) σε κάθε επίπεδο (τοπική-απομακρυσμένη πρόσβαση). Όπως φαίνεται και στο σχήμα 4 η αρχιτεκτονική του ΙΚΑΡΟΣ επιτρέπει την ροή των δεδομένων από το ένα επίπεδο στο άλλο σε ένα βήμα χωρίς την χρήση ενδιάμεσων επιπέδων συγχρονισμού και αναδιοργάνωσης, όπως συμβαίνει σε ένα τυπικό σύστημα. Γίνεται αντιληπτό πως η οντότητα μεταδεδομένων (Metadata Server - MDS), η οποία χειρίζεται αποκλειστικά λειτουργικά μεταδεδομένα διαδραματίζει κυρίαρχο ρόλο στην αρχιτεκτονική του ΙΚΑΡΟΣ.



Σχήμα 4: Ροή δεδομένων

Το ΙΚΑΡΟΣ δομήθηκε ως ένα πλαίσιο που υλοποιεί τις βασικές του λειτουργίες στηριζόμενο στο πρωτόκολλό HTTP, και τις επεκτάσεις του, για τους παρακάτω τεχνικούς λόγους:

- Είναι το πρωτόκολλο που έχει υλοποιηθεί περισσότερο από οποιοδήποτε άλλο.
- Μπορεί να προσφέρει μια εξαιρετικά πολυπληθή συλλογή από υψηλής ποιότητας λογισμικό.

- Αλληλεπιδρά ομαλά με μηχανισμούς και υποδομές που είναι ευρέως διαδεδομένες στα δίκτυα υπολογιστών, όπως το τείχος προστασίας (firewall) και η μετάφραση διεύθυνσης δικτύου (Network Address Translation – NAT).
- Είναι η βάση για τις περισσότερες υπηρεσίες ιστού και πλέγματος.
- Το HTTP/1.1 και οι επεκτάσεις του, όπως το Web-based Distributed Authoring and Versioning (WebDav) [rfc4918, 12], προσφέρουν ένα ευρύ φάσμα από μεθόδους (λήψης, τοποθέτησης, αντιγραφής, διαγραφής, κλειδώματος και ο το κάθε έξης) καθώς και μηχανισμούς χειρισμού σφάλματος και επικεφαλίδας.
- Η προσφερόμενη επικεφαλίδα εύρους (Range Header) επιτρέπει την υλοποίηση της λειτουργίας λήψης και τοποθέτησης στα επιθυμητά μέρη και όχι αναγκαστικά στο σύνολο της.
- Στην πράξη μπορεί να επιτυγχάνει τον ίδιο ρυθμό μετάδοσης δεδομένων με άλλα πρωτόκολλα που βασίζονται στο TCP [13], όπως το GridFTP [4].
- Παρέχει δυνατότητες διαλειτουργικότητας με πολιτικές ασφαλείας που βασίζονται στο μοντέλο που προσφέρει το Grid Security Infrastructure (GSI), με την χρήση του HTTPS.
- Χρησιμοποιεί ένα κανάλι για την μετάδοση των δεδομένων και την υποστήριξη των μηχανισμών ελέγχου, σε αντίθεση με το File Transfer Protocol (FTP) και το GridFTP. Γεγονός που το καθιστά, υπο συνθήκες και ανα περίπτωση, περισσότερο αποδοτικό “φιλικό” στην προσπάθεια διαχείρισης και παραμετροποίησης των συστημάτων, ενώ ταυτόχρονα υπάρχει η δυνατότητα χρήσης του πρωτοκόλλου oAuth [82].
- Δεν απαιτείται η εξαρχής επαναδημιουργία των συνδέσεων. Έχει την δυνατότητα να διοχετεύσει πολλαπλά αιτήματα στην ίδια TCP σύνδεση, ελαχιστοποιώντας με αυτόν τον τρόπο το επιπλέον κόστος (three-way handshake connection establishment overhead).
- Υπάρχουν σε σχεδόν όλες τις γλώσσες προγραμματισμού και σχεδόν σε όλα τα περιβάλλοντα υλοποιήσεις προγραμμάτων τύπου πελάτη συμβατές με το HTTP.
- Υπάρχει η δυνατότητα αξιοποίησης της ευρείας διείσδυσης που διαθέτει σε επίπεδο χρήστη και καθημερινής λειτουργίας των συστημάτων.

Στα επόμενα κεφάλαιο αναπτύσσονται, διεξοδικά, όλες οι λειτουργίες του ΙΚΑΡΟΣ, με την ακόλουθη σειρά:

- Στο κεφάλαιο 4 παρουσιάζεται η οντότητα μεταδεδομένων του ΙΚΑΡΟΣ καθώς και η λογική που αυτή εισάγει στην ροή των δεδομένων. Η οντότητα μεταδεδομένων έχει κυρίαρχο ρόλο στην συνολική λειτουργία του ΙΚΑΡΟΣ, καθώς επιτρέπει την ομοιόμορφη λειτουργία στη συνολική ροή των δεδομένων (τοπική-απομακρυσμένη πρόσβαση).
- Στο κεφάλαιο 5 παρουσιάζονται οι μηχανισμοί εισόδου/εξόδου, οι τεχνικές που επιτρέπουν στο ΙΚΑΡΟΣ να υπερέχει κυρίως κατά την εκτέλεση των διεργασιών εγγραφής και διενεργούνται δύο ομάδες μετρήσεων. Στην πρώτη ομάδα μετρήσεων εκτελούνται δοκιμές απόδοσης και φόρτου, χρησιμοποιώντας δεδομένα του πειράματος KM3NeT και οι εφαρμογές seatray [52] και ROOT [53], σε πραγματικό περιβάλλον παραγωγής. Στην δεύτερη ομάδα μετρήσεων χρησιμοποιείται το

benchmark tool IOR-HPC που διενεργεί αιτήματα τυχαίας προσπέλασης σε παράλληλα προγραμματιστικά περιβάλλοντα, όπως το MPICH.

- Στο Κεφάλαιο 6 παρουσιάζονται οι μηχανισμοί μεταφοράς δεδομένων σε δίκτυα ευρείας περιοχής καθώς και η ενσωμάτωση τεχνικών παράλληλων καναλιών μεταφοράς από το HTTP. Εδώ παρουσιάζεται η υπεροχή του ΙΚΑΡΟΣ σε πραγματικό περιβάλλον παραγωγής στην συνολική ροή των δεδομένων σε σχέση με την απαιτούμενη συνδυασμένη χρήση άλλων λογισμικών για το ίδιο λειτουργικό αποτέλεσμα. Επιπλέον διενεργούνται μετρήσεις στα πλαίσια του πειράματος CMS-LHC του CERN δείχνοντας την υπεροχή του ΙΚΑΡΟΣ στην συνολική ροή των δεδομένων.
- Στο κεφάλαιο 7 παρουσιάζεται το ΙΚΑΡΟΣ ως ένα πλαίσιο δημιουργίας on demand αποθηκευτικών σχηματισμών που επιτρέπει την απομόνωση των λειτουργιών I/O της διεργασίας από τις αντίστοιχες άλλων διεργασιών, στοχεύοντας στην μέγιστη αξιοποίηση των διαθέσιμων πόρων και του διαθέσιμου εύρους ζώνης (I/O και δίκτυο).
- Στο κεφάλαιο 8 αναλύονται οι πιθανές συνέργειες των δικτύων δεδομένων με τα τηλεπικοινωνιακά δίκτυα, υπό το πρίσμα του ΙΚΑΡΟΣ, κάτι που δυνητικά θα μπορούσε να ενισχύσει τις προσπάθειες υλοποίησης των eXascale υποδομών.

4. ΟΝΤΟΤΗΤΑ ΜΕΤΑΔΕΔΟΜΕΝΩΝ ΤΟΥ ΙΚΑΡΟΣ

Από την δεκαετία του 1980 και έπειτα έχουν υπάρξει πολυάριθμες προσπάθειες για την δημιουργία κοινόχρηστων και παράλληλων συστημάτων αρχείων όπως το NFS [54], AFS [55], GPFS, PVFS, Lustre, Panases [56], Microsoft Distributed File System [57], GlusterFS [58] , OneFS, [59] POHMELFS [60] και το XtreemFS [61]. Τα περισσότερα από αυτά τα συστήματα παρέχουν POSIX λειτουργίες και έχουν χρησιμοποιηθεί ευρέως σε συστοιχίες υπολογιστών, σε υποδομές πλέγματος και σε υπερυπολογιστές. Η μεγαλύτερη κριτική προς τα παραπάνω συστήματα έχει να κάνει με το γεγονός ότι υποθέτουν πως οι αποθηκευτικοί κόμβοι είναι πολύ λιγότεροι από τους πελάτες ή τους υπολογιστικούς κόμβους που ζητούν πρόσβαση στο εκάστοτε σύστημα αρχείων [39]. Όπως αναφέρεται και στο [48] από τους I. Raicu, I. Foster και P. Beckman το οποίο αποτελεί μια από τις πληρέστερες και ταυτόχρονα ευρύτερα αποδεκτές μελέτες για την ανάπτυξή των eXascale υποδομών η παραπάνω λογική οδηγεί σε μη ισορροπημένες αρχιτεκτονικές.

Έτσι αναπτύχθηκε ένα πλήθος από κατανεμημένα συστήματα αρχείων με σκοπό να διευθετηθεί αυτή η δυσλειτουργία. Ενδεικτικά κατανεμημένα συστήματα αρχείων είναι το GFS [62], HDFS [63], Sector [64], CloudStore [65], Ceph [66], Gfarm [67-69], MooseFS [70], Chirp [71], MosaStore [72], PAST [73], Circle [74] και το RAMCloud [75]. Παρόλα αυτά, πολλά από αυτά τα συστήματα είναι ισχυρά συνδεδεμένα με συγκεκριμένα πλαίσια εκτέλεσης διεργασιών, για παράδειγμα το Hadoop. Αυτό σημαίνει πως οι εφαρμογές που δεν τα χρησιμοποιούν θα πρέπει να τροποποιηθούν έτσι ώστε να χρησιμοποιήσουν το υποκείμενο, ασύμβατο με το POSIX, σύστημα αρχείων. Όσα μεν διαθέτουν συμβατό με το POSIX σύστημα αρχείων δεν διαθέτουν κατανεμημένους μηχανισμούς διαχείρισης μεταδεδομένων. Όσο για τα ελάχιστα που διαθέτουν κατανεμημένους μηχανισμούς διαχείρισης μεταδεδομένων, όπως το Circle και το Ceph, αυτά με τη σειρά τους αποτυγχάνουν να αποσυνδέσουν τα δεδομένα από τα μεταδεδομένα με αποδοτικό τρόπο [48].

Εδώ θα πρέπει επίσης να αναφερθεί πως ένα από τα σημαντικότερα πρόβλημα που αντιμετωπίζουν όλα τα παραπάνω συστήματα έχει να κάνει με το γεγονός ότι σε αντίθεση με το ΙΚΑΡΟΣ αδυνατούν να προσαρμόσουν δυναμικά και on demand, ανάλογα με τις απαιτήσεις των εφαρμογών, το λόγο μεταξύ του αριθμό των χρηστών-πελατών και των χρησιμοποιούμενων κάθε φορά κόμβων I/O ή σκληρών δίσκων. Η συγκεκριμένη λειτουργία του ΙΚΑΡΟΣ θα αναλυθεί διεξοδικά στο κεφάλαιο 7.

Πιο συγκεκριμένα, τα τεχνικά ζητήματα που πρέπει να αντιμετωπισθούν συνολικά από το σύστημα αρχείων και την αρχιτεκτονική υλοποίησης της αποθηκευτικής υποδομής είναι ότι:

- Το διαθέσιμο εύρος ζώνης (I/O και δίκτυο) δεν κλιμακώνει με κόστος ανάλογο με αυτό της κλιμάκωσης της διαθέσιμης χωρητικότητας των αποθηκευτικών συστημάτων στα μεγάλης κλίμακας συστήματα.
- Η I/O κίνηση στο δίκτυο μπορεί να επηρεαστεί από άλλες, μη σχετικές, διεργασίες ή αντίστοιχα να επηρεάσει την απόδοση άλλων διεργασιών.
- Η I/O κίνηση στα αποθηκευτικά συστήματα μπορεί να επηρεαστεί από άλλες, μη σχετικές, διεργασίες ή αντίστοιχα να επηρεάσει την απόδοση άλλων διεργασιών.

4.1 Αρχιτεκτονική της οντότητας μεταδεδομένων

Μια προσέγγιση για να αρθούν οι παραπάνω περιορισμοί είναι να δημιουργηθούν πολλαπλοί μηχανισμοί αποθήκευσης μικρής χωρητικότητας και υψηλού διαθέσιμου εύρους ζώνης κοντά στην υπολογιστική υποδομή (nearby storage). Αυτοί οι πολλαπλοί αποθηκευτικοί σχηματισμοί μπορούν συνολικά να παρέχουν ικανοποιητικό εύρος ζώνης (I/O και δίκτυο) για περιβάλλοντα eXascale.

Η δυνατές παραμετροποιήσεις που προσφέρει η οντότητα μεταδεδομένων του ΙΚΑΡΟΣ επιτρέπουν την επέκταση αυτής της λογικής στοχεύοντας στη δημιουργία nearby storage σχηματισμών με ad-hoc τρόπο on demand. Έτσι είναι δυνατόν να απομονώνονται οι πόροι που χρησιμοποιεί η εκάστοτε διεργασία και να μην επηρεάζεται η I/O απόδοση από άλλες ξένες προς αυτήν διεργασίες [42].

θα πρέπει να συνυπολογιστεί πως το περιβάλλον εργασίας μιας μελλοντικής eXascale μηχανής θα είναι εξαιρετικά πολύπλοκο και τα αιτήματα που θα δέχεται καθώς και οι συνδέσεις που θα υλοποιούνται θα έχουν εξαιρετικά μεταβαλλόμενα χαρακτηριστικά. Έτσι είναι απαραίτητο να αναπτυχθούν υβριδικές πλατφόρμες που θα προσφέρουν συνδιαχείριση της υποδομής από χρήστες και διαχειριστές, προσεγγίζοντας πιο ομαλά τα τεχνοοικονομικά ζητήματα καθώς και τα ζητήματα ασφαλείας που αναφέρθηκαν στο προηγούμενο κεφάλαιο.

Αναλύοντας περισσότερο την παραπάνω λογική γίνεται φανερό ότι θα πρέπει ταυτόχρονα με τις ως άνω προκλήσεις, σχετικά με τα I/O συστήματα, να συνυπολογιστούν και ζητήματα όπως το πως οι ανεξάρτητοι χρήστες, οι ομάδες εργασίας καθώς και οι μεγαλύτεροι σχηματισμοί παράγουν, μεταφέρουν, διαμοιράζονται και εν γένη διαχειρίζονται τα δεδομένα και κατ επέκταση τα μεταδεδομένα. Οι υπάρχουσες υποδομές δεν είναι αρκετά ευέλικτες ώστε να επιτρέπουν σε μεμονωμένους χρήστες και ομάδες να εκμεταλλεύονται πλήρως τις δυνατότητες που αυτές μπορούν να προσφέρουν.

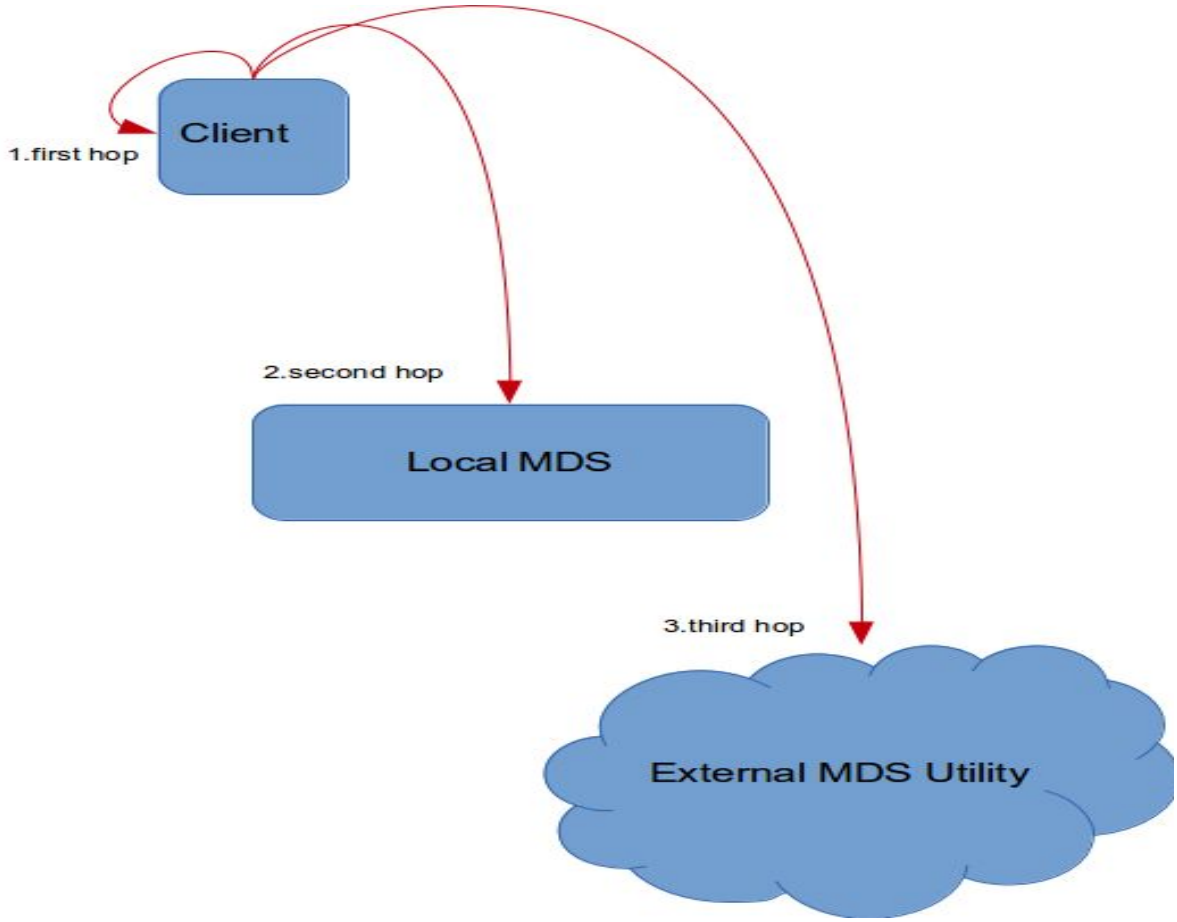
Είναι φανερό πως είναι απαραίτητο να δημιουργηθεί μια περισσότερο δυναμική λειτουργία των μηχανισμών διαχείρισης δεδομένων, αντίστοιχη με την δυναμική διαχείριση άλλων πόρων όπως οι υπολογιστικοί κύκλοι (CPU cycles). Έτσι επιλέχθηκε να δομηθεί ένα υβριδικό μοντέλο με το οποίο θα παρέχεται η δυνατότητα της πλήρους εκμετάλλευσης των υπαρχουσών υποδομών που χρησιμοποιούν τεχνολογίες Web 2.0, όπως το Facebook, και παρέχουν μεγαλύτερη ευχρηστία ενώ ταυτόχρονα διατηρούνται χαρακτηριστικά όπως η διαλειτουργικότητα, η παροχή υψηλού επιπέδου ποιότητας υπηρεσίας και η συνεργατική λογική που παρέχουν οι υποδομές πλέγματος [43].

Όπως χαρακτηριστικά αναφέρεται στο [42] και στο [48], η οντότητα των μεταδεδομένων θα δεχτεί σημαντική πίεση και μπορεί να αποτελέσει σημαντικό περιοριστικό παράγοντα στην ανάπτυξη των υποδομών νέας γενιάς. Η δημιουργία στατικών υποδομών όπου οι διαχειριστές επιλέγουν στην φάση της αρχικοποίησης της εγκατάστασης τον αριθμό των κόμβων μεταδεδομένων και I/O δεν μπορεί να λειτουργήσει σε ένα δυναμικό περιβάλλον, όπου οι ανάγκες και οι απαιτήσεις συνεχώς μεταβάλλονται. Ταυτόχρονα η χρήση πολλαπλών κόμβων μεταδεδομένων δημιουργεί τεράστια νέα ζητήματα που έχουν να κάνουν με τον συγχρονισμό και την ενημέρωση τους καθώς και με την αποσύνδεση των δεδομένων από τα μεταδεδομένα στοχεύοντας πάντα στον αποδοτικό εντοπισμό των δεδομένων.

Η αρχιτεκτονική του ΙΚΑΡΟΣ δίνει την δυνατότητα δημιουργίας υβριδικών υποδομών με τη χρήση εξωτερικών μηχανισμών και εφαρμογών που παρέχουν μεγαλύτερη ευελιξία και κλιμάκωση των υποσυστημάτων της κάθε υποδομής ανεξάρτητα ανάλογα με τις

ανάγκες. Το ΙΚΑΡΟΣ είναι μια ανοιχτή αρχιτεκτονική που μπορεί να εκμεταλλευτεί τις οικονομίες κλίμακας για να δομήσει περισσότερο αποδοτικές υποδομές.

Η οντότητα μεταδεδομένων του ΙΚΑΡΟΣ ακολουθεί μια προοδευτική/ιεραρχική λογική. Σε περίπτωση αιτήματος η υπηρεσία αρχικά αναζητά την απάντηση εσωτερικά στον ίδιο τον αιτούντα. Μπορεί να είναι ο ίδιος που σε προγενέστερο χρόνο να έχει δημιουργήσει την πληροφορία, άρα βάση της αρχιτεκτονικής του ΙΚΑΡΟΣ θα την αποθηκεύσει στους εσωτερικούς του μηχανισμούς, ή μπορεί να την έχει λάβει από την περιοδική ενημέρωση των συστημάτων. Σε περίπτωση που η απάντηση δεν βρεθεί εσωτερικά το επόμενο βήμα είναι να ερωτηθεί κάποιος MDS σε τοπικό επίπεδο.

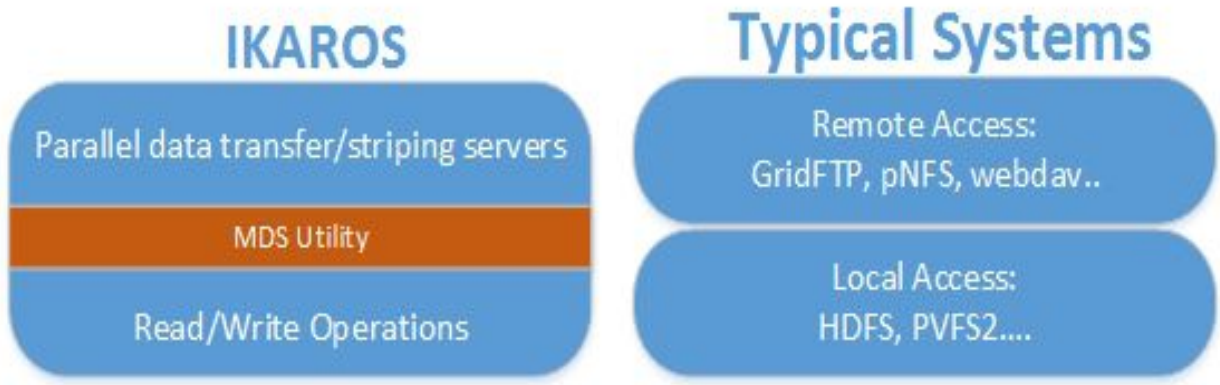


Σχήμα 5:Ιεραρχία οντότητας μεταδεδομένων ΙΚΑΡΟΣ

Αν και αυτή η προσπάθεια είναι ανεπιτυχής τότε θα αναζητήσει την απάντηση σε κάποιο εξωτερικό μηχανισμό που μπορεί να δρα ως μια γενικότερη οντότητα μεταδεδομένων και να έχει εικόνα για όλες τις ανεξάρτητες ιεραρχίες της υποδομής. Η συγκεκριμένη ιεράρχηση (σχήμα 5) δεν είναι υποχρεωτική και μπορεί να μεταβάλλεται δυναμικά ανάλογα με τις απαιτήσεις της εφαρμογής ή των χρηστών.

Η οντότητα μεταδεδομένων δίνει την δυνατότητα να ακολουθείται κοινή μεθοδολογία και αντιμετώπιση ανεξάρτητα από τα διαφορετικά επίπεδα στα οποία ενεργεί το ΙΚΑΡΟΣ. Με αυτόν τον τρόπο αποφεύγεται η απομονωμένη λογική ανάπτυξης που ακολουθούν τα άλλα συστήματα, διατηρώντας παράλληλα την αυτονομία υλοποίησης των υπηρεσιών του κάθε επιπέδου. Αρκεί η κατάλληλη παραμετροποίηση της οντότητας των μεταδεδομένων για να να διαχωριστούν οι λειτουργίες και να υπάρχει μετάβαση

από το ένα επίπεδο στο άλλο, μεταξύ τοπικής και απομακρυσμένης πρόσβασης (σχήμα 6).



Σχήμα 6: Επίπεδα λειτουργιών ΙΚΑΡΟΣ σε σύγκριση με ένα τυπικό σύστημα

Η οντότητα μεταδεδομένων παρέχει δύο βασικές πληροφορίες, τον αριθμό των διαθέσιμων I/O κόμβων ή εξυπηρετητών αποθήκευσης (ανάλογα με το σε ποιο περιβάλλον γίνεται αναφορά) κατά τις λειτουργίες εγγραφής (εικόνα 1) και την κατανομή του αρχείου, στους κόμβους αποθήκευσης, κατά τις λειτουργίες ανάγνωσης (εικόνα 2). Το ΙΚΑΡΟΣ διαμορφώνει τα μεταδεδομένα σε JavaScript Object Notation (JSON) μορφή.

{"io_total":10,"io_urls":[{"url":"compute-0-0:8000"},{"url":"compute-0-1:8000"},
{"url":"compute-0-2:8000"},{"url":"compute-0-3:8000"},{"url":"compute-0-4:8000"},
{"url":"compute-0-5:8000"},{"url":"compute-0-6:8000"},{"url":"compute-0-8:8000"},
{"url":"compute-0-9:8000"},{"url":"compute-0-10:8000"}]}

Εικόνα 1: Διαθέσιμοι κόμβοι

Η αρχιτεκτονική του ΙΚΑΡΟΣ και πιο συγκεκριμένα η αρχιτεκτονική που ακολουθεί η οντότητα των μεταδεδομένων επιτρέπει την δυναμική επιλογή του αριθμού των I/O κόμβων που θα χρησιμοποιηθούν σε κάθε διεργασία. Σε αντίθεση με το PVFS2 που πάντοτε χρησιμοποιεί όλους τους διαθέσιμους κόμβους. Με αυτόν τον τρόπο γίνεται εφικτό να δομηθούν ad-hoc nearby storage σχηματισμοί on demand. Έτσι αποφεύγονται φαινόμενα συμφόρησης, “παρεμβολής” της μίας διεργασίας από την εκτέλεση της άλλης και ταυτόχρονη χρήση των πόρων. Στην παρούσα φάση αυτή η λειτουργία επιτυγχάνεται με την επέμβαση του χρήστη καθώς δεν έχει υλοποιηθεί κάποια αυτοματοποιημένη διαδικασία. Τα οφέλη της συγκεκριμένης λειτουργίας αναλύονται διεξοδικά στο κεφάλαιο 7.

Επιπλέον, αν πρέπει να προστεθούν ή να αφαιρεθούν κόμβοι στην όλη υποδομή αυτό μπορεί να επιτευχθεί με την ίδια διαδικασία, on demand. Ο νέος κόμβος στην φάση της αρχικοποίησης του ενημερώνει το σύστημα μεταδεδομένων ότι είναι ενεργός και διαθέσιμος. Στην συνέχεια είναι στην διακριτική ευχέρεια του χρήστη ή της εφαρμογής να τον χρησιμοποιήσει. Αντίθετα στο PVFS2 για να συμβεί κάτι αντίστοιχο θα πρέπει να

διαγραφούν οι παρούσες ρυθμίσεις και να αρχικοποιηθεί ξανά η υποδομή, κάτι που προϋποθέτει συμμετοχή των διαχειριστών της υποδομής αλλά και απώλεια των δεδομένων.

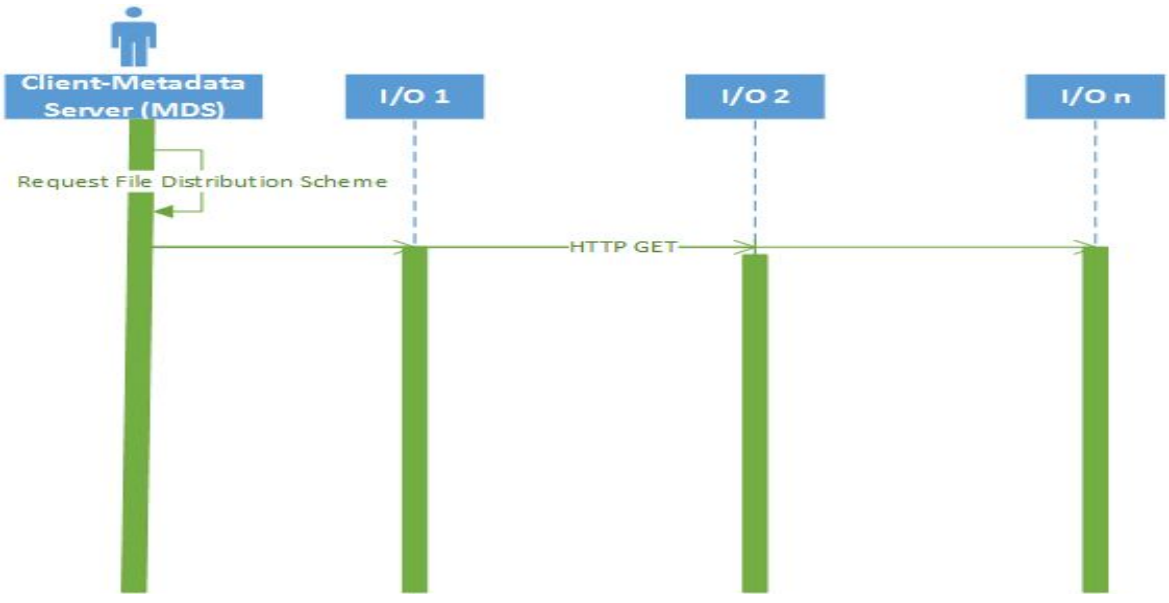
{"file_size":2752577938,"timestamp":1355301777,"io_total":10,"schema":1,"io_urls":
[{"part":"0","url":"compute-0-0:8000","start":"0","end":"275257793"},
{"part":"1","url":"compute-0-1:8000","start":"275257794","end":"550515586"},
{"part":"2","url":"compute-0-2:8000","start":"550515587","end":"825773379"},
{"part":"3","url":"compute-0-3:8000","start":"825773380","end":"1101031172"},
{"part":"4","url":"compute-0-4:8000","start":"1101031173","end":"1376288965"},
{"part":"5","url":"compute-0-5:8000","start":"1376288966","end":"1651546758"},
{"part":"6","url":"compute-0-6:8000","start":"1651546759","end":"1926804551"},
{"part":"7","url":"compute-0-8:8000","start":"1926804552","end":"2202062344"},
{"part":"8","url":"compute-0-9:8000","start":"2202062345","end":"2477320137"},
{"part":"9","url":"compute-0-10:8000","start":"2477320138","end":"2752577938"}]}

Εικόνα 2: Κατανομή του αρχείου στους κόμβους

4.2 Περιπτώσεις χρήσης της οντότητας μεταδεδομένων

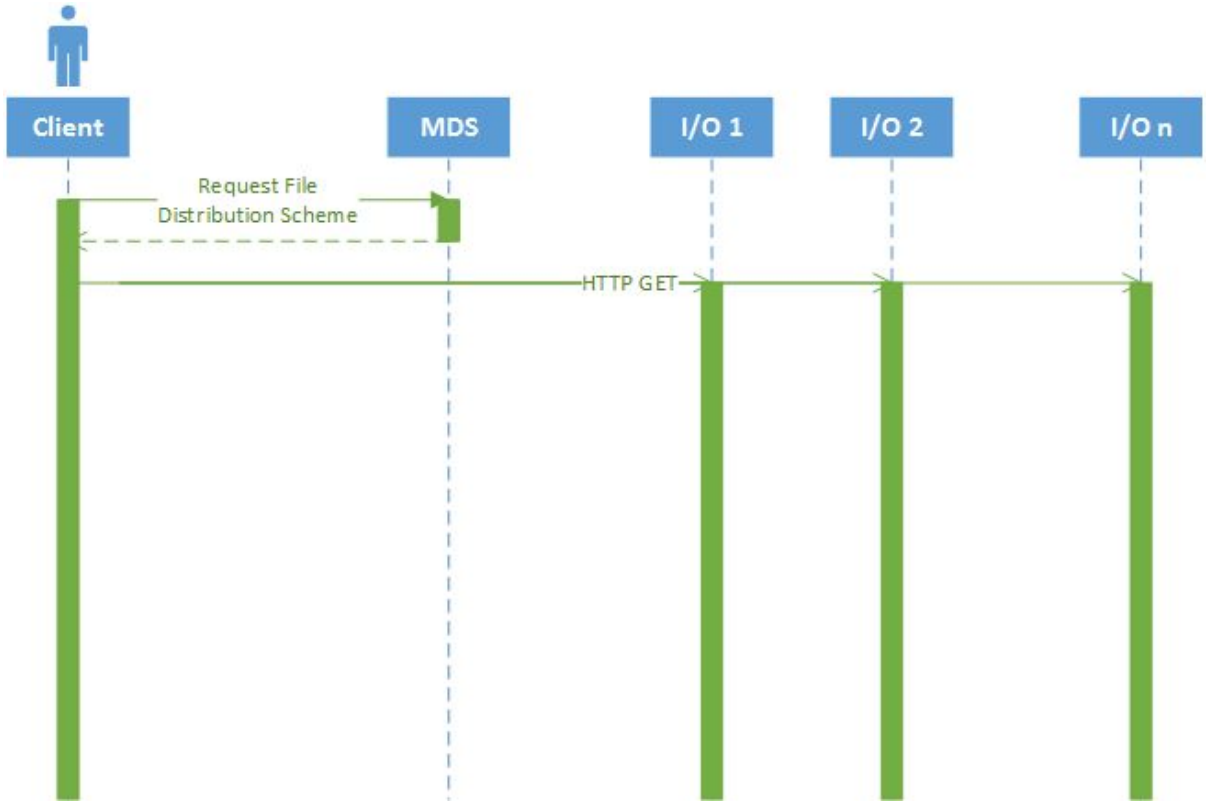
Το ΙΚΑΡΟΣ παράγει μεταδεδομένα σε μορφή JSON και διατηρεί ένα αντίγραφο στους κόμβους τύπου “μεταδεδομένων”. Σύμφωνα με την αρχιτεκτονική του ΙΚΑΡΟΣ [24] κάθε κόμβος της υποδομής θεωρείται ομότιμος και μπορεί να ενεργεί ταυτόχρονα ως “πελάτης”, ως κόμβος εισόδου/εξόδου και ως κόμβος “μεταδεδομένων”. Επιπροσθέτως, τα μεταδεδομένα μπορεί να διανέμονται σε εξωτερικές οντότητας όπως σε υπηρεσίες κοινωνικής δικτύωσης, στην προκειμένη περίπτωση στο Facebook. Η μορφοποίηση των μεταδεδομένων του ΙΚΑΡΟΣ σε μορφή JSON καθώς και η χρήση τεχνικών που είναι συμβατές με τις web 2.0 τεχνολογίες επιτρέπουν τη δημιουργία τέτοιων υβριδικών υποδομών. Οι χρήσεις της οντότητας μεταδεδομένων είναι τρεις και κατηγοριοποιούνται ανάλογα με την κλίμακα υλοποίησης της εφαρμογής.

1. *Αυτόνομη χρήση (σχήμα 7)*, ο κόμβος τύπου πελάτη αποθηκεύει όλα τα μεταδεδομένα και λειτουργεί και ως MDS.



Σχήμα 7:Αυτόνομη χρήση

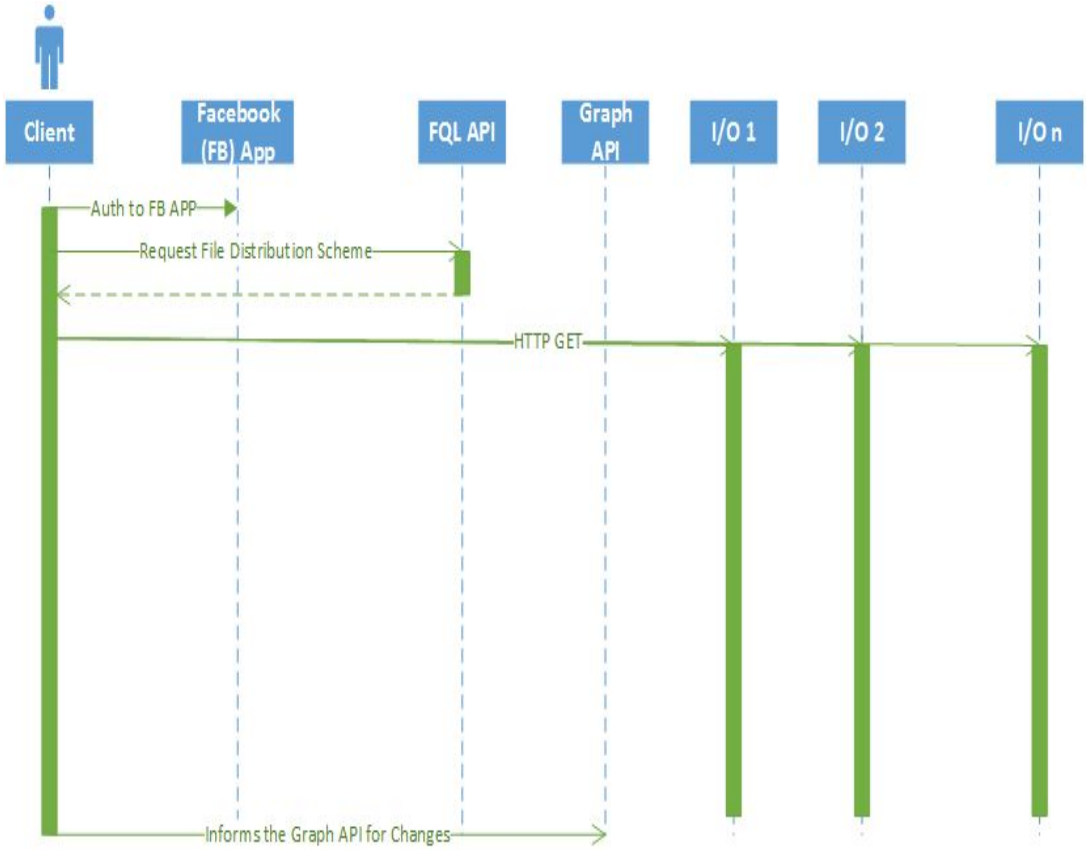
2. *Τοπική χρήση (σχήμα 8),* η λειτουργία τού ΙΚΑΡΟΣ ενσωματώνει εξωτερικούς κόμβους μεταδεδομένων οι οποίοι συνήθως τοποθετούνται στο τοπικό δίκτυο .



Σχήμα 8:ΙΚΑΡΟΣ MDS, Τοπική χρήση

3. *Υβριδική χρήση (σχήμα 9)*, η λειτουργία του ΙΚΑΡΟΣ χρησιμοποιεί όλες τις διαθέσιμες δυνατότητες σχετικά με τα μεταδεδομένα. Αυτό περιλαμβάνει και εξωτερικές υποδομές με την χρήση των οποίων μπορούν να δομηθούν πολύπλοκα σχήματα. Στην υβριδική χρήση το ΙΚΑΡΟΣ ενσωματώνει στην λειτουργία του υπάρχουσες υπηρεσίες κοινωνικής δικτύωσης, όπως το Facebook. Έτσι είναι εφικτό με δυναμικό τρόπο να υλοποιηθεί η διαχείριση, ο διαμοιρασμός και η δημοσίευση των μεταδεδομένα και κατ' επέκταση των δεδομένα. Καθώς τα πρώτα περιέχουν πληροφορίες σχετικά με την τοποθεσία των δεδομένων. Με αυτόν τον τρόπο δεν χρειάζεται να δημιουργηθούν από την αρχή δομές για την αναζήτηση αλλά και την γενικότερη διαχείριση των μεταδεδομένων. Ταυτόχρονα επιτρέπεται στους χρήστες να έχουν περισσότερο ενεργό ρόλο, παρέχοντας τους πραγματική διαχειριστική ισχύ στα δεδομένα που παράγουν. Αυτή η λειτουργία προσδίδει στην υποδομή σημαντικά στοιχεία που ακολουθούν τις έννοιες που αναλύθηκαν στο προηγούμενο κεφάλαιο. Κατ' αυτόν τον τρόπο επιτυγχάνεται η δημιουργία πιο αποδοτικών υποδομών nearby αποθήκευσης. Θα πρέπει να τονισθεί ότι το ΙΚΑΡΟΣ δεν περιορίζεται και δεν εξαρτάται από το Facebook καθώς η μορφοποίηση των μεταδεδομένα σε JSON είναι εξαιρετικά κοινή σε τεχνολογίες Web 2.0 και έτσι υπάρχει η δυνατότητα σύνδεσης του με πολυάριθμες υπάρχουσες εξωτερικές υποδομές. Το ΙΚΑΡΟΣ,

με αυτόν τον τρόπο, καταφέρνει να κλιμακώνει ανεξάρτητα τις λειτουργίες των δεδομένων με τις αντίστοιχες των μεταδεδομένων, κάτι που είναι εξαιρετικά χρήσιμο για την ανάπτυξη των eXascale συστημάτων.



Σχήμα 9:ΙΚΑΡΟΣ MDS, υβριδική χρήση

Στο σχήμα 9 παρουσιάζονται οι αλληλεπιδράσεις μεταξύ του ΙΚΑΡΟΣ και του Facebook, ως παράδειγμα χρήσης υφιστάμενης υποδομής που δεν έχει σχεδιαστεί από το ΙΚΑΡΟΣ. Αυτό επιτυγχάνεται με τη χρήση διαδεδομένων προτύπων και πρωτοκόλλων όπως το HTTP και το JSON. Για τη συγκεκριμένη υλοποίηση το ΙΚΑΡΟΣ αλληλεπιδρά με το Facebook Graph API και με το Facebook Query Language (FQL) API [44].

Για να επιτύχει κάτι τέτοιο θα πρέπει πρώτα να έχει αυθεντικοποιηθεί και πιστοποιηθεί στο Facebook μέσω κάποιας ενεργής εφαρμογής (Facebook App). Επιπλέον, ο εκάστοτε χρήστης θα πρέπει να επιτρέπει στην εφαρμογή να έχει πρόσβαση στα ”Extended Permissions”: ”create note”, ”user notes” and ”friends notes”. Στην συνέχεια μπορεί να δημοσιοποιήσει, σε όσους επιθυμεί, και να διαμοιράσει τα μεταδεδομένα του ως “notes” στο προφίλ του με την βοήθεια του Graph API αλλά και να τα αναζητήσουμε μέσω του FQL API. Αυτές οι δύο ενέργειες ακολουθούν την ροή διεργασίας της υπηρεσίας ΙΚΑΡΟΣ [24] και καλούνται στα στάδια όπου απαιτείται κάποια αλληλεπίδραση με την οντότητα των μεταδεδομένα.

Οι μηχανισμοί μεταδεδομένων του ΙΚΑΡΟΣ έχουν κυρίαρχο ρόλο στην όλη αρχιτεκτονική. Είναι αυτοί που επιτρέπουν τη δημιουργία κοινών μεθόδων αντιμετώπισής των προβλημάτων στην γενικότερη ροή των δεδομένων, διασφαλίζοντας παράλληλα την αυτονομία υλοποίησης των επιμέρους υπηρεσιών. Έτσι υλοποιείται ένας από τους βασικούς στόχους του ΙΚΑΡΟΣ που είναι η άρση των πολλαπλών επιπέδων συγχρονισμού στην συνολική ροή των δεδομένων από ένα δίκτυο ευρείας περιοχής προς ένα τοπικό δίκτυο και αντίστροφα.

5. ΕΙΣΟΔΟΣ/ΕΞΟΔΟΣ ΣΤΟ ΙΚΑΡΟΣ

Το ΙΚΑΡΟΣ έχει δομηθεί ως ένα “λεπτό” στρώμα (thin layer) που έχει την δυνατότητα να προσφέρει υπηρεσίες σε πολλαπλά επίπεδα. Σε τεχνικό επίπεδο, επιτρέπει την απευθείας πρόσβαση σε κάθε έναν αποθηκευτικό κόμβο (I/O κόμβο) από οποιαδήποτε επίπεδο και αν ενεργεί. Με αυτόν τον τρόπο επιτυγχάνεται η ροή των δεδομένων με χρήση παράλληλων τεχνικών σε όλη την διαδρομή, χωρίς να απαιτούνται ενδιάμεσα στάδια συγχρονισμού και αναδιοργάνωσης, όπως συμβαίνει στα όλα συστήματα [81].

5.1 Μηχανισμοί εισόδου/εξόδου του ΙΚΑΡΟΣ

Οι μηχανισμοί εισόδου/εξόδου και η λογική τους αποτελούν το βασικό μηχανισμό μεταφοράς του ΙΚΑΡΟΣ που με την σειρά τους δομούν τις πρωταρχικές λειτουργίες με τις οποίες αλληλεπιδρά με το αποθηκευτικό σύστημα και το σύστημα αρχείων.

Η σύνταξη των αιτημάτων προς το ΙΚΑΡΟΣ ακολουθεί την μορφή του uniform resource identifier (URI):

< scheme name >:< hierarchical part > [? < query >]

Για παράδειγμα:

http : //hostname : port/ikaros?case&n2&n3&n4&n5&n6

Όπου:

1. “case” είναι ο επιλογέας της λειτουργίας.
2. “n2” είναι το ζητούμενο seek point στο αρχείο.
3. “n3” είναι το ζητούμενο μέγεθος για το buffer.
4. “n4” είναι το ζητούμενο μέγεθος του μέρους του αρχείου (chunk size).
5. “n5” είναι ο ζητούμενος αριθμός των παράλληλων καναλιών μεταφοράς.
6. “n6” είναι το ζητούμενο αρχείο.

Στην συνέχεια παρουσιάζονται πιο αναλυτικά οι βασικές λειτουργίες χρήσης του ΙΚΑΡΟΣ. Στις παρακάτω λειτουργίες φαίνεται η αλληλεπίδραση με την τοπική οντότητα μεταδεδομένων αλλά κυρίως με τους I/O κόμβους, κάτι που σε αυτήν την ενότητα αναλύεται σε βάθος. Γιαυτό έχουν επιλεγεί παραδείγματα που επικεντρώνονται σε αυτές τις αλληλεπιδράσεις.

Μηχανισμοί ανάγνωσης (σχήμα 10):

1. Ο πελάτης στέλνει ένα αίτημα με την ακόλουθη μορφή:

http : //hostname : port/ikaros?2&0&65536&2048&0&datafile

επιλέγει δηλαδή την περίπτωση 2 (case 2), της οποίας η ροή διεργασίας θα αναλυθεί στην συνέχεια, με σημείο έναρξης το 0, συνολικό μέγεθος 65536, με chunk size 2048 και κανένα παράλληλο κανάλι μεταφοράς.

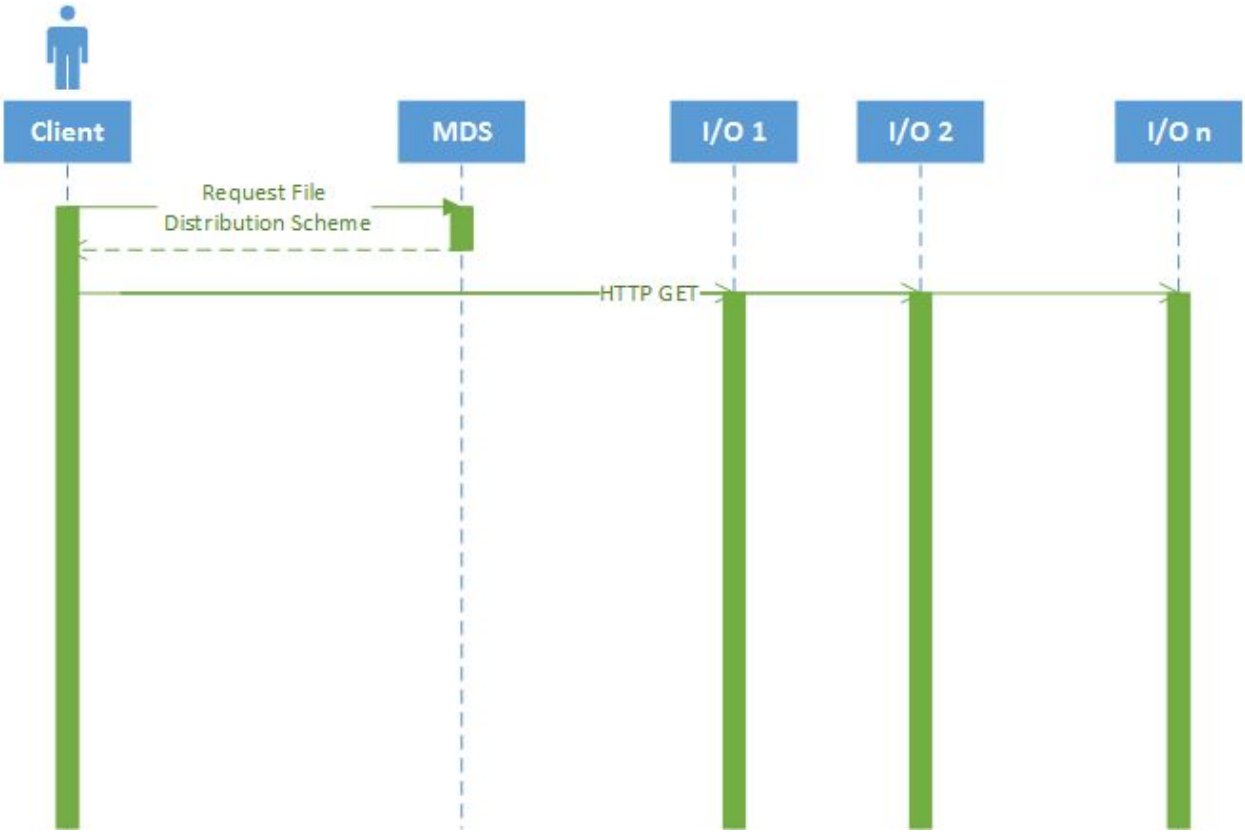
2. Το ΙΚΑΡΟΣ στέλνει ένα αίτημα προς την οντότητα των μεταδεδομένων ρωτώντας σχετικά με την κατανομή του ζητούμενου αρχείου.
3. Η οντότητα των μεταδεδομένων αποκρίνεται και στέλνει την κατανομή.

4. Ο πελάτης στέλνει HTTP GET αιτήματα προς κάθε κόμβο εισόδου/εξόδου, με την ακόλουθη μορφή:

   http : //nas03.domain.org/PATH/datafilepart1

   http : //nas04.local/PATH/datafilepart2

5. Τελικά, ο πελάτης αντιγράφει τον κάθε HTTP buffer/stream στο αντίστοιχο σημείο του αρχείου.



Σχήμα 10: Μηχανισμοί Ανάγνωσής

Μηχανισμοί εγγραφής (σχήμα 11):

1. Ο πελάτης στέλνει ένα αίτημα με την ακόλουθη δομή:

   http : //hostname : port/ikaros?4&0&65536&2048&0&datafile

επιλέγει δηλαδή την περίπτωση 4 (case 4), της οποίας η ροή διεργασίας θα αναλυθεί στην συνέχεια, με σημείο έναρξης το 0, συνολικό μέγεθος 65536, με chunk size 2048 και κανένα παράλληλο κανάλι μεταφοράς.

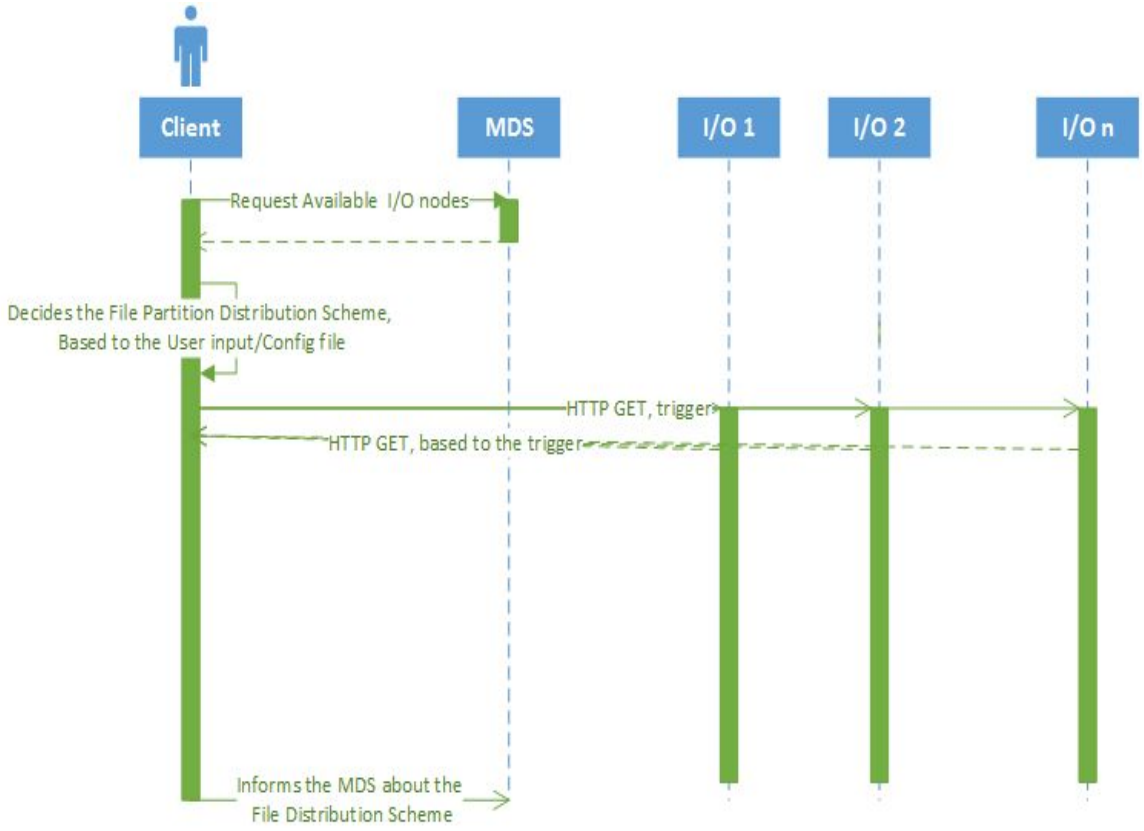
2. Στην συνέχεια, το ΙΚΑΡΟΣ ρωτάει την οντότητα των μεταδεδομένων σχετικά με τους διαθέσιμους κόμβους εισόδου/εξόδου. Η προεπιλογή είναι να χρησιμοποιεί όλους τους κόμβους. Υπάρχει όμως η δυνατότητα αυτό να αλλάξει ανάλογα με τις απαιτήσεις του χρήστη. Αυτή η επιλογή αποτελεί ένα από τα σημαντικότερα πλεονεκτήματα του ΙΚΑΡΟΣ αφού επιτρέπει την δημιουργία αποθηκευτικών σχηματισμών, με δυναμικό τρόπο και on demand. Έτσι είναι εφικτό να οργανωθούν κατάλληλα οι I/O πόροι, που χρησιμοποιεί μια διεργασία, ώστε να

μην επηρεάζονται από άλλες ξένες προς αυτή διεργασίες και τελικά να επιτευχθεί η μέγιστη δυνατή I/O απόδοση.

3. Η οντότητα των μεταδεδομένων αποκρίνεται στέλνοντας τον συνολικό αριθμό των διαθέσιμων κόμβων εισόδου/εξόδου, για την συγκεκριμένη συνδιαλλαγή και τον όνομα του κάθε ενός (π.χ http://nas03.domain.org/, http://nas04.local/, ...)
4. Ο πελάτης εξάγει το μέγεθος του αρχείου ή του chunk και αποφασίζει την κατανομή του στους διαθέσιμος κόμβους εισόδου/εξόδου.
5. Ο πελάτης “σκανδαλίζει” κάθε κόμβο εισόδου/εξόδου ο οποίος με την σειρά του στέλνει ένα HTTP αίτημα προς τον πελάτη ζητώντας το “κομμάτι” που του αναλογεί (reverse read). Το αίτημα προς τον πελάτη ακολουθεί την παρακάτω μορφή:

   http : //ui − client/ikaros?1&1&65536&2048&0&datafile

   http : //ui − client/ikaros?1&65536&131072&2048&0&datafile
6. Ο πελάτης ενημερώνει την οντότητα μεταδεδομένων σχετικά με την κατανομή του αρχείου στους κόμβους εισόδου/εξόδου.



Σχήμα 11:Μηχανισμοί εγγραφής, ΙΚΑΡΟΣ

Η αρχιτεκτονική του ΙΚΑΡΟΣ και πιο συγκεκριμένα η αρχιτεκτονική που ακολουθεί η οντότητα των μεταδεδομένων επιτρέπει τη δυναμική επιλογή του αριθμού των I/O κόμβων που θα χρησιμοποιηθούν σε κάθε διεργασία. Σε αντίθεση με το PVFS2 που πάντοτε χρησιμοποιεί όλους τους διαθέσιμους κόμβους.

5.2 Παραμετροποίηση των αιτημάτων

Με την χρήση των συγκεκριμένων παραμέτρων είναι εφικτό να δομηθούν εξαιρετικά πολύπλοκα σχήματα που με την σειρά τους επιτρέπουν να υλοποιηθούν λειτουργίες όπως το partitioning του αρχείου, η δημιουργία παράλληλων καναλιών μεταφοράς και η επιλογή τοποθέτησης του αρχείου σε πολλαπλούς αποθηκευτικούς εξυπηρετητές (striping). Οι παράμετροι έχουν ως ακολούθως:

1. file size (fs): το συνολικό μέγεθος του αρχείου.
2. shares (sh): ο συνολικός αριθμός των κόμβων εισόδου/εξόδου, που είναι διαθέσιμοι για την συγκεκριμένη μεταφορά.
3. partition size (ps): ο αριθμός των bytes που θα αποθηκευτούν σε κάθε partition.
4. partition number (pn): ο αύξων αριθμός του partition.
5. file remainder (fr): ο αριθμός των bytes που παραμένουν αφαιρώντας το γινόμενο του (ps) με το (sh) από το συνολικό μέγεθος του αρχείου.
6. partition location (pl): το offset από την αρχή του αρχείου μέχρι την αρχή του partition.
7. requested range (rr): το byte range για μια την εξαγωγή μίας τυχαίας λωρίδας.
8. requested range, start point (rs): το αρχικό σημείο του ζητούμενου διαστήματος.
9. requested range, end point (re): το τελικό σημείο του ζητούμενου διαστήματος.
10. number of the first partition (pnf): ο αύξων αριθμός του πρώτου partition του ζητούμενου διαστήματος.
11. number of the last partition (pnl): ο αύξων αριθμός του τελικού partition του ζητούμενου διαστήματος.

Στις εξισώσεις 1 και 2 απεικονίζεται η τοποθεσία ενός αρχείου στα αντίστοιχα partition:

$$ps = [fs/sh] \tag{1}$$

$$pl = (pn - 1) * ps \tag{2}$$

Για να ζητηθεί ένα τυχαίο stripe του αρχείου θα πρέπει να εκτελεστούν οι εξισώσεις 3-6.

$$pnf = [rs/ps] + 1 \tag{3}$$

$$pnl = [re/ps] + 1 \tag{4}$$

$$fr = fs - ps * sh \tag{5}$$

$$if : pn = sh, (then), ps = ps + fr, (else), ps = ps \tag{6}$$

Το ΙΚΑΡΟΣ ανασύρει ένα τυχαίο stripe ζητώντας από κάθε partition τα ακόλουθα byte (εξίσωση 7):

1. (pnf), από το ((pnf -1)* ps-rs) έως το τέλος του partition

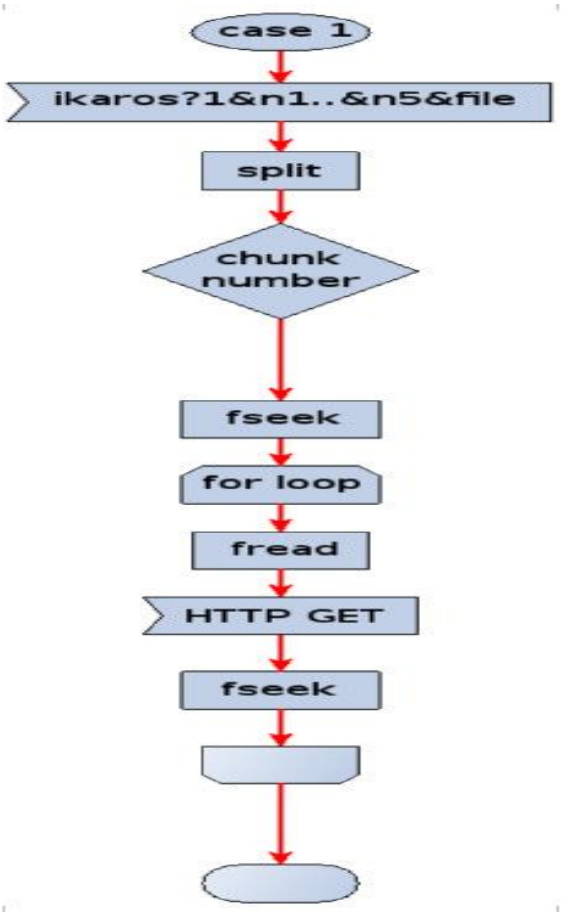
2. Από όλα τα partition το range: [(pnf+1) έως (pnl-1])], από το αρχικό σημείο μέχρι το τελικό σημείο του partition.
3. (pnl), από τον αρχικό σημείο του partition έως το ((pnl -1)* ps-re).

$$rr = ((pnf - 1) * ps - rs) + \sum_{pn=pnf+1}^{pnl-1} ps_pn + ((pnl - 1) * ps - re) \qquad (7)$$

5.3 Ροή διεργασίας των λειτουργιών του IKAROΣ Apache module

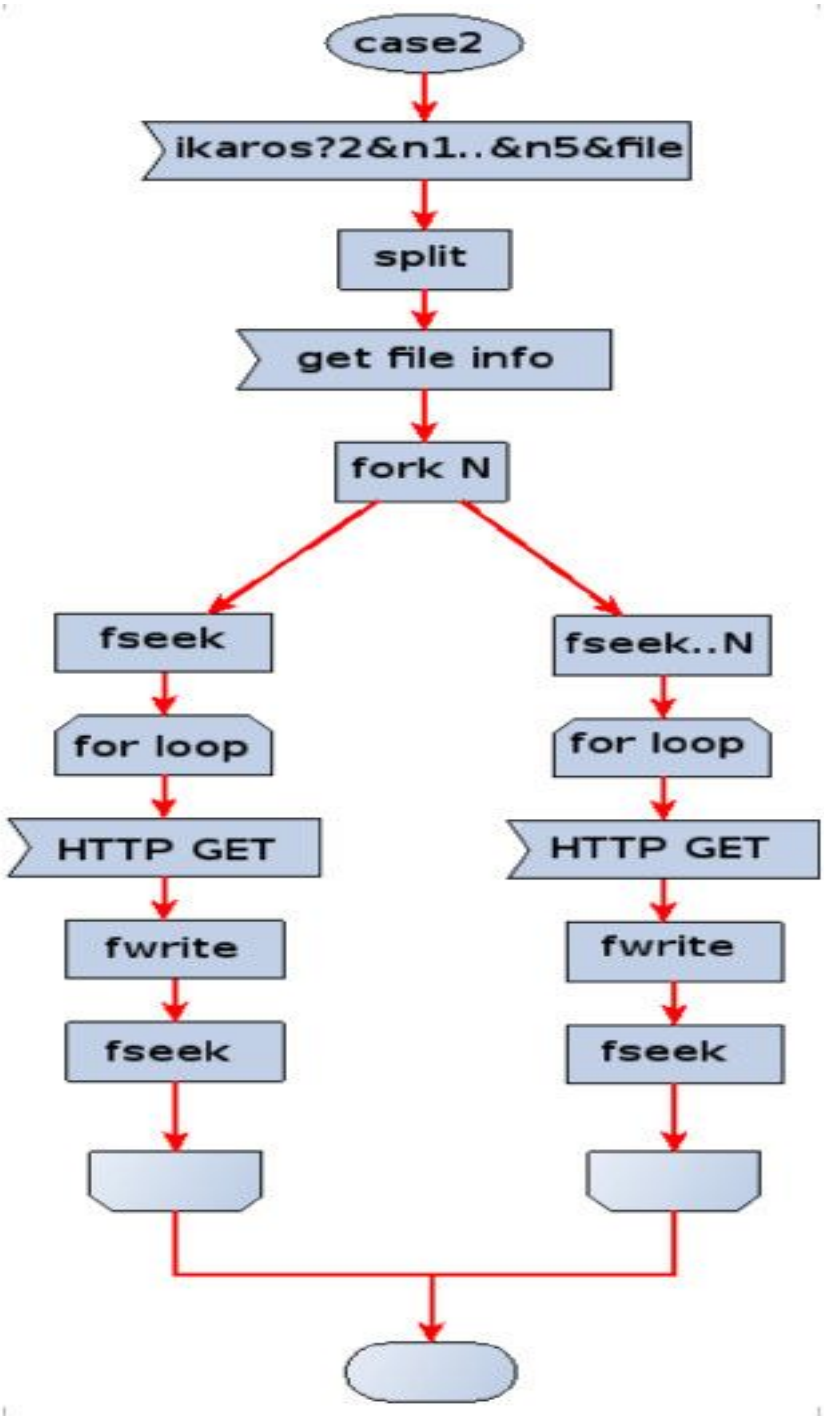
Το IKAROΣ Apache module διαθέτει τις ακόλουθες λειτουργίες:

1. (case 1) Ανασύρει το ζητούμενο “τμήμα” του αρχείου (σχήμα 12). Η συγκεκριμένη λειτουργία δρα επιβοηθητικά της περίπτωσης 4 (διεργασίες εγγραφής - reversed read). Αυτή η λειτουργία, μπορεί να προσομοιωθεί με την τεχνική Range Header του HTTP καθώς, εν μέρει, χρησιμοποιείται για να δομηθεί η όλη λειτουργία. Για τις ακόλουθες σχηματικές παραστάσεις ακολουθείται η σημασιολογία των λογικών διαγραμμάτων όπως παρουσιάζονται στο [77].



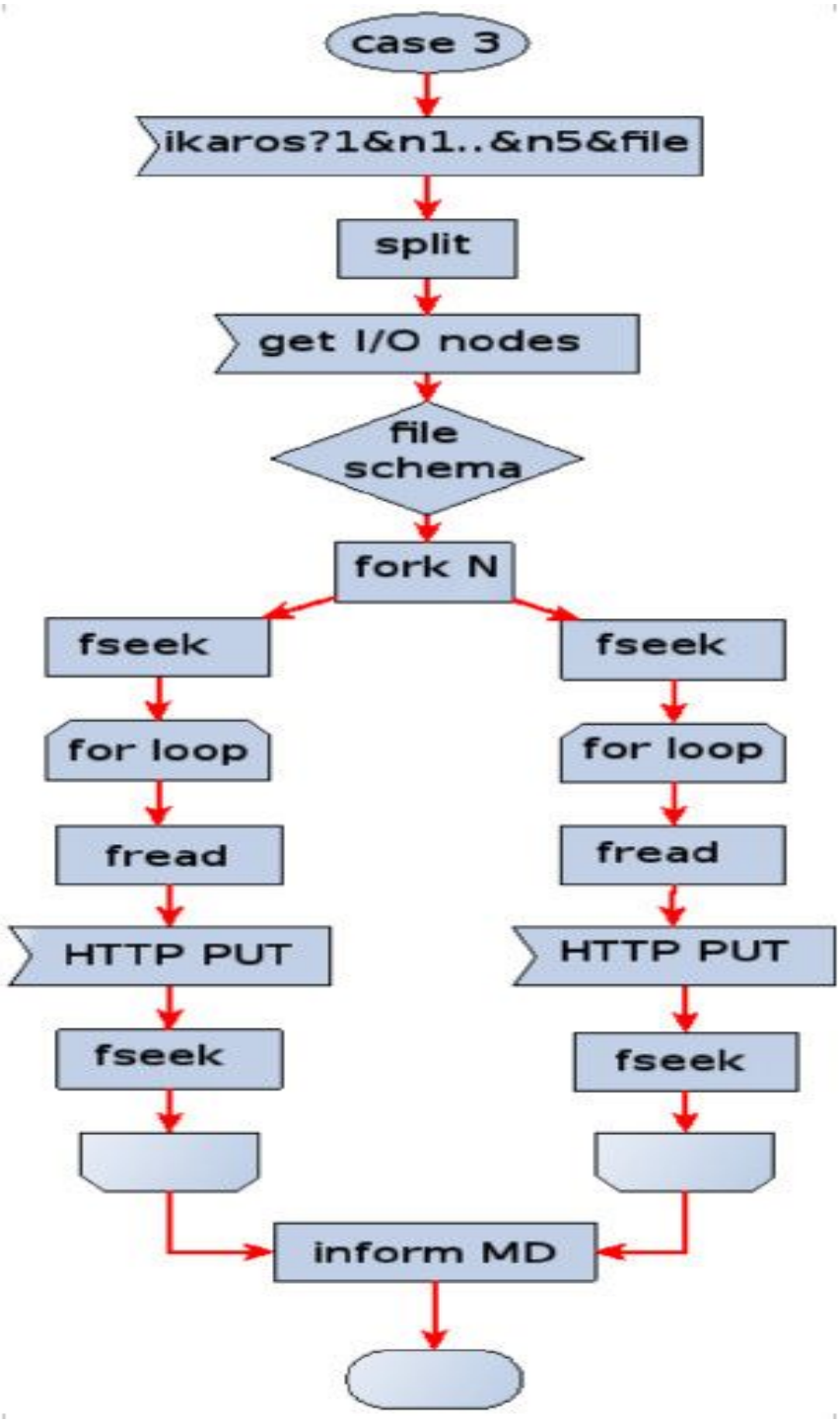
Σχήμα 12:IKAROΣ module, διάγραμμα ροής περίπτωση 1, (ανάσυρση τμήματος αρχείου)

2. (case 2) Ανασύρει το ζητούμενο αρχείο στο σύνολο του, τα μέρη του αρχείου βρίσκονται στους κόμβους εισόδου/εξόδου (διεργασία ανάγνωσης) (σχήμα 13). Αρχικά οι μηχανισμοί εισόδου/εξόδου επικοινωνούν με την οντότητα μεταδεδομένων ώστε να ανασύρουν την κατανομή του αρχείου στους αποθηκευτικούς κόμβους. Στην συνέχεια εκτελούνται κοινά HTTP GET αιτήματα προς τους εμπλεκόμενους αποθηκευτικούς κόμβους.



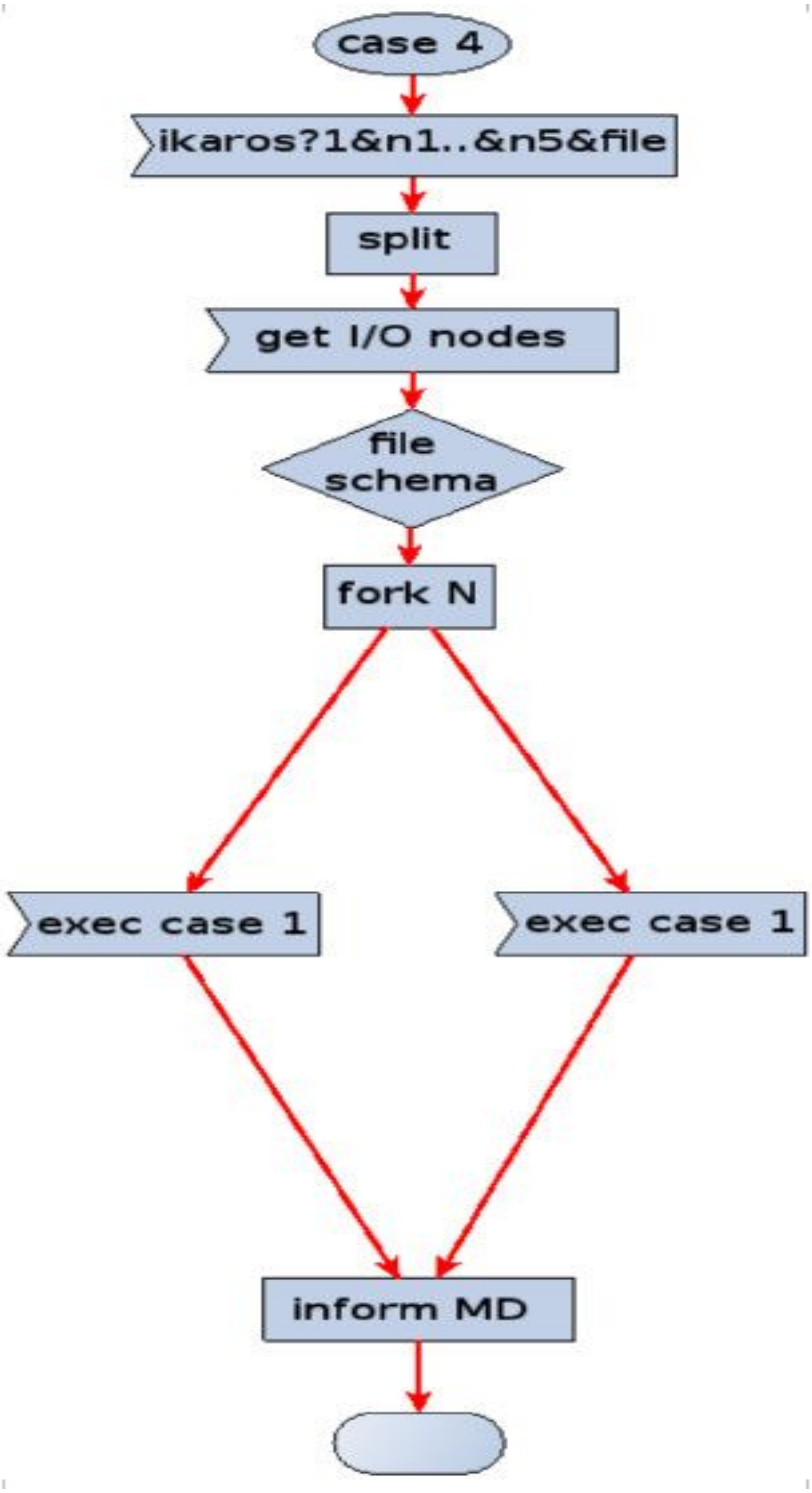
Σχήμα 13: ΙΚΑΡΟΣ module, διάγραμμα ροής περίπτωση 2, (διεργασία ανάγνωσης)

3. (case 3) Αποθήκευση αρχείου στους κόμβους εισόδου/εξόδου με χρήση της HTTP PUT (διεργασίες εγγραφής) (σχήμα 14). Αρχικά οι μηχανισμοί εισόδου/εξόδου επικοινωνούν με την οντότητα μεταδεδομένων ώστε να ενημερωθούν σχετικά με τους διαθέσιμους αποθηκευτικούς κόμβους. Στην συνέχεια εκτελούν HTTP PUT αιτήματα προς τους εμπλεκόμενους αποθηκευτικούς κόμβους.



Σχήμα 14:ΙΚΑΡΟΣ module, διάγραμμα ροής περίπτωση 3, (διεργασίες εγγραφής)

4. (case 4) Αποθήκευση αρχείου στους κόμβους εισόδου/εξόδου με χρήση της HTTP GET (διεργασίες εγγραφής). Με αυτήν την μέθοδο υλοποιούνται πιο πολύπλοκες δομές που επιτρέπουν μεγάλη ευελιξία σε πολλαπλά επίπεδα (σχήμα 15).



Σχήμα 15: ΙΚΑΡΟΣ module, διάγραμμα ροής περίπτωση 4, (διεργασίες εγγραφής- reversed read)

Η διεργασία κατανέμει το αρχείο σε κομμάτια (chunks) σύμφωνα με την λογική που παρουσιάζεται στην παράγραφο 5.2. Θα πρέπει πρώτα να υπάρξει ενημέρωση από την οντότητα μεταδεδομένων σχετικά με τους διαθέσιμους αποθηκευτικούς κόμβους εισόδου/εξόδου ώστε αποφασιστεί το πως θα κατανεμηθούν τα κομμάτια του αρχείου στους διαθέσιμους αποθηκευτικούς κόμβους, ανάλογα με το αν υπάρχει κάποιο συγκεκριμένο αίτημα από τον χρήστη ή όχι (σχήμα 11). Τέλος εκτελείται η προαναφερθείσα περίπτωση 1 (case 1) τόσες φορές όσες και τα κομμάτια στα οποία έχει κατανεμηθεί το αρχείο.

5. Ανάσυρση του ζητούμενου αρχείου, τα μέρη του αρχείου βρίσκονται στους κόμβους εισόδου/εξόδου. Οι κόμβοι εισόδου/εξόδου είναι GridFTP εξυπηρετητές. Στο σχήμα 13, απλά, αντικαθίστανται τα HTTP GET αιτήματα με αιτήματα συμβατά με το GridFTP.
6. Αποθήκευση αρχείου στους κόμβους εισόδου/εξόδου, Οι κόμβοι εισόδου/εξόδου είναι GridFTP εξυπηρετητές. Στο σχήμα 14, απλά, αντικαθίστανται τα HTTP GET αιτήματα με αιτήματα συμβατά με το GridFTP.

Είναι φανερό πως το ΙΚΑΡΟΣ επιτυγχάνει να ενσωματώνει πολλαπλές λειτουργίες και να διαλειτουργεί αρμονικά με εξωτερικά πρωτόκολλα. Αυτό είναι εφικτό εξαιτίας της πληθώρας επιλογών παραμετροποίησης που διαθέτει, όπως τα case λειτουργίας που αναπτύχθηκαν σε αυτήν την παράγραφο, αλλά και το γεγονός ότι μπορεί να τα εφαρμόζει σε όλη την ροή των δεδομένων (τοπική και απομακρυσμένη πρόσβαση), κυρίως λόγω της χρήσης προτύπων όπως το HTTP.

Οι μηχανισμοί εισόδου/εξόδου του ΙΚΑΡΟΣ αποτελούν την καρδιά του πλαισίου. Στο οικοσύστημα των εφαρμογών στο οποίο ενεργεί κυριαρχούν οι λειτουργίες εγγραφής. Το ΙΚΑΡΟΣ σε αυτό το περιβάλλον υπερέχει έναντι των άλλων συστημάτων κυρίως λόγω των τεχνικών buffering και caching στην πλευρά του πελάτη, της χρήσης της τεχνικής reverse HTTP καθώς και στην υλοποίηση των διεργασιών write ως “reversed read”. Ο συνδυασμός του reversed read με την reverse HTTP, ανά περίπτωση, επιτρέπουν στο ΙΚΑΡΟΣ να ενεργεί κυρίως στο επίπεδο του δικτύου κάνοντας κυρίως routing των δεδομένων και αποφεύγοντας την εμπλοκή του λειτουργικού συστήματος. Οι συγκεκριμένες τεχνικές παρέχουν μέγιστη ευελιξία και επιτρέπουν σε τεχνικό επίπεδο να επιτευχθεί η άμεση πρόσβαση σε οποιουδήποτε αποθηκευτικό κόμβο εισόδου/εξόδου ανεξάρτητα από την βαθμίδα στην οποία αυτός ενεργεί, διασφαλίζοντας παράλληλα την αυτονομία υλοποίησης των επιμέρους υπηρεσιών.

5.4 Μηχανισμοί συστήματος αρχείου του ΙΚΑΡΟΣ (POSIX - Συμβατότητα)

Η επιλογή του HTTP ως το βασικό μηχανισμό λειτουργίας του ΙΚΑΡΟΣ επιτρέπει την χρήση αναρίθμητων πρωτοκόλλων και μηχανισμών που ήδη υπάρχουν και διαλειτουργούν διάφανα με το HTTP. Με τη χρήση του Web Distributed Authoring and Versioning (WebDav), το οποίο αποτελεί επέκταση του HTTP, δεν είναι απαραίτητο να δομηθούν επιπλέον μηχανισμοί που να υλοποιούν λειτουργίες συστήματος αρχείου.

Επιπροσθέτως, δεν απαιτείται η μετατροπή των εφαρμογών για να μπορέσουν να χρησιμοποιήσουν το ΙΚΑΡΟΣ, καθώς με την χρήση του WebDav είναι συμβατό με το πρότυπο POSIX. Λόγω της χρήσης του HTTP και των επεκτάσεων του το ΙΚΑΡΟΣ έχει τη δυνατότητα να εκμεταλλευτεί τις υπάρχουσες υλοποιήσεις που επιτρέπουν στους χρήστες να έχουν πρόσβαση σε WebDav πόρους κάνοντας χρήση της ομάδας εντολών τύπου POSIX.

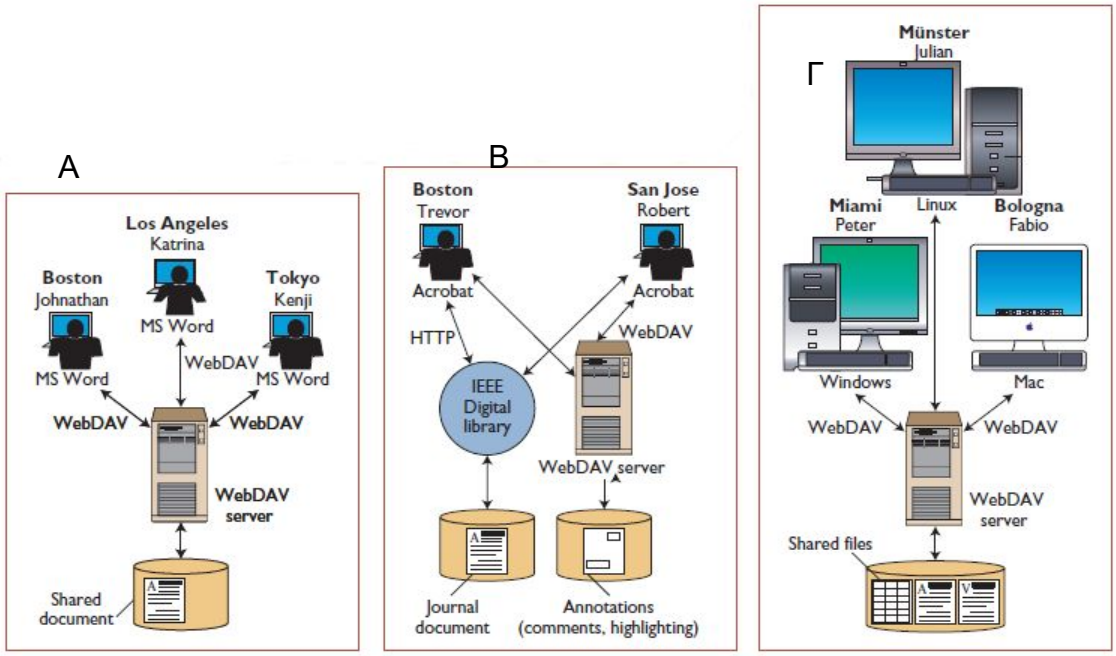
Παράδειγμα τέτοιων υλοποιήσεων αποτελεί το DavFS. Το WebDav διαθέτει ένα σύνολο από μεθόδους, επικεφαλίδες και τύπους περιεχομένου, που λειτουργούν επικουρικά ως προς το HTTP/1.1. Οι επιπλέον μέθοδοι που εισάγει το WebDav στοχεύουν στην

διαχείριση των ιδιοτήτων των πόρων, στην δημιουργία και διαχείριση των συλλογών των πόρων, τον χειρισμό των URL και το κλείδωμα των πόρων με σκοπό την αποφυγή φαινομένων σύγκρουσης. Έτσι δεν υπάρχουν φαινόμενα αντικατάστασης κάποιου πόρου που βρίσκεται υπό επεξεργασία από κάποιον άλλον χρήστη [31]. Τους παραπάνω μηχανισμούς δεν τους διαθέτει το HTTP/1.1.

Στην συνέχεια θα παρουσιαστούν εν συντομία κάποιες βασικές λειτουργίες και μηχανισμοί που παρέχονται από το WebDav. Το WebDav σχεδιάστηκε ώστε να έχει δυνατότητα ενσωμάτωσης σε υπάρχοντα εργαλεία με σκοπό την επέκταση τους ή τη ταυτόχρονη λειτουργία τους. Για παράδειγμα η προσάρτηση μιας WebDav υλοποίησης με την μορφή module στον πηγαίο κώδικα (source code) του apache HTTP εξυπηρετητή. Το WebDav περιλαμβάνει μια συλλογή από μεθόδους που επεκτείνουν αυτές που παρέχονται από το HTTP (get, head, post, options, put, delete, trace) και μπορεί να είναι εξαιρετικά χρήσιμες σε εφαρμογές που υποστηρίζουν την συνεργατική διαχείριση συστημάτων και πόρων. Οι μέθοδοι που προσφέρει το WebDav μπορούν να κατηγοριοποιηθούν σε τρεις ομάδες, ως ακολούθως [32]:

- Αποφυγή αντικατάστασης. Χρησιμοποιούνται τεχνικές όπως το edit token (χρήση εκ περιτροπής του δικαιώματος επεξεργασίας του διαμοιραζόμενου πόρου), shared locks (οι χρήστες δηλώνουν την πρόθεσή τους για επεξεργασία του διαμοιραζόμενου πόρου διατηρώντας το δικαίωμα τους να το επεξεργαστούν σε περίπτωση που ήδη το επεξεργάζεται κάποιος άλλος) και exclusive locking (μόλις κάποιος δηλώσει την πρόθεσή του να επεξεργαστεί το αρχείο τότε το σύστημα αποτρέπει οποιονδήποτε άλλον στο να προχωρήσει σε ταυτόχρονη επεξεργασία).
- Διαχείριση μεταδεδομένων. Για ένα μεγάλο εύρος δραστηριοτήτων που σχετίζονται με την διαχείριση των πόρων σε λογικό και φυσικό επίπεδο η ικανότητα της συσχέτισης των μεταδεδομένων με συγκεκριμένους πόρους είναι εξαιρετικά χρήσιμη. Σε ένα κατανεμημένο περιβάλλον είναι απαραίτητος ένας μεγάλος αριθμός από ιδιότητες για να επιτευχθεί η πλήρης περιγραφή της κατάστασης των πόρων [32].
- Διαχείριση πεδίου ονομάτων. Το WebDav επιτρέπει την δημιουργία συλλογών από πόρους ιστού. Οι εφαρμογές μπορούν να παράγουν λίστες με ιεραρχική δομή χρησιμοποιώντας τις συγκεκριμένες συλλογές και δίνοντας έτσι μεγαλύτερη ευελιξία στη διαχείριση των πόρων.

Με τη χρήση των εξαιρετικά ισχυρών δυνατοτήτων που παρέχει το WebDav είναι εφικτό να σχεδιαστούν πολλαπλές δομές συνεργατικής αλληλεπίδρασης μεταξύ εφαρμογών και χρηστών, παρέχοντας ταυτόχρονα σημαντικούς μηχανισμούς διαλειτουργικότητας. Στην συνέχεια παρουσιάζονται τρία λειτουργικά παραδείγματα (εικόνα 3) της χρήσης του WebDav.



Εικόνα 3:Παραδείγματα χρήσης του WebDav

Στην πρώτη περίπτωση (Α) παρουσιάζεται το σενάριο κατά το οποίο τρεις διαφορετικοί χρήστες επεξεργάζονται ταυτόχρονα ένα διαμοιραζόμενο αρχείο. Στην δεύτερη περίπτωση (Β) δυο χρήστες δημιουργούν ένα αρχείο μεταδεδομένων σχετικά με το διαμοιραζόμενο πόρο, εδώ απλά δημιουργούν και επεξεργάζονται ένα αρχείο με σχόλια που αφορά τον διαμοιραζόμενο πόρο. Στην τρίτη περίπτωση (Γ) πραγματοποιείται διαμοιρασμός αρχείων μεταξύ ετερογενών υποδομών, διαφορετικά λειτουργικά συστήματα.

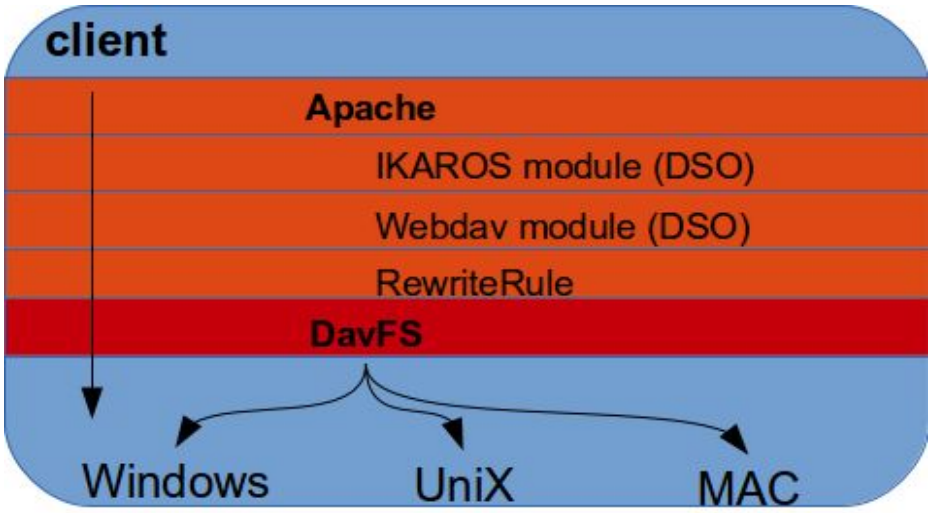
Οι μηχανισμοί υλοποίησης λειτουργιών συστήματος αρχείου του ΙΚΑΡΟΣ επιτυγχάνουν να παρουσιάζουν τα υποκείμενα αποθηκευτικά συστήματα ή συσκευές, τα οποία συντονίζει, ως ένα ενιαίο διαμοιραζόμενο σύστημα αρχείου με την χρήση των μηχανισμών που παρέχει το WebDav (σχήμα 16). Για να επιτευχθεί κάτι τέτοιο υλοποιείται μια τυπική WebDav εγκατάσταση και επιπρόσθετα απαιτείται από κάθε αίτημα, που αφορά την επεξεργασία των δεδομένων σε επίπεδο συστήματος αρχείου (για παράδειγμα λειτουργίες όπως: read, write), να διοχετεύεται για περαιτέρω επεξεργασία στο ΙΚΑΡΟΣ Apache module το οποίο έχει τη δυνατότητα να αλληλεπιδρά με τις αποθηκευτικές μονάδες.

Το ΙΚΑΡΟΣ γνωρίζει την ακριβή τοπολογία των δεδομένων και μπορεί να εξυπηρετήσει τέτοιου είδους αιτήματα. Κάτι τέτοιο επιτυγχάνεται ορίζοντας έναν κανόνα τύπου “rewrite” στο αρχείο παραμετροποίησης του Apache εξυπηρετητή. Ένα τυπικό παράδειγμα παρουσιάζεται στην συνέχεια:

RewriteRule data/(.*) ikaros?2&0&$1*

Με το πιο πάνω κανόνα απαιτείται από το σύστημα να επεξεργάζεται όλα τα αιτήματα που αφορούν τον φάκελο “data” κάνοντας χρήση του Apache module “ikaros”,

θέτοντας του ταυτόχρονα ως παράμετρο εισόδου τον αριθμό 2. Δηλαδή επιλέγεται η περίπτωση 2 (σχήμα 13)



Σχήμα 16: Μηχανισμοί συστήματος αρχείου του ΙΚΑΡΟΣ (POSIX - συμβατότητα)

5.5 Περιβάλλον εφαρμογών και μετρήσεων

Σε αυτήν την ενότητα παρουσιάζονται τα αποτελέσματα από ένα αριθμό πειραμάτων που υλοποιήθηκαν για τη μέτρηση της απόδοσης και της εν γένει συμπεριφοράς του ΙΚΑΡΟΣ.

Δημιουργήθηκαν δυο ομάδες πειραμάτων, στην πρώτη ομάδα μετρήσεων εκτελούνται δοκιμές απόδοσης και φόρτου χρησιμοποιώντας δεδομένα του πειράματος KM3NeT κάνοντας χρήση των εφαρμογών seatray [52] και ROOT [53] (σε πραγματικό περιβάλλον παραγωγής), μεταξύ διαφορετικών παραμετροποιήσεων του ΙΚΑΡΟΣ και των συστημάτων αρχείου HDFS και PVFS2. Ως αποθηκευτικές μονάδες χρησιμοποιήθηκαν συσκευές τύπου SOHO-NAS, συσκευές χαμηλών τεχνικών προδιαγραφών.

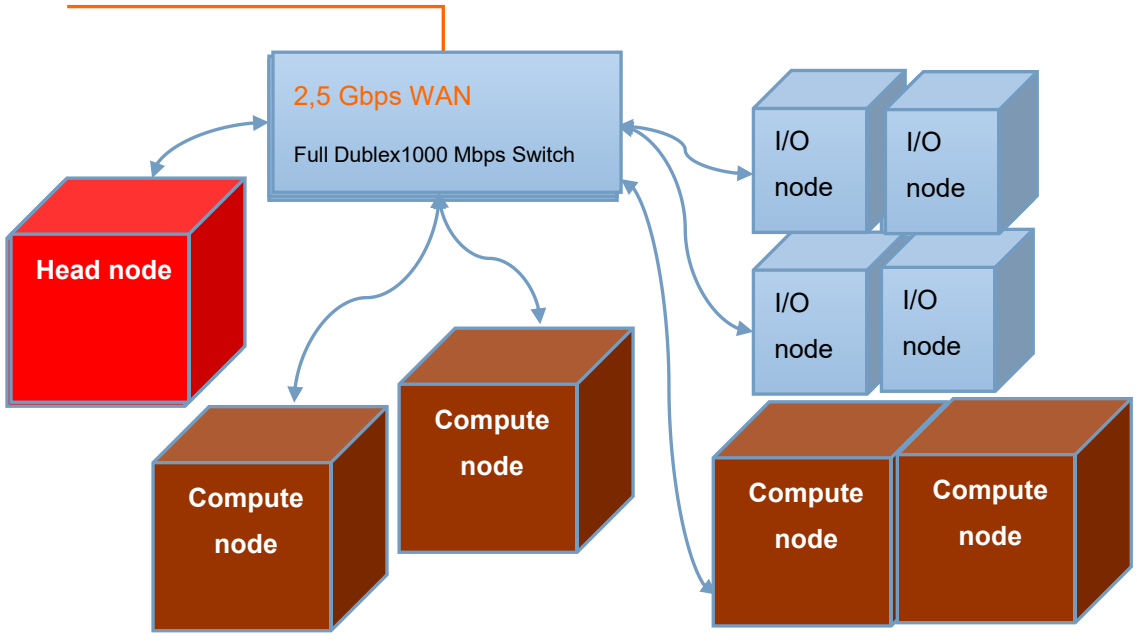
Στην δεύτερη ομάδα μετρήσεων χρησιμοποιείται το benchmark tool IOR-HPC που διενεργεί αιτήματα τυχαίας προσπέλασης σε παράλληλα προγραμματιστικά περιβάλλοντα, όπως το MPICH, μεταξύ διαφορετικών παραμετροποιήσεων του ΙΚΑΡΟΣ και του συστήματος αρχείου PVFS2. Οι αποθηκευτικές μονάδες που χρησιμοποιήθηκαν για αυτόν τον σκοπό έχουν ως βάση κοινό υλικό (commodity hardware). Ως commodity hardware ορίζεται το υλικό που είναι διαθέσιμο από κατασκευαστές υπολογιστικών συστημάτων που για τη κατασκευή του δεν απαιτείται κάποιος εξειδικευμένος μηχανισμός ή μεθοδολογία και παρέχει τυπικές τεχνικές προδιαγραφές όπως φαίνεται στον πίνακα 2.

Στον πίνακα 2 παρουσιάζονται οι τεχνικές προδιαγραφές των συστημάτων αποθήκευσης που χρησιμοποιούνται και διασαφηνίζονται οι όροι “χαμηλές τεχνικές προδιαγραφές” και “commodity υλικό”. Τα στοιχεία που δίνονται αφορούν την επεξεργαστική ισχύ και την κατανάλωση σε ηλεκτρική ενέργεια.

Πίνακας 2: Τεχνικές προδιαγραφές αποθηκευτικών συστημάτων

	(CPU-MHz)	Power Consumption (Watts)
Χαμηλές τεχνικές προδιαγραφές (soho-NAS)	200-800	5-20
commodity υλικό	>2000	>300

Το σύστημα που χρησιμοποιήθηκε για την εξαγωγή των αποτελεσμάτων είναι η συστοιχία υπολογιστών “ZEUS” του Ινστιτούτου Πυρηνικής και Σωματιδιακής Φυσικής (ΙΠΣΦ) του Εθνικού Κέντρου Έρευνας Φυσικών Επιστημών “Δημόκριτος” (Ε.Κ.Ε.Φ.Ε “Δημόκριτος”). Η συστοιχία υπολογιστών “ZEUS” (εικόνα 4) αποτελείται από 6 εξυπηρετητές βασιζόμενους σε επεξεργαστές AMD Opteron 270 (ο καθένας διαθέτει 4 επεξεργαστικούς πυρήνες και 4 GB μνήμης), 5 εξυπηρετητές βασιζόμενους σε επεξεργαστές AMD Opteron 2352 (ο καθένας διαθέτει 8 επεξεργαστικούς πυρήνες και 16 GB μνήμης), έναν εξυπηρετητή βασιζόμενο σε επεξεργαστές INTEL Xeon E5405 (διαθέτει 4 επεξεργαστικούς πυρήνες και 4 GB μνήμης) και ενεργεί ως κεντρικός κόμβος της συστοιχίας.



Εικόνα 4: Υπολογιστική συστοιχία Zeus

Ως κύριο αποθηκευτικό σύστημα, για τις συγκεκριμένες μετρήσεις, χρησιμοποιούνται 4 συσκευές SOHO-NAS βασιζόμενες σε επεξεργαστές τύπου ARM (με ρυθμό ρολογιού στα 800 Mhz, μνήμη 256 MB, δυνατότητα δικτυακής διασύνδεσης της τάξεως των 1000 Mbps και αποθηκευτική ικανότητα της τάξεως των 3 TB έκαστος). Η δικτυακή υποδομή

εξυπηρετείται από μια συσκευή μεταγωγής τύπου Ethernet που παρέχει πλήρη αμφίδρομη διασύνδεση μεταξύ των εξυπηρετητών της τάξεως των 1000 Mbps σε τοπικό επίπεδο και 2,5 Gbps WAN διασύνδεσης. Στις παρακάτω εικόνες διακρίνεται μέρος της υποδομής που χρησιμοποιήθηκε, το οποίο αποτελεί την συστοιχία υπολογιστών “ZEUS” και έχει σχεδιαστεί και υλοποιηθεί ώστε να καλύψει τις ανάγκες των εφαρμογών και των χρηστών του ΙΠΣΦ.

Το Ε.Κ.Ε.Φ.Ε “Δημόκριτος” συμμετέχει σε παγκόσμιας κλίμακας συνεργατικά πειράματα όπως το CMS [33] στο CERN και το KM3NeT [34]. Παράλληλα χρησιμοποιούνται επιστημονικές εφαρμογές που απαιτούν την πρόσβαση σε υψηλής απόδοσης παράλληλα συστήματα αρχείου σε τομείς όπως η απεικόνιση δεδομένων στο φάσμα των ακτίνων Χ, της υπολογιστικής ρευστό-δυναμικής και στην γενετική κωδικοποίηση των πρωτεϊνών [49]. Η συστοιχία υπολογιστών “ZEUS” έχει άμεση πρόσβαση σε δεδομένα που παράγονται από τα πιο πάνω πειράματα και εφαρμογές σε τοπικό επίπεδο αλλά και απομακρυσμένα μέσω της υποδομής πλέγματος European Grid Infrastructure (EGI) [7].

Η επιλογή των SOHO-NAS συσκευών για την δημιουργία ενός ενιαίου συστήματος αποθήκευσης που θα ικανοποιεί τις προαναφερόμενες δραστηριότητες βασίστηκε κυρίως στο χαμηλό κόστος κτήσης, τη χαμηλή κατανάλωση ενέργειας και τη μικρή προσπάθεια που απαιτείται στην παραμετροποίηση τους. Ταυτόχρονα είναι εμφανές πως αυτού του τύπου η συσκευές μπορούν να βοηθήσουν προς την κατεύθυνση της δημιουργίας μιας αποθηκευτικής υποδομής που θα επιτρέπει την κλιμάκωση του διαθέσιμου εύρους ζώνης (I/O και δίκτυο) σε αναλογία με την κλιμάκωση της διαθέσιμης χωρητικότητας αλλά και θα επιτρέπει την απομόνωση των λειτουργιών I/O μιας διεργασίας από τις αντίστοιχες λειτουργίες των άλλων διεργασιών με σκοπό της μέγιστη I/O απόδοση.

Όλα τα αυτά τα χαρακτηριστικά λειτουργούν επικουρικά στην προσπάθεια να είναι δυνατή η χρήση λιγότερο εξειδικευμένων αποθηκευτικών συσκευών και συστημάτων σε υψηλών απαιτήσεων εφαρμογές, που εκτελούνται σε κατανεμημένες υποδομές. Τα επιστημονικά πειράματα μεγάλης κλίμακας, όπως αυτό του LHC τείνουν να μετατρέψουν, σε λογικό επίπεδο σε σχέση με τα δεδομένα, τις υποδομές πλέγματος σε κεντρικοποιημένες υποδομές. Το γεγονός αυτό έρχεται σε αντίθεση με την αρχιτεκτονική μιας υποδομής πλέγματος απεμπολώντας έτσι μέρος των πλεονεκτημάτων που μπορεί να παρέχει μια τέτοιου είδους υποδομή.

Είναι προφανές ότι οι μηχανισμοί πλέγματος που χειρίζονται τα αποθηκευτικά συστήματα δεν έχουν εξελιχθεί αρκετά ώστε να παρέχουν την ίδια ποιότητα υπηρεσίας και την ίδια κατανεμημένη λογική χρησιμοποιώντας το ίδιο αποδοτικά ένα μεγάλο εύρος συσκευών, συστημάτων και εξαρτημάτων όπως συμβαίνει με τα επεξεργαστικά συστήματα.

Με την υλοποίηση του ΙΚΑΡΟΣ επιτυγχάνονται υψηλοί ρυθμοί μεταφοράς δεδομένων σε LAN και WAN περιβάλλοντα (τοπική και απομακρυσμένη πρόσβαση) ξεπερνώντας το φαινόμενο συμφόρησης που δημιουργούν οι κεντρικοί εξυπηρετητές παρέχοντας άμεση πρόσβαση των “πελατών” προς τις αποθηκευτικές μονάδες. Το ΙΚΑΡΟΣ παρέχει την δυνατότητα της πλήρους αξιοποίησης συσκευών χαμηλού κόστους, χαμηλής κατανάλωσης ενέργειας και χαμηλών τεχνικών χαρακτηριστικών με σκοπό την δημιουργία συστημάτων και υποδομών με υψηλούς ρυθμούς μεταφοράς δεδομένων.

5.6 Πειραματικά αποτελέσματα σύγκρισης των ΙΚΑΡΟΣ, NFS, HDFS και PVFS2 (χρήση υποδομής τύπου: soho-NAS)

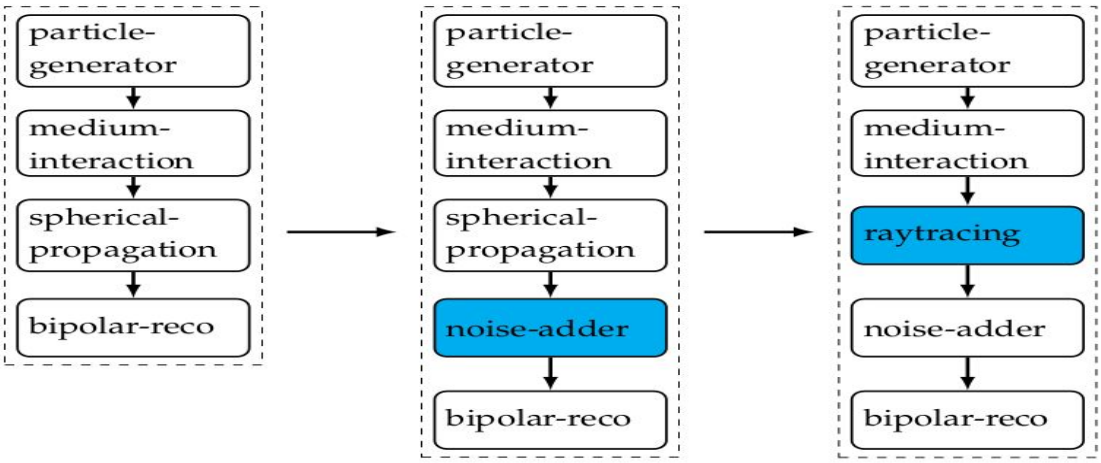
Στην συνέχεια παρουσιάζονται συγκριτικά αποτελέσματα μεταξύ των ΙΚΑΡΟΣ, NFS, HDFS και PVFS2, για ένα εύρος όγκου δεδομένων. Για τις μετρήσεις χρησιμοποιήθηκαν οι συσκευές αποθήκευσης τύπου SOHO-NAS που παρέχονται από την συστοιχία υπολογιστών “ZEUS”. Κατά την διάρκεια των μετρήσεων η συνολική υποδομή ήταν απομονωμένη από οποιαδήποτε άλλη δραστηριότητα, στοχεύοντας στην συγκέντρωση αποτελεσμάτων που θα υπόκεινται σε όσο τον δυνατόν μικρότερη εξωτερική παρεμβολή. Κάθε σημείο αναπαριστά τον μέσο όρο από 10 εκτελέσεις ενός πειράματος.

Για την μέτρηση του Throughput χρησιμοποιήθηκε το εργαλείο ganglia, το οποίο διατίθεται στην συστοιχία υπολογιστών Zeus, και τα αποτελέσματά επιβεβαιώθηκαν μετρώντας τον συνολικό χρόνο που απαιτείται για την μεταφορά των bits που αποτελούν το αρχείο και όχι μόνο το χρόνο που απαιτείται για τη μεταφορά αυτών κάθε αυτών των bits. Δηλαδή δεν μετρήθηκε μόνο ο χρόνος μεταφοράς των “ωφέλιμων” δεδομένων αλλά ο χρόνος ολοκλήρωσης της ενέργειας στο σύνολό της. Οι αιτήσεις μεταξύ των κόμβων εισόδου/εξόδου κατανέμονται ισόρροπα χωρίς να γίνεται χρήση κάποιου συγκεκριμένου αλγορίθμου.

Για τις μετρήσεις χρησιμοποιήθηκαν δεδομένα του πειράματος KM3NeT και των εφαρμογών seatray και ROOT. Το seatray διαθέτει μια βασική δομική μονάδα το “tray”, μια δομή δεδομένων στην οποία αποθηκεύονται αντικείμενα με βάση ένα μοναδικό όνομα. Κάθε “tray” αναπαριστά μια συγκεκριμένη χρονική περίοδο. Τα αντικείμενα που αποθηκεύονται στο “tray” περιγράφουν την κατάσταση του ανιχνευτή, δεδομένα φυσικής καθώς και την γεωμετρία του ανιχνευτή. Το σχήμα επεξεργασίας των δεδομένων καθορίζεται ως ακολούθως:

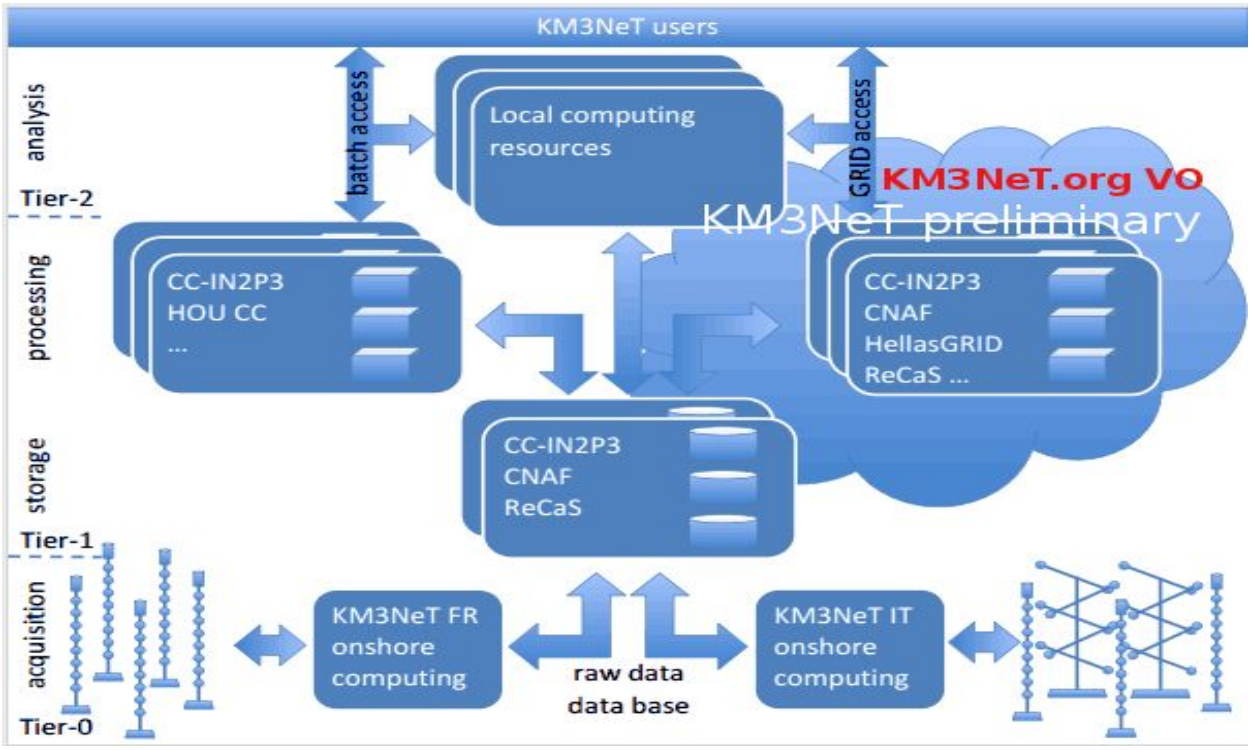
- Τα “trays” μετακυλίονται κατά μήκος μίας αλυσίδας από modules.
- Κάθε module προσθέτει νέα αντικείμενα ανάλογα με το ρόλο του.
- Τα αντικείμενα που προσθέτονται από κάθε module περιλαμβάνουν αποτελέσματα που έχουν προέλθει από αντικείμενα που είναι είδη αποθηκευμένα στο “tray”.

Με βάση την παραπάνω λογική, σχετικά με τη δομή ενός “tray”, μπορεί να οριστεί ένα format αρχείου το οποίο μπορεί να δημιουργηθεί οπουδήποτε στη συγκεκριμένη αλυσίδα. Η ροή των δεδομένων με τη μορφή των “trays” μπορεί να εγγραφεί σε ένα αρχείο και να είναι διαθέσιμο για το επόμενο module [52]. Γίνεται άμεσα αντιληπτό πως οι διαδικασίες εγγραφής στο σύστημα αρχείων και η απόδοσή τους αποτελούν σημαντικό κομμάτι της όλης διαδικασίας. Στο σχήμα 17 παρουσιάζεται η αλυσίδα ανάλυσης που ακολουθεί το seatray. Τα αρχεία εξόδου του seatray οπτικοποιούνται με την χρήση του ROOT. Το ROOT είναι ένα πλαίσιο ανάλυσης δεδομένων μεγάλης κλίμακας που έχει σχεδιαστή και υλοποιηθεί στο CERN.



Σχήμα 17: Αλυσίδα Ανάλυσης seatray

Στο σχήμα 18 παρατίθεται το υπολογιστικό μοντέλο του πειράματος KM3NeT μέρος του οποίου αποτελεί το seatray.

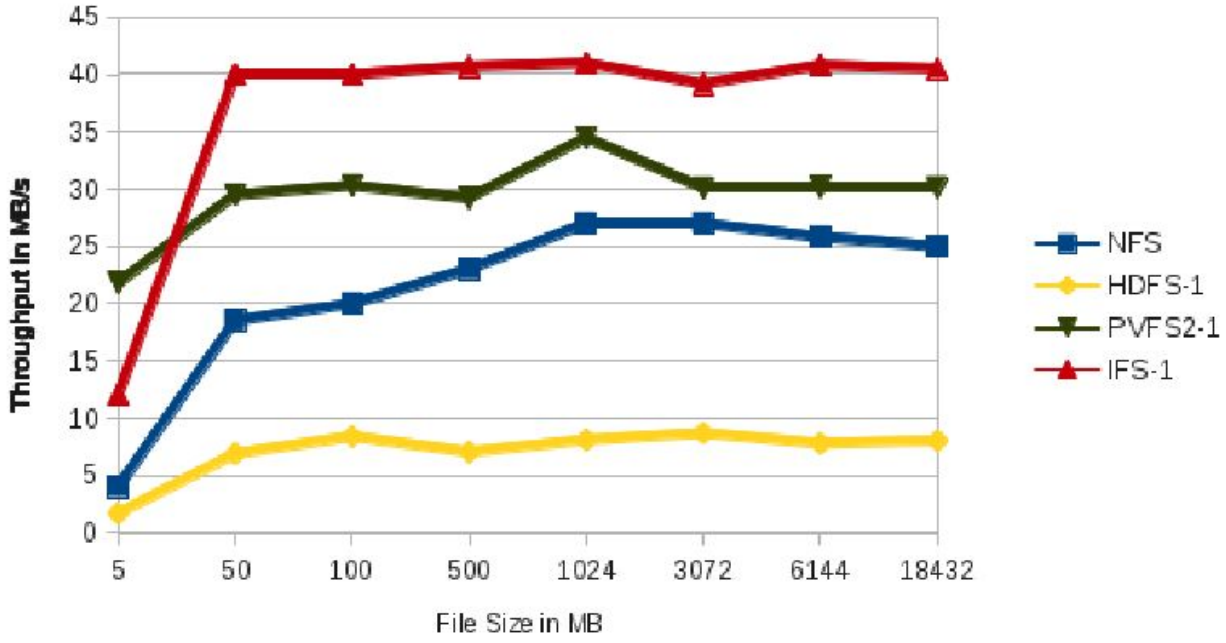


Σχήμα 18: Υπολογιστικό μοντέλο KM3NeT

Στις εικόνες 5-8 παρουσιάζεται η απόδοση του ΙΚΑΡΟΣ σε σύγκριση με τα NFS, HDFS, PVFS2 σε σχέση με την μεταφορά δεδομένων διαφόρων μεγεθών και κάνοντας χρήση ενός εύρους I/O κόμβων. Όλες οι μεταφορές δεδομένων διενεργήθηκαν μεταξύ του κεντρικού κόμβου της συστοιχίας υπολογιστών “ZEUS” και των συσκευών αποθήκευσης τύπου SOHO-NAS.

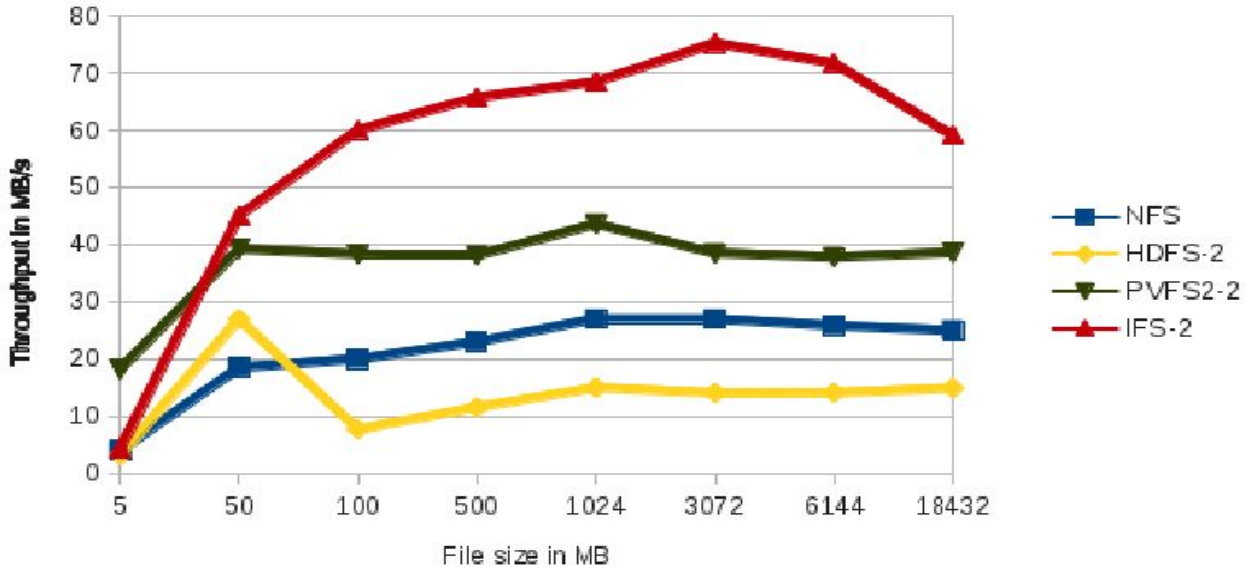
Το ΙΚΑΡΟΣ υπερτερεί στις περισσότερες περιπτώσεις παρουσιάζοντας καλύτερη κλιμάκωση, με την αύξηση των I/O κόμβων, και καταφέρνει να αξιοποιήσει καλύτερα το διαθέσιμο bandwidth. Αυτό επιτυγχάνεται κυρίως λόγω της τεχνικής buffering που εφαρμόζεται και στην πλευρά του πελάτη και στην πλευρά του εξυπηρετητή καθώς και

στην κατάλληλη επιλογή της κατανομή των HTTP GET αιτημάτων σε πολλαπλούς κόμβους εισόδου/εξόδου. Η επιλογή της κατανομής στους I/O κόμβους θα αναλυθεί διεξοδικά στο κεφάλαιο 7. Η απόδοση του HDFS είναι εξαιρετικά χαμηλή, γεγονός που οδηγεί στο συμπέρασμά ότι το HDFS δεν μπορεί να λειτουργήσει ομαλά στο συγκεκριμένο περιβάλλον και να εκμεταλλευτεί συσκευές αποθήκευσης χαμηλών τεχνικών προδιαγραφών τύπου SOHO-NAS.



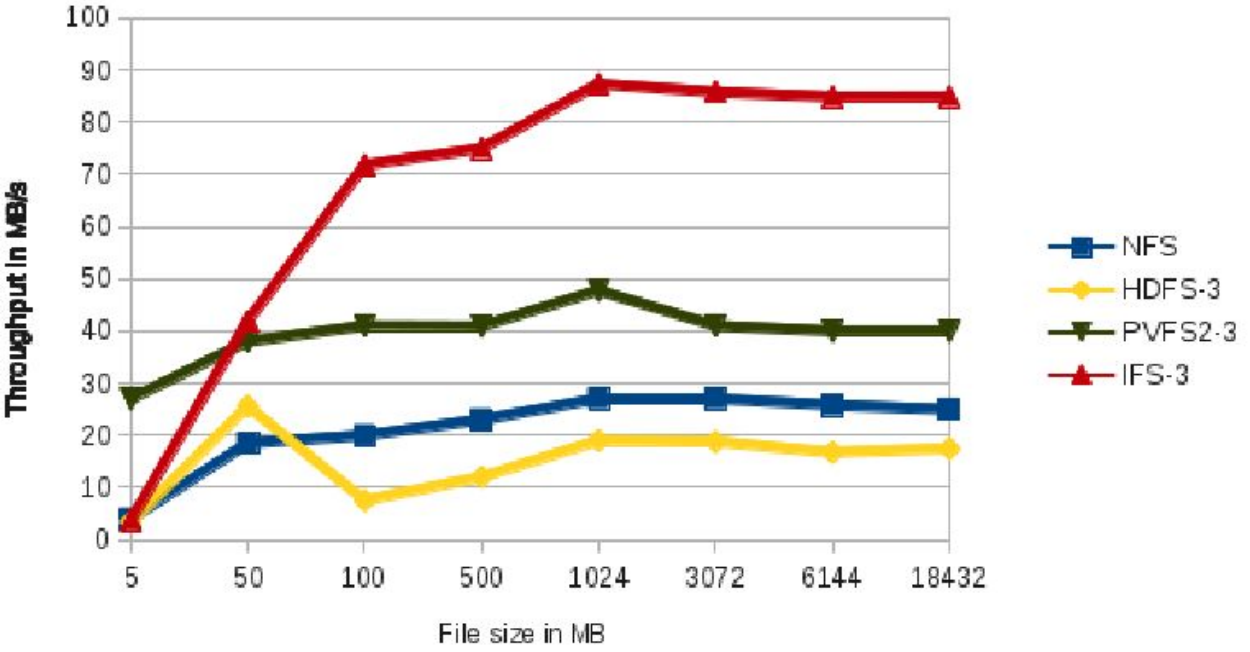
Εικόνα 5: Συγκριτικές μετρήσεις ΙΚΑΡΟΣ, HDFS, PVFS2, NFS (με την χρήση 1 I/O κόμβου), πείραμα:KM3NeT, Διεργασίες εγγραφής: case 4, υποδομή: soho-NAS

Απο την εικόνα 5 εξάγεται ότι όλα τα συστήματα έχουν παραπλήσια κλιμάκωση σε σχέση με την απόδοση όταν ενεργούν χρησιμοποιώντας έναν κόμβο εισόδου/εξόδου. Έτσι η απόδοσή τους είναι άμεσα συγκρίσιμη με την απόδοση του NFS. Το NFS χρησιμοποιείται ως μια σταθερά απόδοσης, δεν είναι δυνατόν να δειχθεί αν είναι αποδοτικότερο ή όχι σε σχέση με τα αλλά συστήματα καθώς ουσιαστικά ανήκει σε διαφορετική κατηγορία. Στις επόμενες μετρήσεις τα άλλα συστήματα θα χρησιμοποιήσουν περισσότερους από έναν αποθηκευτικούς κόμβους εισόδου/εξόδου ενώ το NFS λόγω της αρχιτεκτονικής του θα συνεχίσει να χρησιμοποιεί μονό έναν.

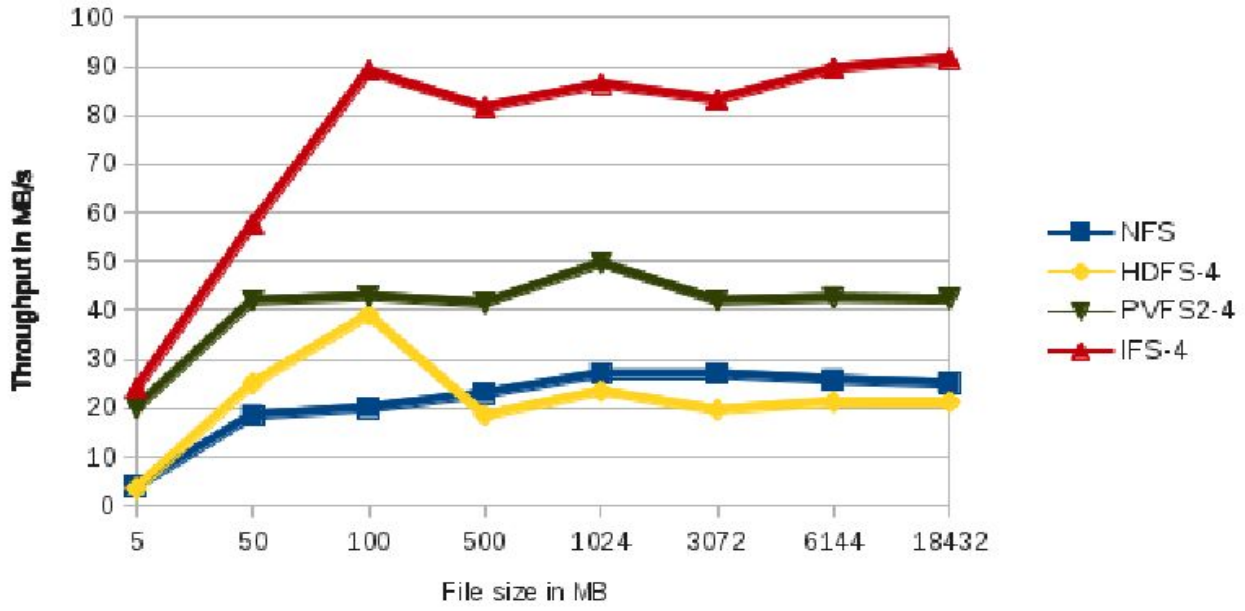


Εικόνα 6:Συγκριτικές μετρήσεις ΙΚΑΡΟΣ, NFS, HDFS, PVFS2, (με την χρήση 2 I/O κόμβων), πείραμα:KM3NeT, Διεργασίες εγγραφής: case 4, υποδομή: soho-NAS

Στην εικόνα 6 τα συστήματα χρησιμοποιούν δυο κόμβους εισόδου/εξόδου. Φαίνεται ότι ενώ τα άλλα συστήματα διατηρούν σταθερή την απόδοσή τους το ΙΚΑΡΟΣ σχεδόν την διπλασιάζει δείχνοντας τις δυνατότητες του. Έτσι φτάνει σε ένα μέγιστο για αρχεία των 3GB και στην συνέχεια περιορίζει την απόδοσή του κάτι που είναι απολύτως λογικό. Το μέγεθος των αρχείων που πρέπει να διαχειριστεί στην συνέχεια 6 και 18 GB χρησιμοποιώντας μόνο 2 αποθηκευτικούς κόμβους εισόδου/εξόδου τύπου SOHO-NAS δημιουργεί πολλά προβλήματα. Αυτού του τύπου οι I/O κόμβοι διαθέτουν εξαιρετικά χαμηλές τεχνικές προδιαγραφές με αποτέλεσμα να προκαλείται κορεσμός στο διαθέσιμο I/O bandwith, κυρίως στο επίπεδο των αποθηκευτικών μέσων.



Εικόνα 7:Συγκριτικές μετρήσεις ΙΚΑΡΟΣ, NFS, HDFS, PVFS2(με την χρήση 3 I/O κόμβων), πείραμα:KM3NeT, Διεργασίες εγγραφής: case 4, υποδομή: soho-NAS

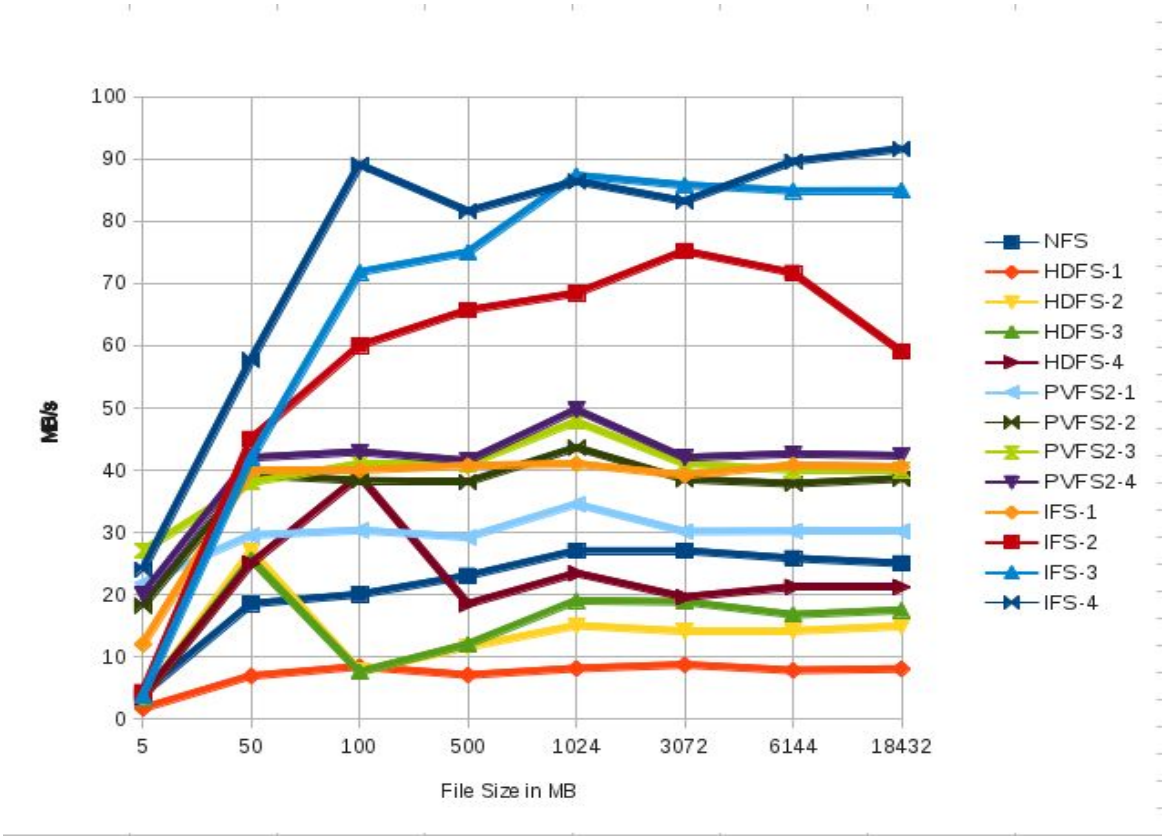


Εικόνα 8:Συγκριτικές μετρήσεις ΙΚΑΡΟΣ, NFS, HDFS PVFS2 (με την χρήση 4 I/O κόμβων), πείραμα:KM3NeT, Διεργασίες εγγραφής: case 4, υποδομή: soho-NAS

Στις εικόνες 7 και 8 το ΙΚΑΡΟΣ συνεχίζει να αυξάνει την απόδοση του και ουσιαστικά σταθεροποιείται κοντά στο “πρακτικά” μέγιστό διαθέσιμο εύρος ζώνης, σε επίπεδο δικτύου. Το θεωρητικό διαθέσιμο εύρος ζώνης αγγίζει τα 125 MB/s καθώς η δικτυακή διασύνδεση των κόμβων είναι στα 1000 Mbps). Όπως διαπιστώνεται στο κεφάλαιο 7 ο κυριότερος παράγοντας συμφόρησης του όλου συστήματος είναι η απόδοση του σκληρού δίσκου στο κεντρικό κόμβο της συστοιχίας. Γίνεται λοιπόν αντιληπτό ότι στη συγκεκριμένη περίπτωση η απόδοση του σκληρού δίσκου συμβαδίζει με αυτήν της δικτυακής υποδομής και άρα δεν έχει νόημα να χρησιμοποιηθούν περισσότεροι I/O κόμβοι, για μία διεργασία εγγραφής/ανάγνωσης. Αυτή η συμπεριφορά αναλύεται διεξοδικά στο κεφάλαιο 7.

Θα πρέπει επίσης να σημειωθεί ότι η απόδοση του PVFS2 στις διεργασίες εγγραφής όπως παρουσιάζεται στις προηγούμενες μετρήσεις είναι η αναμενόμενη. Η ομάδα που αναπτύσσει το όλο σύστημα αναφέρει πως κατά τις διεργασίες εγγραφής το PVFS2 έχει οριζόντια απόδοση και υπό συνθήκες μπορεί να παρουσιάσει χαμηλότερη απόδοση και από το NFS. Αυτό φαίνεται ότι οφείλεται στον αλγόριθμο που χρησιμοποιεί για την εγγραφή ενός αρχείου.

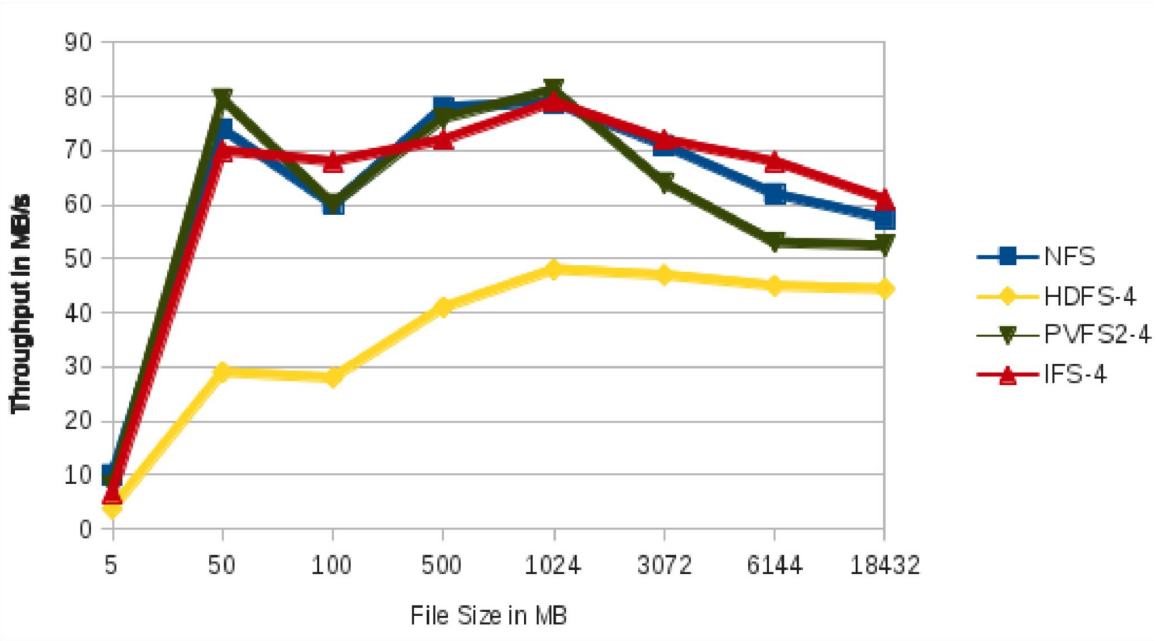
Ο αλγόριθμος απαιτεί την κατανομή συγκεκριμένου μεγέθους λωρίδων του αρχείου εκ περιτροπής σε όλους τους διαθέσιμους κόμβους I/O. Αυτό δημιουργεί διάφορα προβλήματα καθώς απαιτούνται συνεχώς νέες TCP συνδέσεις προς τους I/O κόμβους ενώ ταυτόχρονα δεν εξασφαλίζεται η σειριακή λειτουργία γραφής και ανάγνωσης της κεφαλής του σκληρού δίσκου. Η απουσία buffering και caching στην πλευρά του χρήστη επιτείνει αυτή τη δυσλειτουργία. Τα προβλήματα μπορεί να γίνουν ακόμα μεγαλύτερα αν χρησιμοποιηθούν περισσότεροι I/O κόμβοι από τους τέσσερις που χρησιμοποιούνται στις συγκεκριμένες μετρήσεις. Αυτή η περίπτωση αναλύεται στο κεφάλαιο 7.



Εικόνα 9:Συγκεντρωτικές μετρήσεις ΙΚΑΡΟΣ, NFS, HDFS PVFS2, πείραμα:KM3NeT Διεργασίες εγγραφής: case 4, υποδομή: soho-NAS

Στην εικόνα 9 παρουσιάζεται η απόδοση όλων των συστημάτων και όλων των παραμετροποιήσεων. Εδώ φαίνεται ξεκάθαρα ότι το ΙΚΑΡΟΣ κάνοντας χρήση 2 ή και περισσοτέρων κόμβων εισόδου/εξόδου εκμεταλλεύεται αποδοτικότερα το διαθέσιμο εύρος ζώνης και υπερέχει σε σχέση με τα άλλα συστήματα ακόμα και αν αυτά χρησιμοποιούν 4 κόμβους εισόδου/εξόδου.

Στην εικόνα 10 παρουσιάζεται η απόδοση του ΙΚΑΡΟΣ σε σύγκριση με τα NFS, HDFS, PVFS2 χρησιμοποιώντας 4 I/O κόμβους, σε σχέση με τις διεργασίες ανάγνωσης. Όλες οι μετρήσεις διενεργήθηκαν μεταξύ του κεντρικού κόμβου της συστοιχίας υπολογιστών “ZEUS” και των συσκευών αποθήκευσης τύπου SOHO-NAS. Παρατηρείται ότι το ΙΚΑΡΟΣ καταφέρνει να είναι ανταγωνιστικό με τα άλλα συστήματα και στις διεργασίες ανάγνωσης.

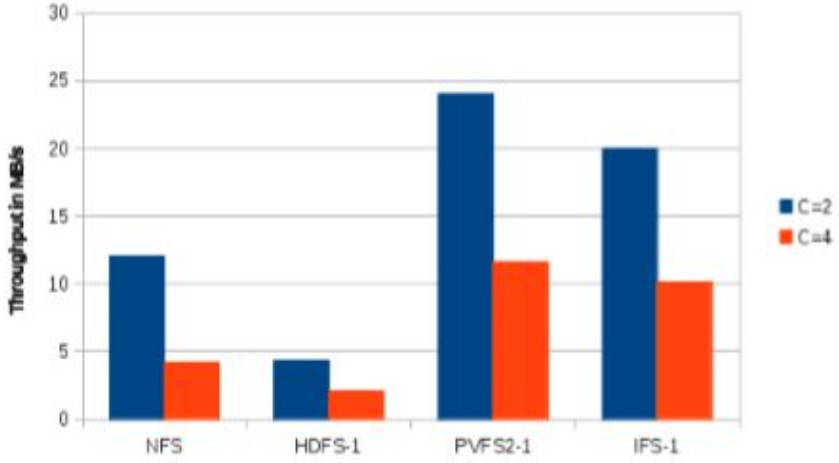


Εικόνα 10: Συγκριτικές μετρήσεις ΙΚΑΡΟΣ, NFS, HDFS PVFS2 (με την χρήση 4 I/O κόμβων), πείραμα:KM3NeT, Διεργασίες ανάγνωσης: case 2, υποδομή: soho-NAS

Στις εικόνες 11-14 παρουσιάζεται η απόδοση του ΙΚΑΡΟΣ σε σύγκριση με τα NFS, HDFS, PVFS2 κάνοντας χρήση ενός εύρους I/O κόμβων. Για τις μετρήσεις χρησιμοποιήθηκε αρχείο μεγέθους 6 GB και πραγματοποιήθηκαν 2 και 4 διεργασίες εγγραφής ταυτόχρονα. Σε αντίθεση με το προηγούμενο σετ μετρήσεων όπου χρησιμοποιήθηκε όλη η αποθηκευτική υποδομή αποκλειστικά και μόνο για μία διεργασία εγγραφής. Όλες οι μετρήσεις διενεργήθηκαν μεταξύ του κεντρικού κόμβου της συστοιχίας υπολογιστών “ZEUS” και των συσκευών αποθήκευσης τύπου SOHO-NAS, θα πρέπει να διευκρινιστεί πως όλα τα αιτήματα πραγματοποιήθηκαν από τον ίδιο πελάτη δηλαδή τον κεντρικό κόμβο της συστοιχίας.

Σε αυτήν την ομάδα μετρήσεων εξετάστηκε η απόκριση των συστημάτων σε πολλαπλά αιτήματα, όπως θα συνέβαινε σε πραγματικές συνθήκες. Εδώ πρέπει να σημειωθεί ότι ο αριθμός των αιτημάτων που πραγματοποιούνται είναι περιορισμένος καθώς οι αποθηκευτικοί κόμβοι εισόδου/εξόδου είναι τύπου SOHO-NAS και όπως αποδεικνύεται οι εξαιρετικά χαμηλές τεχνικές προδιαγραφές που παρέχουν δεν επιτρέπουν την πλήρη εκμετάλλευση των παρεχόμενων τεχνικών τύπου caching.

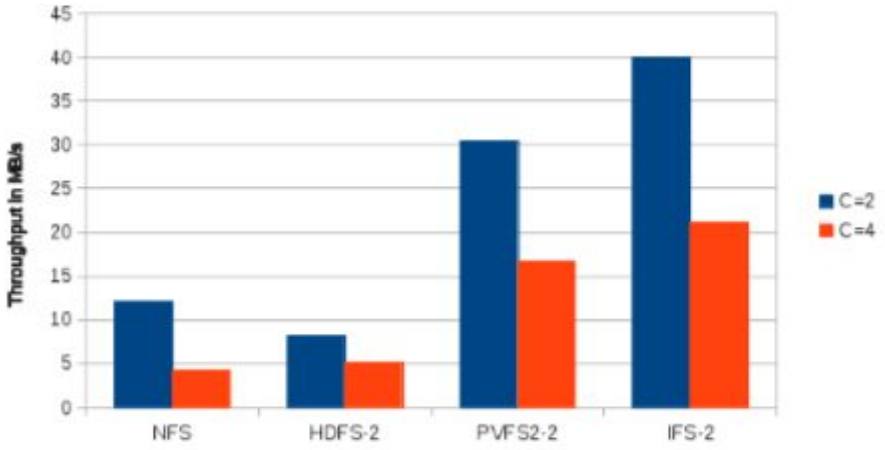
Στην παράγραφο 5.7 πραγματοποιούνται μετρήσεις με μεγάλο αριθμό ταυτόχρονων αιτημάτων από πολλαπλούς πελάτες χρησιμοποιώντας αποθηκευτικούς κόμβους εισόδου/εξόδου συμβατικών τεχνικών προδιαγραφών. Παρατηρείται ότι το ΙΚΑΡΟΣ υπερτερεί στις περισσότερες περιπτώσεις παρουσιάζοντας καλύτερη κλιμάκωση καθώς αυξάνεται ο αριθμός των I/O κόμβων και καταφέρνει να αξιοποιήσει καλύτερα το διαθέσιμο εύρος ζώνης.



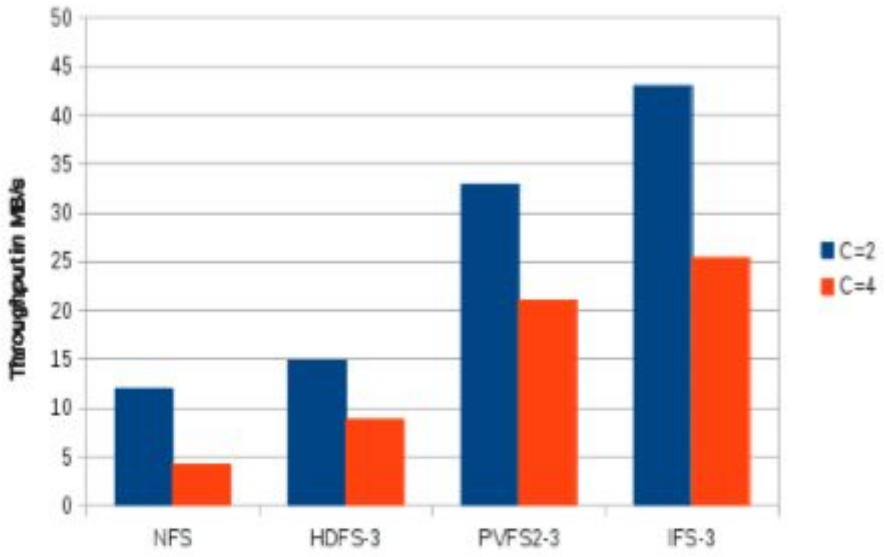
Εικόνα 11:Συγκριτικές μετρήσεις ΙΚΑΡΟΣ, NFS, HDFS PVFS2 (με την χρήση 1 I/O κόμβου, μέγεθος αρχείου 6 GB, 2 και 4 ταυτόχρονες εγγραφές), πείραμα:KM3NeT, Διεργασίες εγγραφής: case 4, υποδομή: soho-NAS

Στις εικόνες 11-14 παρατηρείται ότι και κάτω από μεγαλύτερο φορτίο όλα τα συστήματα διατηρούν την συμπεριφορά που είχαν στις προηγούμενες μετρήσεις απόδοσης. Σε αντίθεση με τις προηγούμενες μετρήσεις, τώρα πραγματοποιούνται ταυτόχρονα αιτήματα από περισσότερους χρήστες. Το ΙΚΑΡΟΣ δείχνει ξεκάθαρα την υπεροχή του όταν χρησιμοποιεί περισσότερους από ένα κόμβο εισόδου/εξόδου. Στις συγκεκριμένες εικόνες παρουσιάζεται η απόδοση μίας και μόνο μεταφοράς και όχι η συνολική απόδοση και των δύο ή και των τεσσάρων μεταφορών που διενεργούνται ταυτόχρονα.

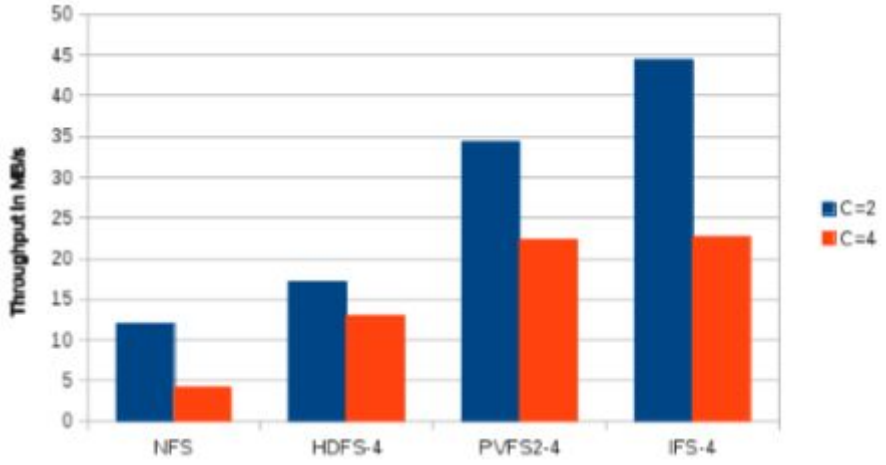
Πρακτικά η τιμή του Throughput στις συγκεκριμένες εικόνες αναφέρεται στο αποτέλεσμα του κλάσματος Throughput ανά transaction πρός τον συνολικό αριθμό των ταυτόχρονων αιτημάτων. Έτσι οι μετρήσεις είναι συγκρίσιμες με την προηγούμενη ομάδα μετρήσεων και μπορεί να γίνει αντιληπτή η συμπεριφορά των συστημάτων όταν θα πρέπει να αντεπεξέλθουν σε ταυτόχρονα πολλαπλά αιτήματα. Θα πρέπει επίσης να αναφερθεί ότι όλες οι υπόλοιπες διεργασίες επιτυγχάνουν την ίδια ρυθμοαπόδοση.



Εικόνα 12:Συγκριτικές μετρήσεις ΙΚΑΡΟΣ, NFS, HDFS PVFS2 (με την χρήση 2 I/O κόμβων, μέγεθος αρχείου 6 GB, 2 και 4 ταυτόχρονες εγγραφές), πείραμα:KM3NeT, Διεργασίες εγγραφής: case 4, υποδομή: soho-NAS



Εικόνα 13:Συγκριτικές μετρήσεις ΙΚΑΡΟΣ, NFS, HDFS PVFS2 (με την χρήση 3 I/O κόμβων, μέγεθος αρχείου 6 GB, 2 και 4 ταυτόχρονες εγγραφές), πείραμα:KM3NeT, Διεργασίες εγγραφής: case 4, υποδομή: soho-NAS



Εικόνα 14:Συγκριτικές μετρήσεις ΙΚΑΡΟΣ, NFS, HDFS PVFS2 (με την χρήση 4 I/O κόμβων, μέγεθος αρχείου 6 GB, 2 και 4 ταυτόχρονες εγγραφές), πείραμα:KM3NeT, Διεργασίες εγγραφής: case 4, υποδομή: soho-NAS

Το ΙΚΑΡΟΣ σχεδιάστηκε με σκοπό να έχει μέγιστη απόδοση στις διεργασίες εγγραφής λόγω της φύσης των εφαρμογών στις οποίες ενεργεί, κάτι που το επιτυγχάνει. Όπως διαπιστώνεται, τα άλλα συστήματα δεν καταφέρνουν να καλύψουν το διαθέσιμο εύρος ζώνης σε αυτού του είδους τις διεργασίες. Έτσι δημιουργούνται διάφορα ζητήματα στις εφαρμογές όπως η καθυστέρηση ολοκλήρωσης των διεργασιών αλλά και ακόμα η μη επιτυχής ολοκλήρωσή τους.

5.7 Πειραματικά αποτελέσματα σύγκρισης των ΙΚΑΡΟΣ και PVFS2 (χρήση commodity hardware)

Σε αυτήν την ενότητα πραγματοποιούνται δοκιμές σε ένα περιβάλλον διαφορετικό από αυτό της εφαρμογής KM3NeT. Σκοπός είναι να γίνει φανερό ότι το ΙΚΑΡΟΣ μπορεί να ανταποκριθεί πλήρως ενεργώντας στο γενικότερο πλαίσιο των παράλληλων προγραμματιστικών εφαρμογών που απαιτούν τυχαία προσπέλαση στο υποκείμενο

σύστημα αρχείων. Για την διενέργεια των μετρήσεων χρησιμοποιήθηκε το περιβάλλον MPICH καθώς και το εργαλείο IOR-HPC. Σε αυτήν την ομάδα μετρήσεων δεν συμπεριλήφθηκε το HDFS καθώς δεν είναι συμβατό με το πρότυπο POSIX και επίσης το IOR-HPC δεν διαθέτει κάποια διεπαφή που να το υποστηρίζει.

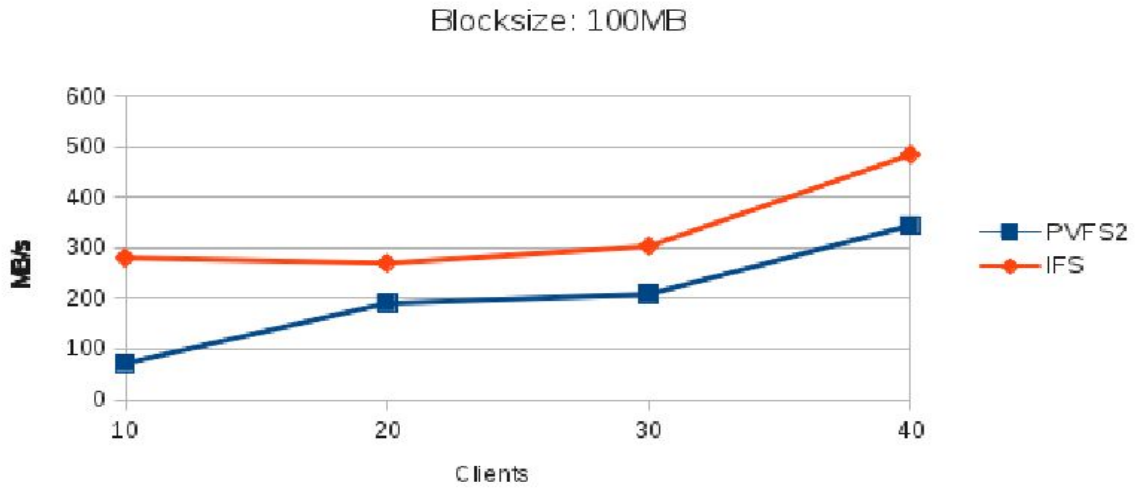
Όλες οι μετρήσεις πραγματοποιήθηκαν μεταξύ των υπολογιστικών κόμβων της συστοιχίας υπολογιστών “ZEUS”. Σε αντίθεση με τα αποθηκευτικά συστήματα τύπου SOHO-NAS που χαρακτηρίζονται ως συσκευές χαμηλών τεχνικών προδιαγραφών, οι υπολογιστικοί κόμβοι που χρησιμοποιούνται στις συγκεκριμένες μετρήσεις χαρακτηρίζονται ως commodity hardware με συμβατικές τεχνικές προδιαγραφές (πίνακας 2). Για την διενέργεια των μετρήσεων ακολουθήθηκε η ακόλουθη παραμετροποίηση και εκτέλεση:

1. mpirun -nolocal -np 10 -machinefile machines ./IOR
2. ./IOR -a MPIIO -r -c -o /mnt/DAV/filename

Εκτελείται η εντολή IOR κάνοντας χρήση του εργαλείου mpirun σε 10 κόμβους που αναφέρονται στο machinefile. Το IOR παραμετροποιήθηκε να χρησιμοποιεί την διεπαφή MPIIO ώστε να διαβάσει το αρχείο (/mnt/DAV/filename) με την επιλογή “-r” ή να το εγγράφει κάνοντας χρήση της επιλογής “-w”. Τέλος, κάνοντας χρήση της επιλογής “-c” εκμεταλλεύεται την collective I/O δυνατότητα της διεπαφής MPIIO.

Στις εικόνες 15 και 16 παρουσιάζεται η απόδοση του ΙΚΑΡΟΣ σε σύγκριση με το PVFS2, στις διεργασίες ανάγνωση κάνοντας χρήση 10 κόμβων I/O και λαμβάνοντας ταυτόχρονα αιτήματα μεταφοράς από πολλαπλούς πελάτες. Στο συγκεκριμένο σετ μετρήσεων πραγματοποιήθηκαν αιτήματα τυχαίας προσπέλασης όπου το Blocksize είναι αντίστοιχα 100MB και 1GB. Το όλο σύστημα ήταν απομονωμένο από εξωτερικούς παράγοντες.

Οι τιμές της ρυμοαπόδοσης που παρουσιάζονται στις πιο κάτω εικόνες παρέχονται απευθείας από το εργαλείο IOR-HPC το οποίο αποτελεί μια αξιόπιστη και γενικώς παραδεκτή πλατφόρμα για τη μέτρηση της απόδοσης των παράλληλων συστημάτων αρχείου, σε υπερυπολογιστικά περιβάλλοντα. Παρατηρείται ότι το ΙΚΑΡΟΣ συμβαδίζει, από πλευράς απόδοσης, με το PVFS2 και ανταποκρίνεται με την ίδια ευχέρεια καθόσον αυξάνεται ο αριθμός των ταυτόχρονων αιτημάτων από την πλευρά των πελατών.

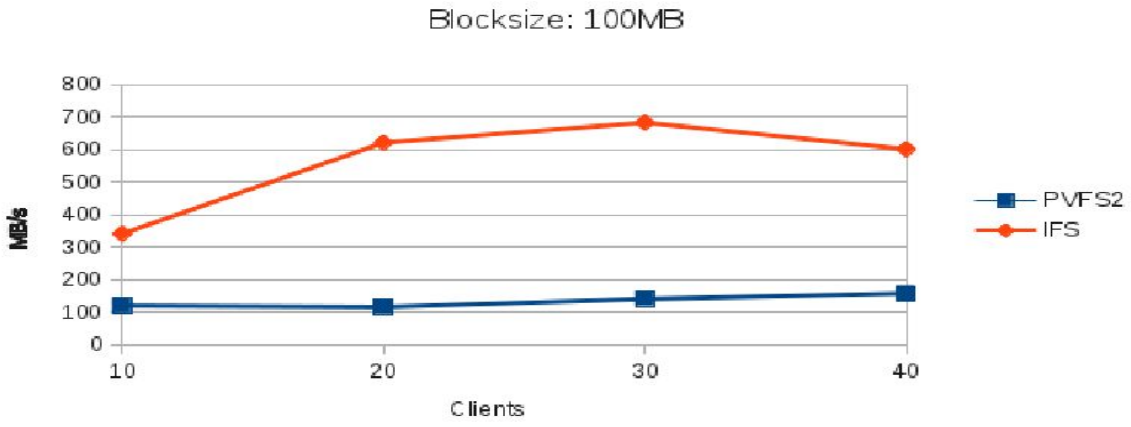


Εικόνα 15:Συγκριτικές μετρήσεις του ΙΚΑΡΟΣ με το PVFS2, στις διεργασίες ανάγνωσης (Blocksize: 100MB) πείραμα: IOR-HPC, Διεργασίες ανάγνωσης: case 2, υποδομή: commodity hardware

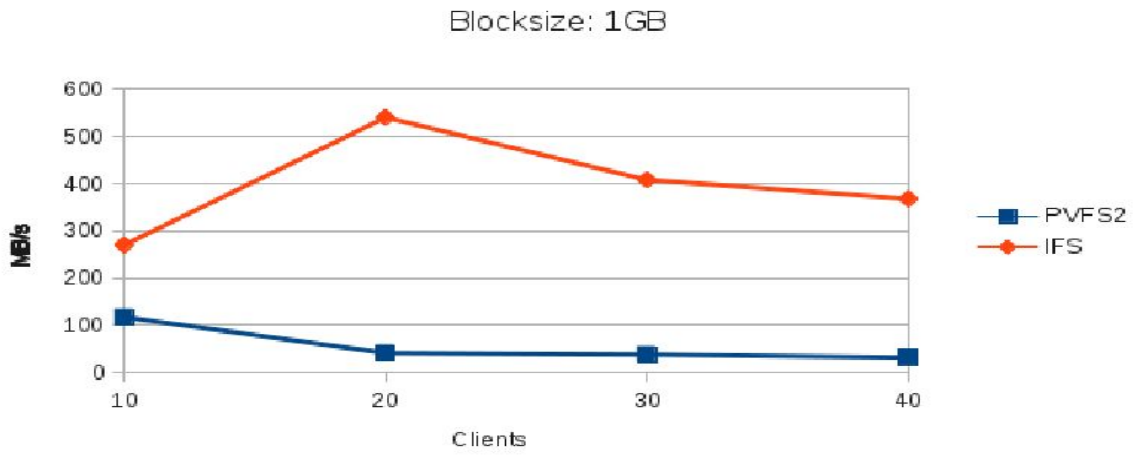
Εικόνα 16:Συγκριτικές μετρήσεις του ΙΚΑΡΟΣ με το PVFS2, στις διεργασίες ανάγνωσης (Blocksize: 1GB) πείραμα: IOR-HPC, Διεργασίες ανάγνωσης: case 2, υποδομή: commodity hardware

Στις εικόνες 17 και 18 παρουσιάζεται η απόδοση του ΙΚΑΡΟΣ σε σύγκριση με το PVFS2, στις διεργασίες εγγραφής, κάνοντας χρήση 10 κόμβων I/O και λαμβάνοντας ταυτόχρονα αιτήματα μεταφοράς από πολλαπλούς πελάτες. Στο συγκεκριμένο σετ μετρήσεων πραγματοποιήθηκαν αιτήματα τυχαίας προσπέλασης όπου το Blocksize είναι αντίστοιχα 100MB και 1GB. Για την διενέργεια των μετρήσεων χρησιμοποιήθηκε το MPICH και το IOR-HPC, όπως παρουσιάστηκε προηγουμένως.

Είναι εμφανές ότι το ΙΚΑΡΟΣ υπερτερεί παρουσιάζοντας καλύτερη κλιμάκωση, με την αύξηση των ταυτόχρονων αιτημάτων, και καταφέρνει να αξιοποιήσει καλύτερα το διαθέσιμο bandwidth. Στον αντίποδα, το PVFS2 φαίνεται πως καταρρέει μειώνοντας την απόδοσή του καθώς αυξάνονται τα αιτήματα.



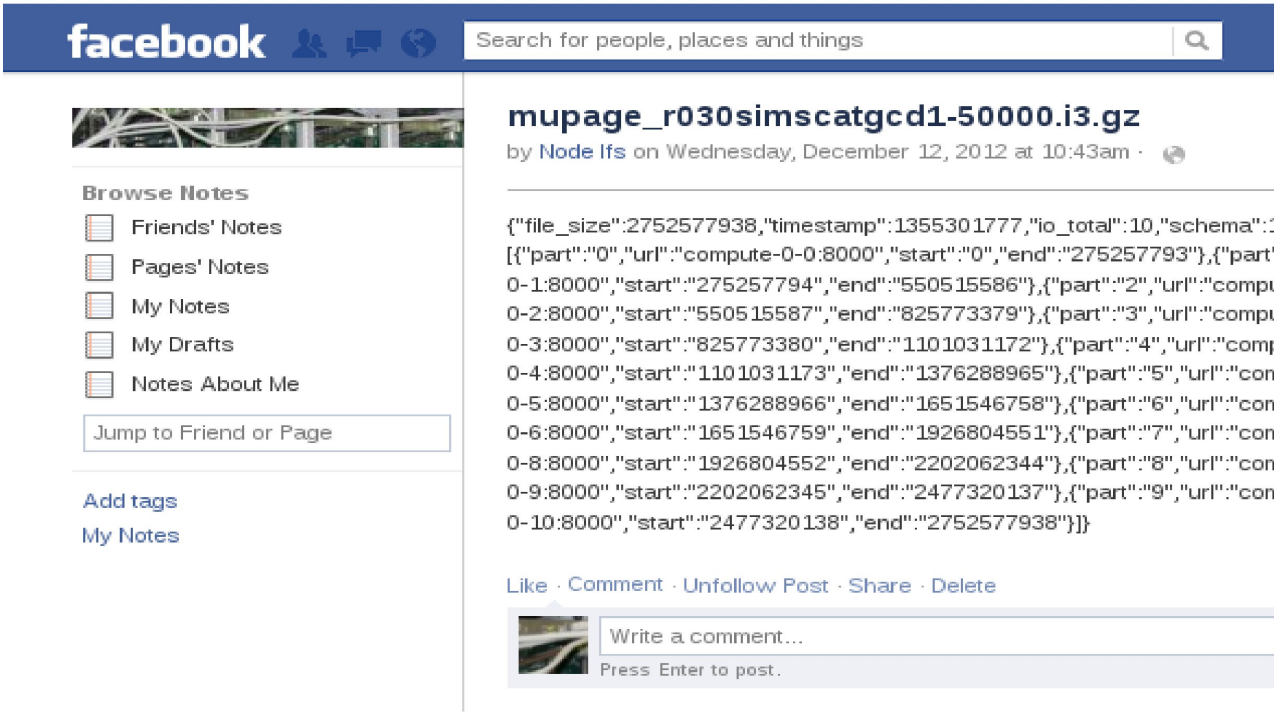
Εικόνα 17:Συγκριτικές μετρήσεις του ΙΚΑΡΟΣ με το PVFS2, στις διεργασίες εγγραφής (Blocksize: 100MB) πείραμα: IOR-HPC, Διεργασίες εγγραφής: case 4, υποδομή: commodity hardware



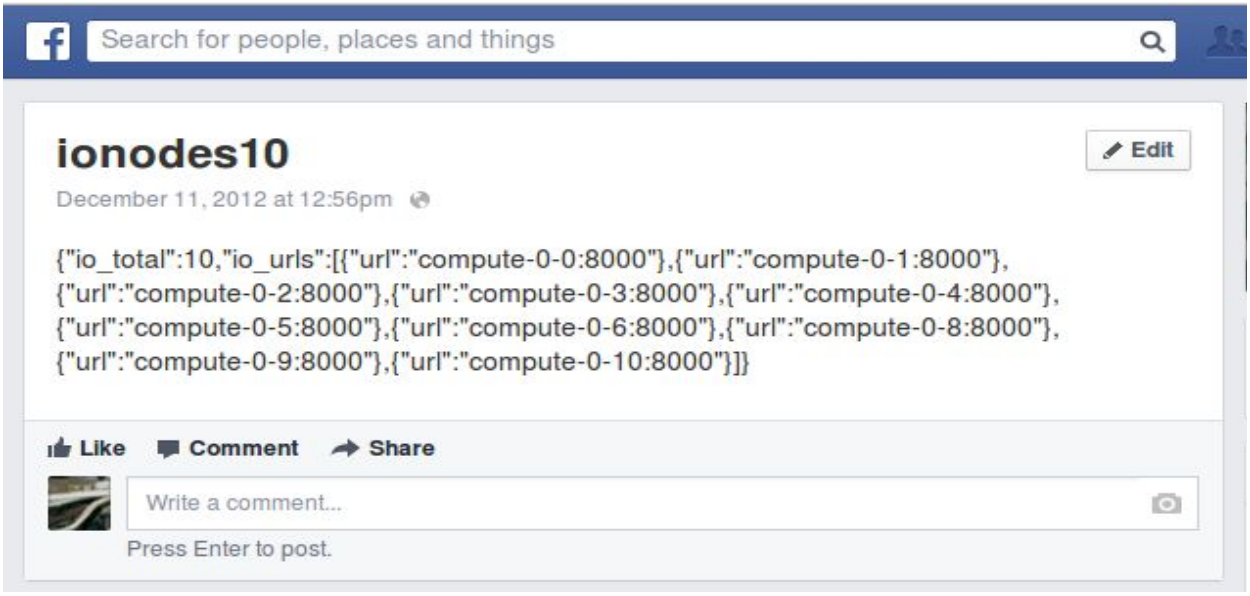
Εικόνα 18:Συγκριτικές μετρήσεις του ΙΚΑΡΟΣ με το PVFS2, στις διεργασίες εγγραφής (Blocksize: 1GB) πείραμα: IOR-HPC, Διεργασίες εγγραφής: case 4, υποδομή: commodity hardware

5.8 Επικοινωνία των μηχανισμών εισόδου/εξόδου με υπηρεσίες κοινωνικής δικτύωσης με σκοπό τη διαχείριση των μεταδεδομένων στο ΙΚΑΡΟΣ

Για την διεκπεραίωση των μετρήσεων είναι απαραίτητη η αλληλεπίδραση των μηχανισμών εισόδου/εξόδου με την οντότητα των μεταδεδομένων. Το ΙΚΑΡΟΣ, στην τυπική του παραμετροποίηση, διατηρεί τα μεταδεδομένα στους τοπικούς κόμβους καθώς αυτοί ενεργούν ως κόμβοι τύπου πελάτη, κόμβοι μεταδεδομένων και ως κόμβοι εισόδου/εξόδου. Ταυτόχρονα μπορεί να χρησιμοποιεί και κάποια εξωτερική οντότητα διαχείρισης των μεταδεδομένων η οποία έχει τη συνολική εικόνα της υποδομής σε όλες τις ιεραρχίες και λειτουργεί ως μια ευρύτερη πλατφόρμα διαχείρισης που παρέχει μεγάλη ευελιξία. Στις παρακάτω εικόνες παρουσιάζονται τα μεταδεδομένα ενός αρχείου του πειράματος KM3NeT, σε JSON μορφή, ως ένα Facebook note.



Εικόνα 19:Μεταδεδομένα ενός αρχείου του πειράματος KM3NeT, σε JSON μορφή, ως ένα Facebook note



Εικόνα 20:ΙΚΑΡΟΣ JSON meta-data object ώς Facebook note, διαθέσιμοι κόμβοι

Το ΙΚΑΡΟΣ αλληλεπιδρά με το Facebook Graph API και με το FQL API ώστε να ενημερώσει τον κατάλογο των μεταδεδομένων, να τα αναζητήσει και να τα διαμοιραστεί. Για να επιτευχθεί κάτι τέτοιο θα πρέπει πρώτα να έχει αυθεντικοποιηθεί και πιστοποιηθεί στο Facebook μέσω κάποιας ενεργής εφαρμογής (Facebook App). Ο εκάστοτε χρήστης θα πρέπει να επιτρέπει στην εφαρμογή να έχει πρόσβαση στα ”Extended Permissions”: ”create note”, ”user notes” and ”friends notes”. Στη συνέχεια ο χρήστης μπορεί να δημοσιοποιήσει, σε όσους επιθυμεί, και να διαμοιράσει τα μεταδεδομένα ως “notes” στο προφίλ του με την βοήθεια του Graph API. Μπορεί επίσης να τα αναζητήσει χρησιμοποιώντας του FQL API.

Αυτές οι δύο ενέργειες ακολουθούν τη ροή διεργασίας της υπηρεσίας ΙΚΑΡΟΣ [24] και καλούνται στα στάδια όπου απαιτείται κάποια αλληλεπίδραση με την οντότητα των μεταδεδομένων. Ενεργώντας σε ένα τόσο πολύπλοκο περιβάλλον με μικτές συνθήκες χρήσης τοπικών και απομακρυσμένων υποδομών γίνεται αντιληπτό πως η χρήση μιας εξωτερικής οντότητας, όπως το Facebook, για την υλοποίηση των μηχανισμών μεταδεδομένων η για την επέκταση της υπάρχουσας οντότητας, δεν αποτελεί παράγοντα μείωσης της απόδοσης του όλου συστήματος. Αντιθέτως, η χρήση μίας τέτοιας υποδομής προσδίδει ευχρηστία, μειωμένο κόστος διαχείρισης-ανάπτυξης και μεγαλύτερο βαθμό διείσδυσης των χρηστών στην διαχείριση του όλου συστήματος (user-driven). Το τελευταίο θεωρείται ως ένας πολύ σημαντικός παράγοντας για την ανάπτυξη των συστημάτων διαχείρισης και αποθήκευσης δεδομένων επόμενης γενιάς.

Γίνεται αντιληπτό ότι, οι μηχανισμοί εισόδου/εξόδου του ΙΚΑΡΟΣ αποτελούν την καρδιά της όλης αρχιτεκτονικής, αφού είναι αυτοί που υλοποιούν όλα τα αιτήματα και επιτρέπουν τη δημιουργία κοινών μεθόδων αντιμετώπισης των προβλημάτων στη γενικότερη ροή των δεδομένων, διασφαλίζοντας παράλληλα την αυτονομία υλοποίησης των επιμέρους υπηρεσιών. Έτσι, υλοποιείται ένας από τους βασικούς στόχους του ΙΚΑΡΟΣ που είναι η άρση των πολλαπλών επιπέδων συγχρονισμού και αναδιοργάνωσης στη συνολική ροή των δεδομένων από ένα δίκτυο ευρείας περιοχής προς ένα τοπικό δίκτυο και αντίστροφα.

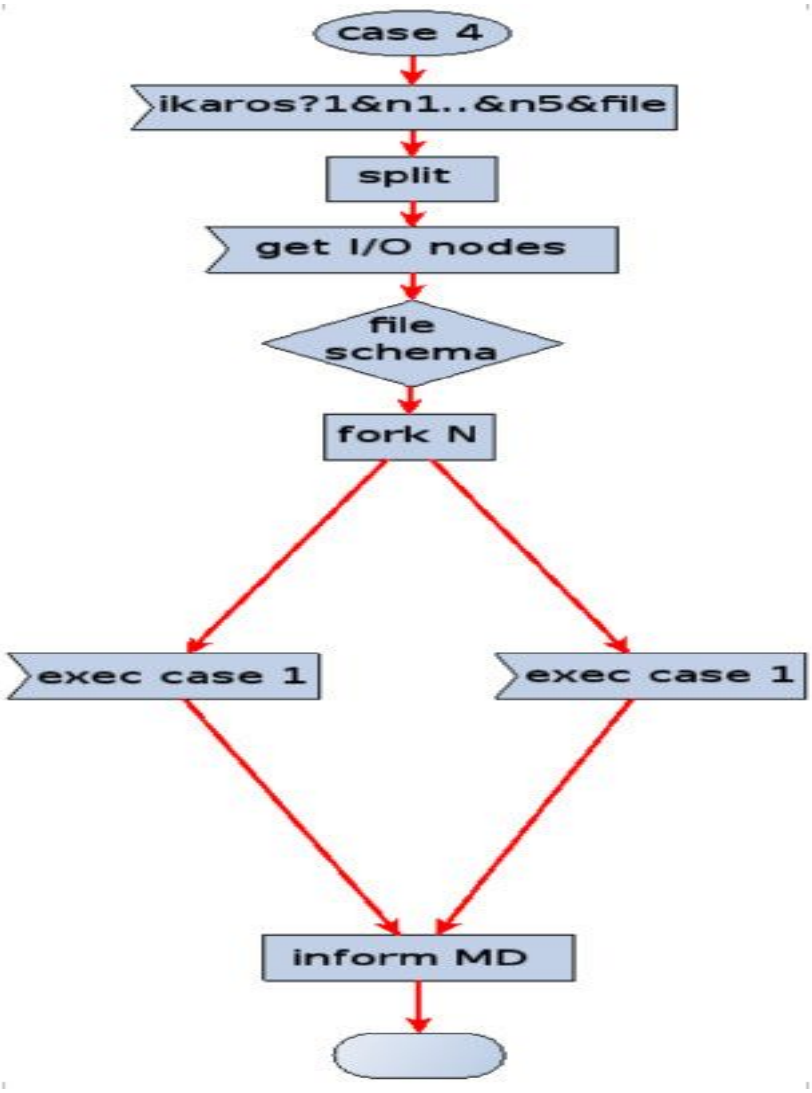
Επιπρόσθετα, η λογική που ακολουθούν οι μηχανισμοί εισόδου/εξόδου οδηγεί στην δημιουργία υποδομών που μειώνουν την αναντιστοιχία μεταξύ της διαθέσιμης

χωρητικότητας και του διαθέσιμου εύρους ζώνης, ενώ ταυτόχρονα επιτρέπει να δομηθούν συνέργειες μεταξύ ευρύτερων κοινοτήτων. Στη προκειμένη περίπτωσή μεταξύ υποδομών που υποστηρίζουν Web 2.0 τεχνολογίες και των επιστημονικών υπολογιστικών εφαρμογών. Τέλος, το ΙΚΑΡΟΣ στοχεύει στο να λειτουργεί ως ένα πλαίσιο που θα επιτρέπει την δημιουργία αποθηκευτικών συστημάτων νέας γενιάς, τα οποία θα μπορούν να ανταποκριθούν στις απαιτήσεις των eXascale συστημάτων από πλευράς κλιμάκωσης, απόδοσης και κατανάλωσης ενέργειας.

6. ΜΕΤΑΦΟΡΑ ΔΕΔΟΜΕΝΩΝ ΣΕ ΔΙΚΤΥΑ ΕΥΡΕΙΑΣ ΠΕΡΙΟΧΗΣ (WAN)

Όπως έχει προαναφερθεί, όλοι οι κόμβοι στο ΙΚΑΡΟΣ είναι ομότιμοι και άρα όλες οι λειτουργίες περιέχονται εντός του ίδιου αρθρώματος (module). Ο διαχωρισμός των λειτουργιών γίνεται σε λογικό επίπεδο και τα αιτήματα που δέχεται ο κόμβος είναι αυτά που ορίζουν την εκάστοτε επιλογή της λειτουργίας του. Η οντότητα μεταδεδομένων δίνει την δυνατότητα να ακολουθείται κοινή μεθοδολογία και αντιμετώπιση στα διαφορετικά επίπεδα στα οποία ενεργεί το ΙΚΑΡΟΣ.

Με αυτόν τον τρόπο αποφεύγεται ο διαχωρισμός που δομούν τα άλλα συστήματα, διατηρώντας παράλληλα την αυτονομία υλοποίησης των υπηρεσιών ανά επίπεδο. Όπως φαίνεται και στο σχήμα 19 το οποίο αναπαριστά την περίπτωση 4 (case 4), το ΙΚΑΡΟΣ ακολουθεί την ίδια λογική είτε λειτουργεί σε τοπικό επίπεδο είτε εκτελεί διεργασίες απομακρυσμένης πρόσβασης. Έτσι, με ακριβώς τον ίδιο μηχανισμό μπορεί να επιλεχθεί η χρήση παράλληλων καναλιών μεταφοράς ή διαχωρισμένων εξυπηρετητών ώστε να λειτουργήσει σε WAN περιβάλλοντα, με την ίδια ευελιξία με πρωτόκολλα όπως το GridFTP.

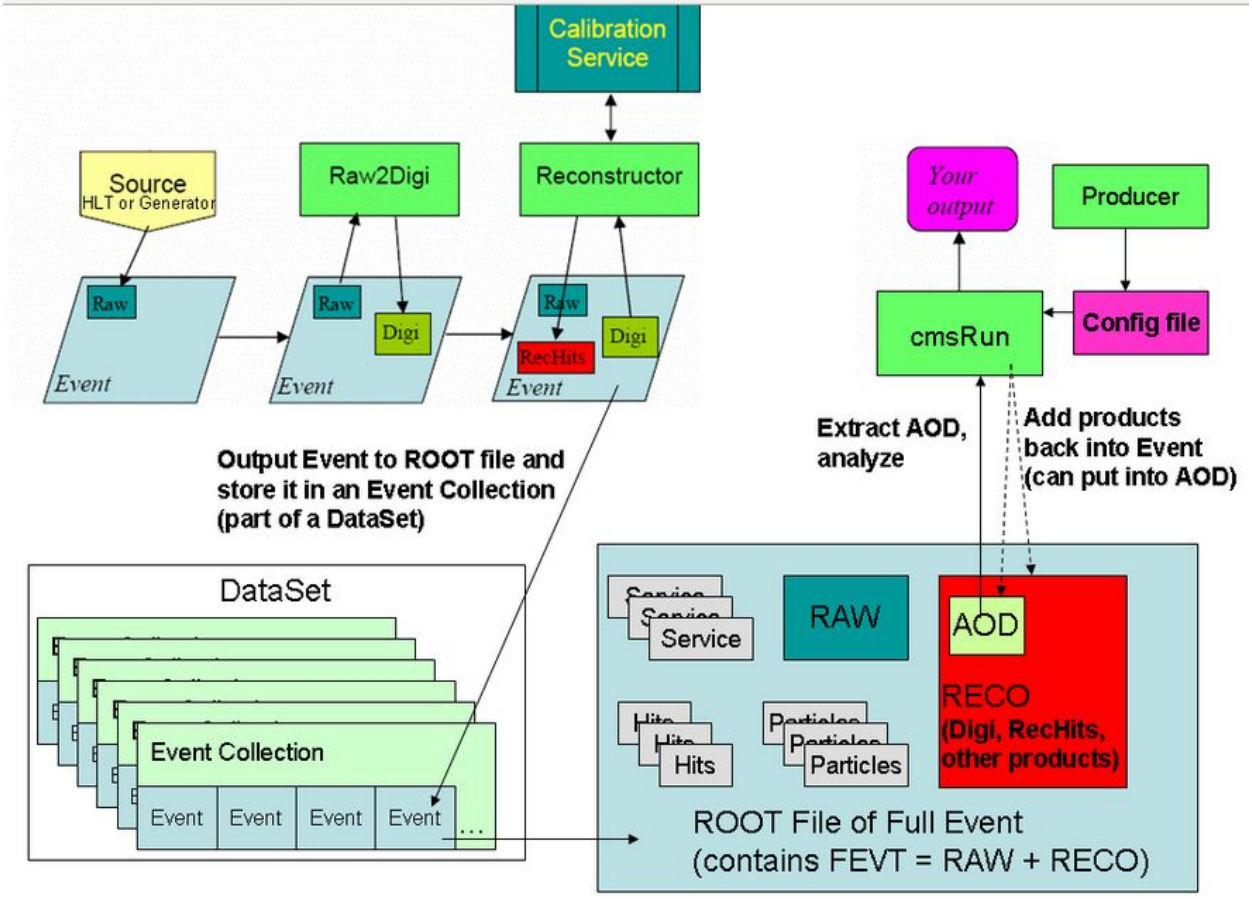


Σχήμα 19: ΙΚΑΡΟΣ module, διάγραμμα ροής περίπτωση 4, (διεργασίες εγγραφής- reversed read)

Στη ροή διεργασίας που περιγράφεται στο σχήμα 19 και η οποία αναφέρεται στην υλοποίηση των υπηρεσιών εγγραφής του προηγούμενου κεφαλαίου αρκεί να εκτελεστεί ένα αίτημα Ιστού (web request) επιλέγοντας την περίπτωση 4 (case 4) για να πραγματοποιηθεί μια μεταφορά αρχείου με τη χρήση παράλληλων καναλιών σε WAN περιβάλλον.

Οι εφαρμογές που χρησιμοποιούνται στο περιβάλλον που ενεργεί το ΙΚΑΡΟΣ, εδώ στο CMS-LHC-CERN, κινούνται σε ένα κατανεμημένο περιβάλλον παγκόσμιας κλίμακας με αποτέλεσμα να είναι απαραίτητη η συνεχής εναλλαγή στην πρόσβαση των δεδομένων, τοπική-απομακρυσμένη. Ενδεικτικά αναφέρεται ότι για τις υπολογιστικές ανάγκες του πειράματος LHC έχει υλοποιηθεί ένα υπολογιστικό πλέγμα με την συμμετοχή ετερογενών πόρων που αποτελούνται από συστοιχίες υπολογιστών και υπερυπολογιστών με περίπου 330 χιλιάδες επεξεργαστικούς πυρήνες και αποθηκευτική ικανότητα 220 Petabytes σε σκληρούς δίσκους και 240 Petabytes σε ταινίες (tapes).

Για τις μετρήσεις αυτής της ενότητας χρησιμοποιήθηκαν δεδομένα του πειράματος CMS-LHC-CERN καθώς και το λογισμικό CMSSW [76]. Το Event Data Model (EDM) που ακολουθεί το CMSSW περιγράφει την επεξεργασία των γεγονότων (Events). Ένα Event ξεκινά ως μια αδιαμόρφωτη συλλογή δεδομένων από κάποιον ανιχνευτή στη μορφή ενός C++ container. Καθώς προχωράει η διεργασία των δεδομένων του Event τα αποτελέσματα το αναδομούν και αποθηκεύεται ως ένα αντικείμενο “reconstructed data object” (RECO). Το Event πλέον διαθέτει τα αρχικά δεδομένα του ανιχνευτή, την ανάλυσή τους, καθώς και μεταδεδομένα. Τα δεδομένα του Event εγγράφονται σε αρχεία τα οποία με την σειρά τους οπτικοποιούνται με το ROOT [76]. Το EDM παρουσιάζεται στο σχήμα 20 [76].



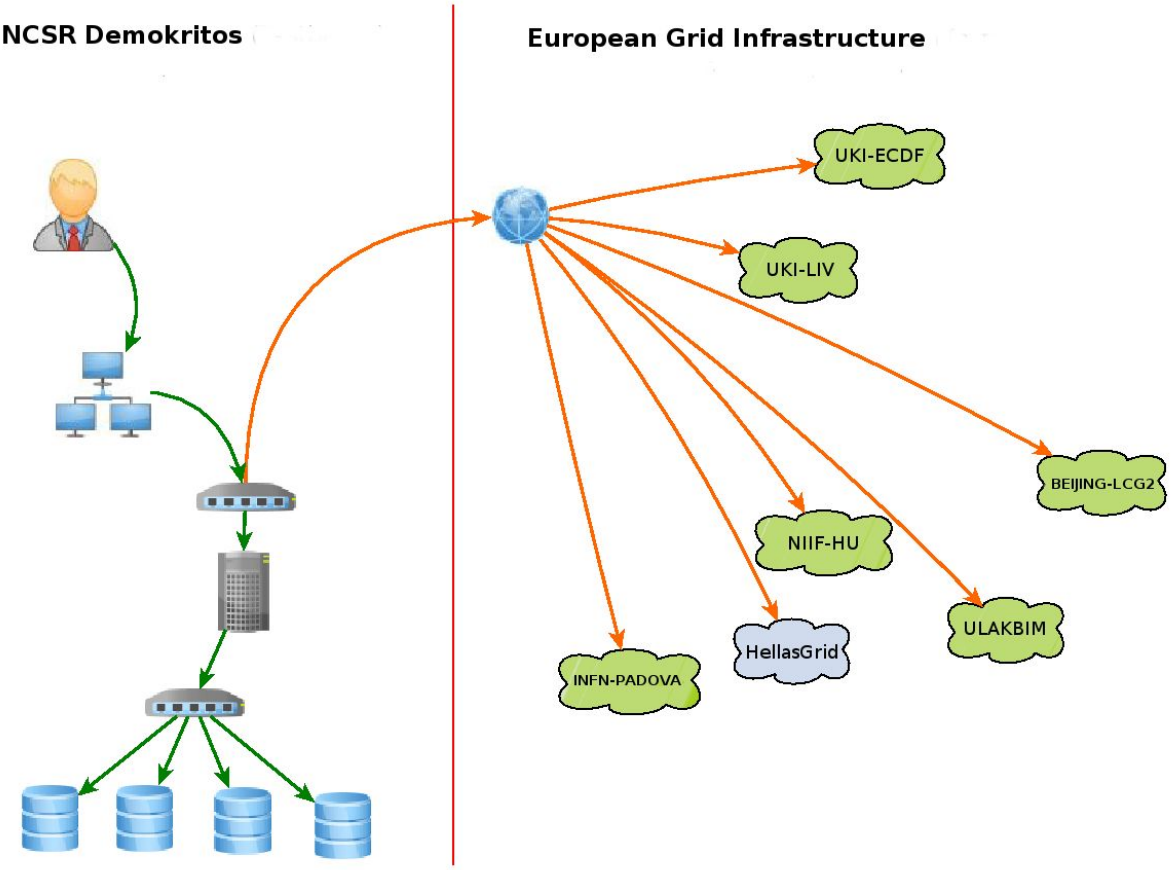
Σχήμα 20:CMS Event Data Model

Στην συνέχεια παρουσιάζεται το γενικότερο περιβάλλον καθώς και το δίκτυο κατανομής δεδομένων του CMS.

6.1 Πειραματικά αποτελέσματα σύγκρισης του ΙΚΑΡΟΣ με το GridFTP

Σε αυτή την ενότητα παρουσιάζονται συγκριτικά πειραματικά αποτελέσματα μεταξύ του ΙΚΑΡΟΣ και του GridFTP, χρησιμοποιώντας την υποδομή πλέγματος που παρέχεται από το EGI. Κάθε σημείο των πιο κάτω διαγραμμάτων αναπαριστά το μέσο όρο από 10 εκτελέσεις ενός πειράματος.

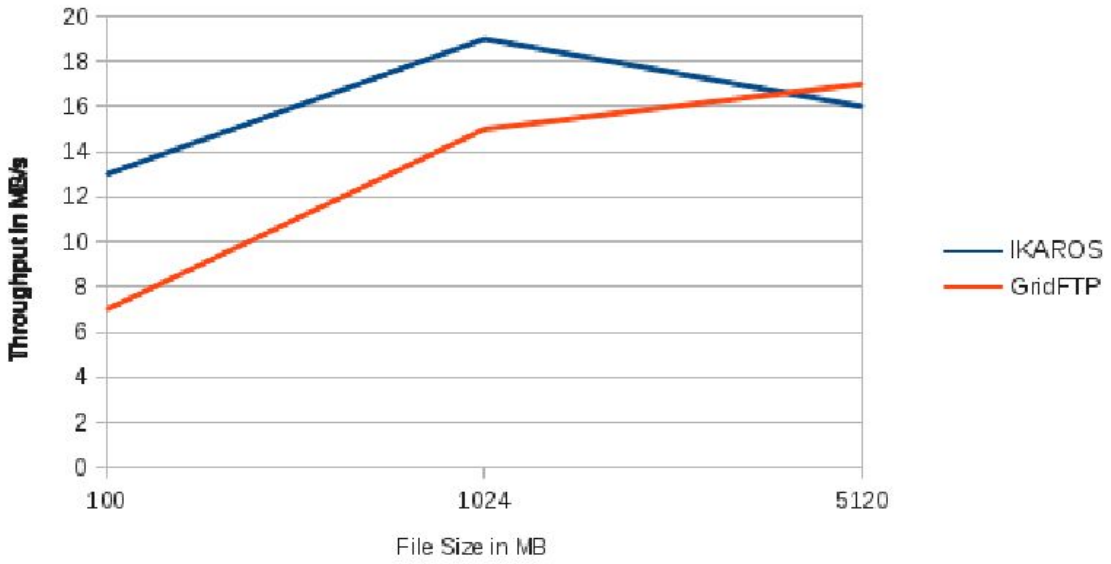
Στην εικόνα 21 παρουσιάζεται το testbed που χρησιμοποιήθηκε για τις μετρήσεις.



Εικόνα 21:ΙΚΑΡΟΣ, WAN Testbed

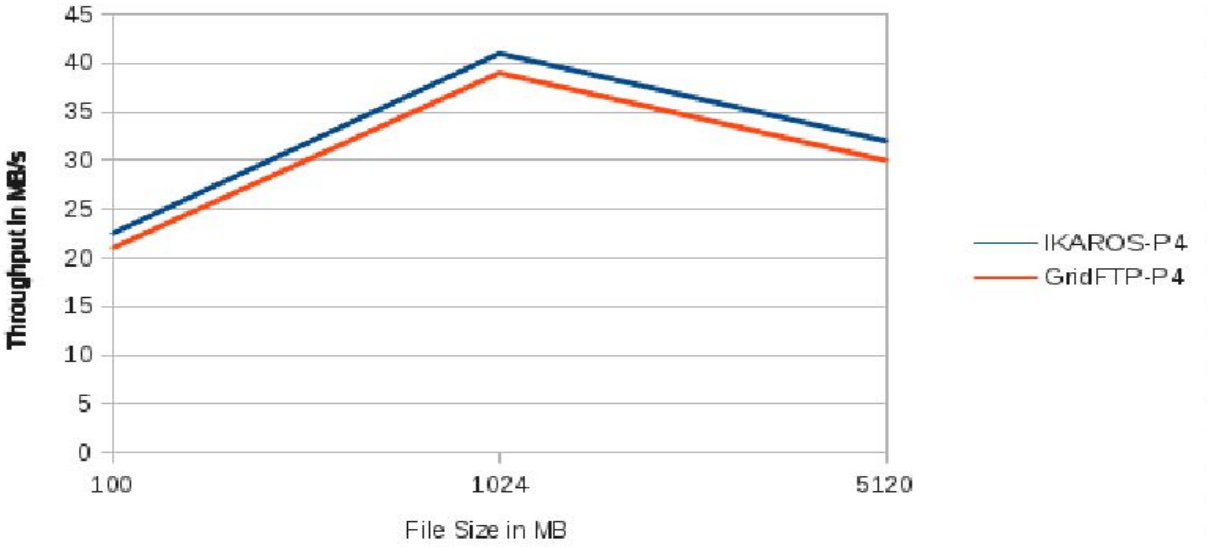
Στις εικόνες 22, 23 και 24 παρουσιάζεται η απόδοση του ΙΚΑΡΟΣ σε σύγκριση με το GridFTP, σε WAN περιβάλλον. Για το σκοπό των μετρήσεων χρησιμοποιήθηκαν διάφορα μεγέθη αρχείων. Για τις μεταφορές των δεδομένων έγινε χρήση 4 και 8 παράλληλων καναλιών μεταφοράς. Όλες οι μεταφορές δεδομένων πραγματοποιήθηκαν μεταξύ του κεντρικού κομβού του ”ZEUS” και τριών τοποθεσιών πλέγματος που ανήκουν στην ελληνική υποδομή πλέγματος HellasGrid. Οι τοποθεσίες πλέγματος είναι HG-01-GRNET (βρίσκεται στο Ε.Κ.Ε.Φ.Ε Δημόκριτος στην Αθήνα), HG-06-EKT (βρίσκεται στην Αθήνα) και HG-03-AUTH (βρίσκεται στη Θεσσαλονίκη).

Το ΙΚΑΡΟΣ, στις περισσότερες περιπτώσεις, αποδίδει ελαφρώς καλύτερα από το GridFTP αποδεικνύοντας ότι έχει την δυνατότητα να αξιοποιήσει την τεχνική των παράλληλων καναλιών μεταφοράς το ίδιο αποδοτικά όπως το GridFTP. Κάτι τέτοιο είναι αναμενόμενο καθώς στην πράξη και τα δύο χρησιμοποιούν το πρωτόκολλο TCP για την μεταφορά των δεδομένων.

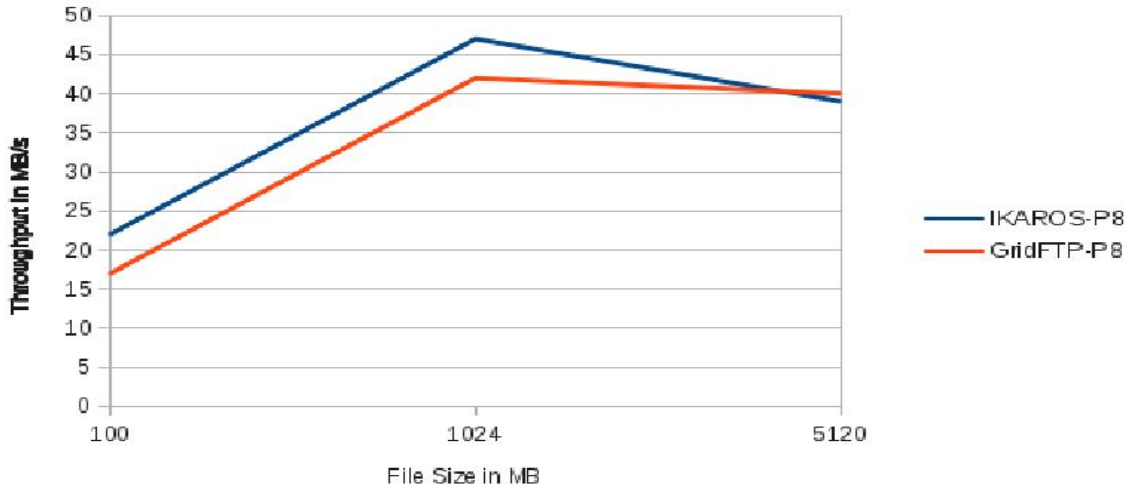


.

Εικόνα 22: Σύγκριση του ΙΚΑΡΟΣ με το GridFTP, HellasGrid



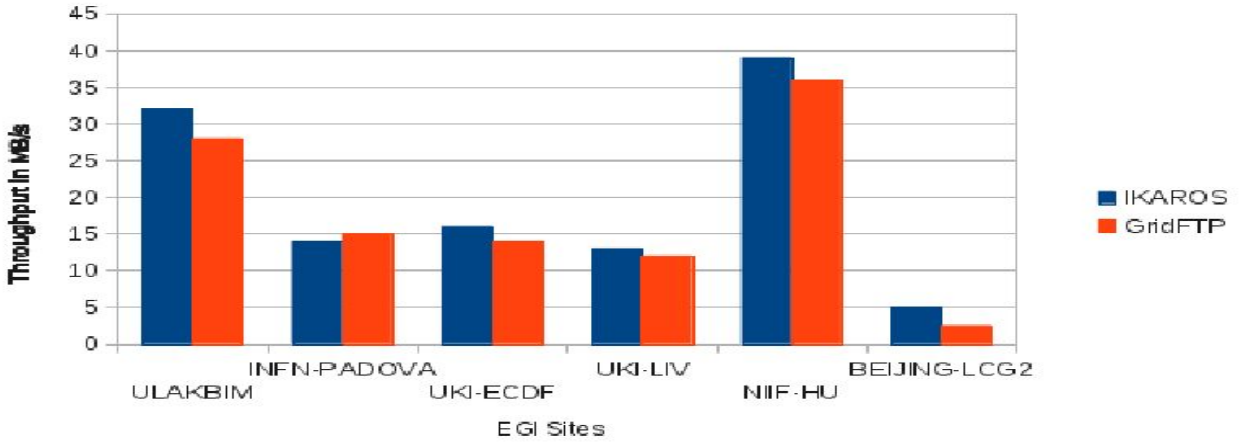
Εικόνα 23:Σύγκριση του ΙΚΑΡΟΣ με το GridFTP, HellasGrid (4 παράλληλα κανάλια μεταφοράς)



Εικόνα 24:Σύγκριση του ΙΚΑΡΟΣ με το GridFTP, HellasGrid (8 παράλληλα κανάλια μεταφοράς)

Στην εικόνα 25 παρουσιάζεται η απόδοση του ΙΚΑΡΟΣ σε σύγκριση με το GridFTP σε WAN περιβάλλον. Για το σκοπό των μετρήσεων χρησιμοποιήθηκε το ίδιο αρχείο, μεγέθους 1GB, σε ένα παγκόσμιας κλίμακας γεωγραφικά κατανεμημένο περιβάλλον. Οι τοποθεσίες πλέγματος που χρησιμοποιήθηκαν ήταν TR-10-ULAKBIM (βρίσκεται στην Τουρκία), INFN-PADOVA (βρίσκεται στην Ιταλία), UKI-SCOTGRID-ECDF (βρίσκεται στη Μεγάλη Βρετανία), UKI-NORTHGRID-LIV-HEP (βρίσκεται στη Μεγάλη Βρετανία), NIIF-HU (βρίσκεται στην Ουγγαρία) και BEIJING-LCG2 (βρίσκεται στην Κίνα).

Το ΙΚΑΡΟΣ στις περισσότερες περιπτώσεις αποδίδει ελαφρώς καλύτερα από το GridFTP αποδεικνύοντας ότι μπορεί να είναι ανταγωνιστικό σε πραγματικές συνθήκες λειτουργίας. Θα πρέπει να τονιστεί ότι οι συγκεκριμένες μετρήσεις δεν έγιναν σε απομονωμένο και ελεγχόμενο περιβάλλον καθώς ταυτόχρονα πραγματοποιούνται και άλλες μεταφορές κάτι που δικαιολογεί την ελαφρά διαφοροποίηση για τις μεταφορές προς το INFN-PADOVA.

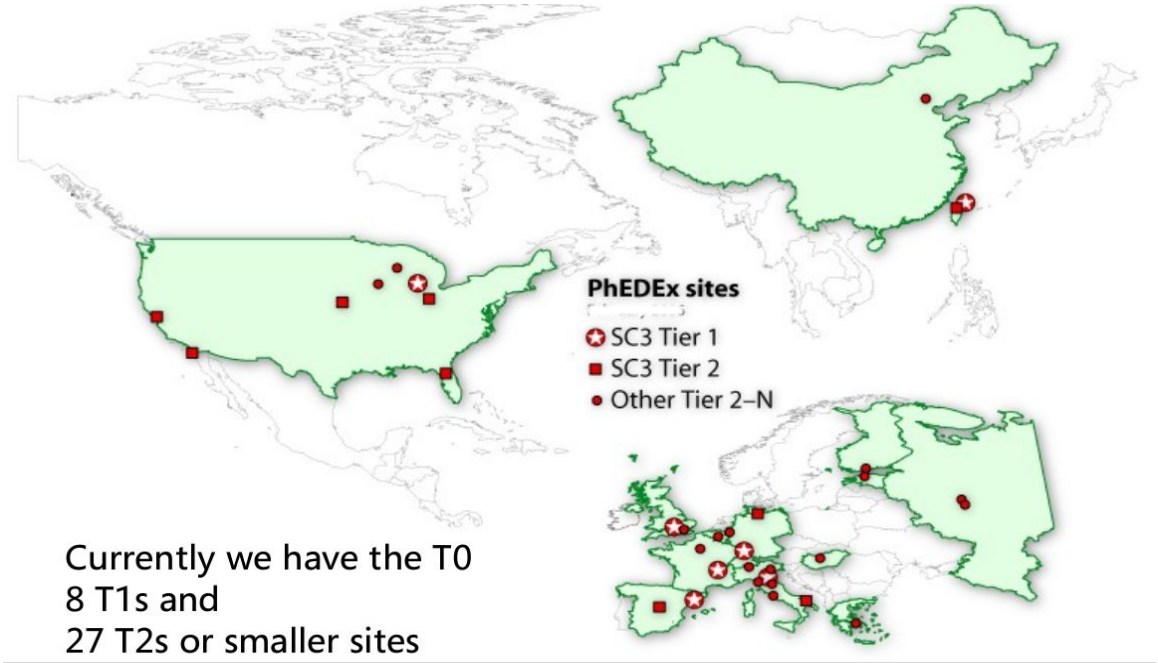


Εικόνα 25: Σύγκριση του ΙΚΑΡΟΣ με το GridFTP, σε παγκόσμια κλίμακα

6.2 Πειραματικά αποτελέσματα από την χρήση των μηχανισμών μεταφοράς δεδομένων στο δίκτυο κατανομής δεδομένων του Πειράματος CMS του LHC στο CERN

Σε αυτήν την ενότητα παρουσιάζεται το παγκόσμιας κλίμακας δίκτυο κατανομής δεδομένων του πειράματος CMS του LHC στο CERN (εικόνα 26) PhedEx [35] σε σύγκριση με το ΙΚΑΡΟΣ. Με την χρήση του ΙΚΑΡΟΣ δομήθηκε ένα αποθηκευτικό σύστημα που μπορεί να ανταποκριθεί στις τεράστιες ανάγκες του πειράματος με το 1/5 του κόστους κτήσης και το 1/3 της κατανάλωση σε ηλεκτρική ενέργεια σε σχέση με ένα τυπικό σύστημα.

Το δίκτυο κατανομής δεδομένων του πειράματος ακολουθεί ιεραρχική δομή. Tier 0 είναι ο επιταχυντής στο CERN, ο οποίος και παράγει τα δεδομένα. Ως Tier 1 κατηγοριοποιούνται οι υπολογιστικές υποδομές στις οποίες γίνεται μαζική αποθήκευση των πρωτογενών δεδομένων. Τέλος ως Tier 2 και Tier 3 κατηγοριοποιούνται οι υπολογιστικές υποδομές στις οποίες γίνεται η τελική επεξεργασία των δεδομένων από τους ερευνητές.



Εικόνα 26:Δίκτυο κατανομής δεδομένων του Πειράματος CMS του LHC στο CERN

Στον πίνακα 3 παρουσιάζεται η μέση απόδοση του ΙΚΑΡΟΣ ως μηχανισμός για την υλοποίηση της συνολικής ροής των δεδομένων από το πείραμα CMS προς την συστοιχία υπολογιστών του ΕΚΕΦΕ Δημόκριτος ZEUS και αντίστροφα. Αρχικά τα δεδομένα μεταφέρονται από μια Tier 1 υποδομή προς τον κεντρικό κόμβο (frontend) του ZEUS και στην συνέχεια από το frontend του ZEUS στο σύστημα αποθήκευσης της συστοιχίας. Η απόδοση του ΙΚΑΡΟΣ συγκρίνεται με τη συνδυασμένη λειτουργία των GridFTP (WAN) και PVFS2 (LAN).

Σε αυτές τις μετρήσεις μεταφέρεται μια συλλογή δεδομένων συνολικού μεγέθους 500 GBs. Η συγκεκριμένη συλλογή αποτελείται από αρχεία των 2 GB. Είναι φανερό ότι η άρση των ενδιάμεσων επιπέδων συγχρονισμού που επιτυγχάνεται με την υλοποίηση της αρχιτεκτονικής που εισάγει το ΙΚΑΡΟΣ υπερτερεί ξεκάθαρα σε σχέση με την αποσπασματική προσέγγιση που προσφέρει ο συνδυασμός του GridFTP με το PVFS2.

Λόγω της συνδυασμένης χρήσης των GridFTP και PVFS2 υπάρχουν περιπτώσεις όπου, ανάλογα με την υλοποίηση, το ένα σύστημα προκαλεί συμφόρηση στο άλλο

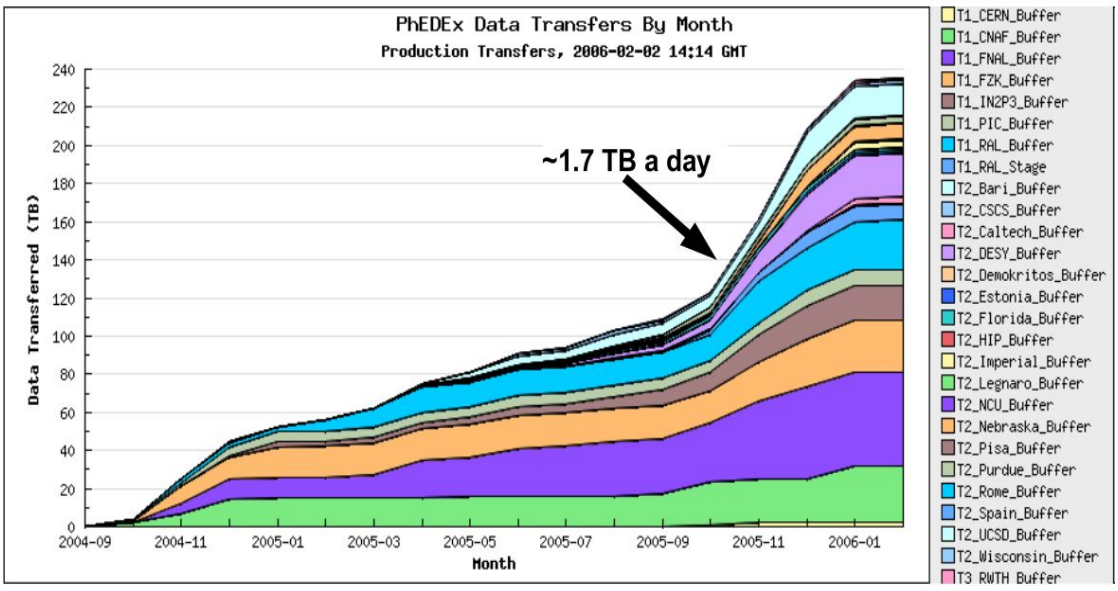
περιορίζοντας έτσι την συνολική απόδοση. Το γεγονός ότι το PVFS2 δεν μπορεί να εκμεταλλευτεί το διαθέσιμο εύρος ζώνης για τις διεργασίες εγγραφής, όπως αποδείχθηκε στην προηγούμενη ενότητα, αποτελεί ένα ακόμα σημαντικό εμπόδιο στην συνολική ροή των δεδομένων.

Πίνακας 3: Συνολική ροή δεδομένων (WAN-LAN)

Συνολική ροή δεδομένων (WAN- LAN)	ΙΚΑΡΟΣ (MB/s)	GridFTP + PVFS2 (MB/s)
Μεταφορά συλλογής δεδομένων 500 GB, του πειράματος CMS	68,92	36,4

Το ΙΚΑΡΟΣ παρέχει άμεση πρόσβαση σε κάθε I/O κόμβο, ανεξάρτητα από την ιεραρχία, με αποτέλεσμα να μπορεί να διαχειρίζεται την συνολική ροή των δεδομένων (τοπική και απομακρυσμένη πρόσβαση) κυρίως στο επίπεδο του δικτύου. Τα άλλα συστήματα είναι απομονωμένα μεταξύ τους και αναγκάζονται να εκτελούν όλες τις διεργασίες του μεταξύ τους συντονισμού στο λειτουργικό σύστημα, με αποτέλεσμα να μην μπορούν να επιτύχουν την απόδοση του ΙΚΑΡΟΣ στην συνολική ροή των δεδομένων.

Στις εικόνες 27 και 28 παρουσιάζονται οι απαιτήσεις για τις μεταφορές δεδομένων στο πείραμα του CMS όπως αυτές αποτυπώθηκαν στις τελικές δοκιμές του πειράματος το έτος 2005, καθώς και οι ρυθμοί μεταφοράς των δεδομένων σε πραγματικές συνθήκες κατά την έναρξη της λειτουργία του πειράματος το έτος 2013.

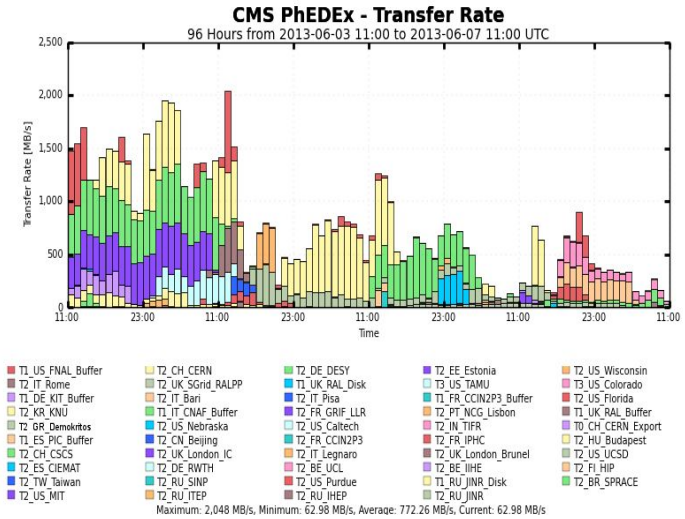


Εικόνα 27: Όγκος δεδομένων του πειράματος, τελικές δοκιμαστικές μετρήσεις (2005-2006)

Στις τελικές δοκιμές του πειράματος το έτος 2005 ο στόχος ήταν να επιτευχθεί ρυθμός μεταφοράς δεδομένων από τα υπολογιστικά κέντρα Tier 1 σε κάθε Tier 2 υποδομή (ΕΚΕΦΕ Δημόκριτος) περίπου 1.7 TB ανά μέρα. Στην εικόνα 27 αποτυπώνεται η επίτευξη του συγκεκριμένου ρυθμού μεταφοράς των δεδομένων.

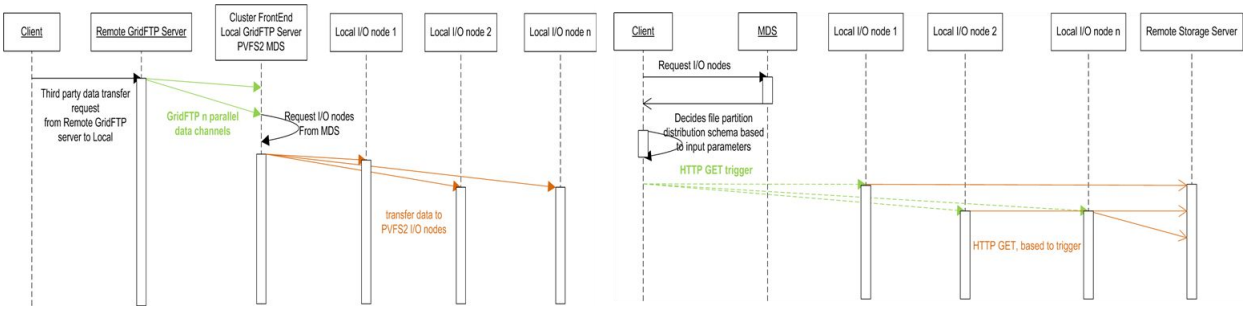
Για την λειτουργία του πειράματος σε πραγματικό περιβάλλον παραγωγής και όχι απλά σε επίπεδο δοκιμών θα έπρεπε να διατηρηθεί ο συγκεκριμένος ρυθμός αλλά και ταυτόχρονα να αυξηθεί η συνολική αποθηκευτική ικανότητα που συνεισέφερε το ΕΚΕΦΕ Δημόκρητος προς το πείραμα από μόλις 2TB (έτος 2005) σε περισσότερα από 20 TB. Στην εικόνα 28 παρουσιάζονται οι ρυθμοί μεταφοράς των δεδομένων σε πραγματικές συνθήκες κατά την λειτουργία του πειράματος το έτος 2013.

Ο ελάχιστος ρυθμός μεταφοράς στην εικόνα 28 είναι τα 68,92 MB/s που ουσιαστικά είναι και ο ρυθμός μεταφοράς δεδομένων στον οποίον λειτουργούν οι περισσότερες Tier 2 υποδομές, μια εκ των οποίων είναι και αυτή που λειτουργεί στο ΕΚΕΦΕ Δημόκριτος.



Εικόνα 28:Ρυθμός μεταφοράς δεδομένων σε πραγματικές συνθήκες (2013)

Στο σχήμα 21 αναλύεται η συνολική ροή των δεδομένων του συνδυασμού των GridFTP+PVFS2 (αριστερά) και του ΙΚΑΡΟΣ (δεξιά). Στη πρώτη περίπτωση δεν είναι εφικτό να διασφαλιστεί το ίδιο μέγεθος stripe και stripe mapping με αποτέλεσμα να εμφανίζεται το φαινόμενο του ανταγωνισμού πόρων μεταξύ των δικτυακών και αποθηκευτικών μέσων και να μειώνεται δραματικά η I/O απόδοση. Στη περίπτωση του ΙΚΑΡΟΣ, εξαιτίας της χρήσης της reversed read τεχνικής, εφαρμόζονται συντονισμένες παράλληλες μεταφορές δεδομένων με αποτέλεσμα τη δραστική μείωση των προηγούμενων φαινομένων ανταγωνισμού για πόρους και τη βελτίωση της απόδοσης [84].



Σχήμα 21: Ανάλυση συνολικής ροής δεδομένων

Γίνεται αντιληπτό πως με τη χρήση των μηχανισμών που παρέχει το πλαίσιο ΙΚΑΡΟΣ το ΕΚΕΦΕ Δημόκριτος κατάφερε να αντεπεξέλθει στις απαιτήσεις του πειράματος. Εκμεταλλεύτηκε πλήρως την δικτυακή διασύνδεση ευρείας περιοχής που διαθέτει με ελάχιστο οικονομικό κόστος και εξαιρετικά χαμηλή κατανάλωση ενέργειας.

7. ΙΚΑΡΟΣ, ΕΝΑ ΠΛΑΙΣΙΟ ΔΗΜΙΟΥΡΓΙΑΣ ΔΥΝΑΜΙΚΩΝ ΑΠΟΘΗΚΕΥΤΙΚΩΝ ΣΧΗΜΑΤΙΣΜΩΝ

Η ικανότητα του ΙΚΑΡΟΣ να χρησιμοποιεί έναν μεγάλο αριθμό I/O κόμβων με χαμηλές τεχνικές προδιαγραφές και χαμηλή κατανάλωση ενέργειας καθώς και η ευελιξία που διαθέτει να αναπροσαρμόζεται δυναμικά και on demand, βασιζόμενο στα αιτήματα των χρηστών ή των εφαρμογών, επιτρέπει να ανάπτυξη μεθοδολογιών που οδηγούν στην επίλυση των προβλημάτων που αντιμετωπίζουν τα υπάρχοντα διαμοιραζόμενα συστήματα αρχείου. Τα βασικότερα προβλήματα που ανακύπτουν όταν αυτά εφαρμόζονται σε μεγάλης κλίμακας συστήματα έχουν ως ακολούθως:

1. Το εύρος ζώνης (I/O και δίκτυο) δεν κλιμακώνει οικονομικά σε αναλογία με την κλιμάκωση της διαθέσιμης χωρητικότητας σε μεγάλης κλίμακας συστήματα.
2. Η I/O κίνηση στην δικτυακή υποδομή μπορεί να επηρεαστεί από άλλες μη σχετικές διεργασίες ή αντίστοιχα να επηρεάσει την απόδοση των άλλων διεργασιών.
3. Η I/O κίνηση στην αποθηκευτική υποδομή μπορεί να επηρεαστεί από άλλες μη σχετικές διεργασίες ή αντίστοιχα να επηρεάσει την απόδοση των άλλων διεργασιών.

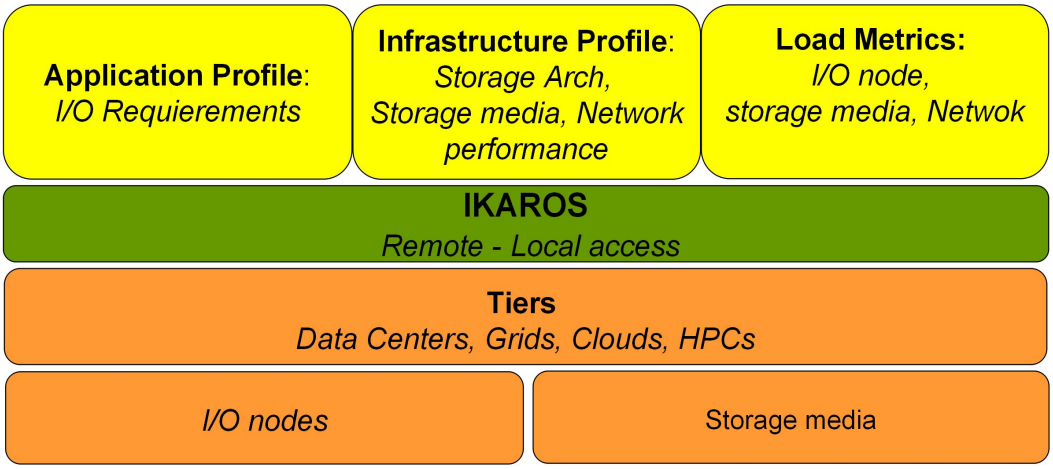
Στα προηγούμενα κεφάλαια παρουσιάστηκαν λύσεις που επικεντρώνονται κυρίως στη διευθέτηση των 1 και 2. Σε αυτό το κεφάλαια παρουσιάζονται οι δυνατότητες που παρέχει το ΙΚΑΡΟΣ για την αντιμετώπιση του 3ου.

7.1 Τεχνικά χαρακτηριστικά του ΙΚΑΡΟΣ

Το ΙΚΑΡΟΣ δομήθηκε ως ένα “λεπτό” στρώμα που έχει τη δυνατότητα να προσφέρει υπηρεσίες σε πολλαπλά επίπεδα. Το μοντέλο ανάπτυξης (Deployment model) του ΙΚΑΡΟΣ επιτρέπει το διαχωρισμό των υπολογιστικών από τους αποθηκευτικούς πόρους χωρίς όμως να αποτρέπει το αντίθετο, σε περίπτωση που αυτό μπορεί να θεωρηθεί επωφελές. Έχει τη δυνατότητα να εκτελεί διεργασίες εγγραφής σε διαφορετικές περιοχές του αρχείου, επιτρέποντας και τις ταυτόχρονες (διαμοιραζόμενες) εγγραφές (Concurrent-shared writes).

Οι διαθέσιμοι μηχανισμοί του ΙΚΑΡΟΣ επιτρέπουν στους χρήστες και στις εφαρμογές να γνωρίζουν την ακριβή απεικόνιση των κομματιών, που αποτελούν ένα αρχείο (Data layout), και ταυτόχρονα παρέχει έναν πολύ μεγάλο αριθμό παραμετροποιήσεων με δυναμικό τρόπο. Οι συγκεκριμένοι μηχανισμοί παρέχουν σημαντικά πλεονεκτήματα στην αντιμετώπιση του 3ου προβλήματος.

Μέσω των αποφάσεων για το εκάστοτε Data Layout, Το ΙΚΑΡΟΣ παρέχει στο χρήστη την άμεση διαχείριση των πόρων (I/O κόμβοι, αποθηκευτικά μέσα) ανεξάρτητα από το επίπεδο που αυτοί ενεργούν. Αυτή η λογική οδηγεί στη βέλτιστη διαχείριση τους με σκοπό την επίτευξη της μέγιστης δυνατής I/O απόδοσης. Σε αυτό συντελεί και το γεγονός ότι παρέχεται η δυνατότητα εισαγωγής στο ΙΚΑΡΟΣ δεδομένων που αφορούν το προφίλ της εφαρμογής, τις I/O απαιτήσεις, το φορτίο του δικτύου και των κόμβων όπως και το προφίλ της αποθηκευτικής υποδομής (Σχήμα 22).



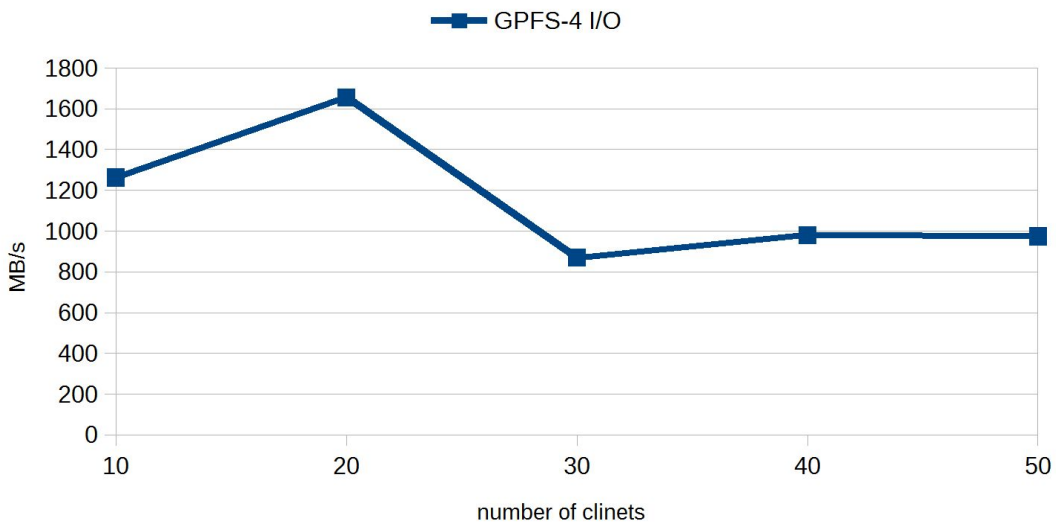
Σχήμα 22: Το πλαίσιο ΙΚΑΡΟΣ

7.2 Ανάλυση των χαρακτηριστικών της Cytera HPC υποδομής

Είναι φανερό πως για να υποβληθούν οι κατάλληλες μετρήσεις και να εξαχθούν ασφαλή συμπεράσματα θα πρέπει πρώτα να γίνει λεπτομερής ανάλυση των χαρακτηριστικών της υπολογιστικής υποδομής στην οποία θα πραγματοποιηθούν οι μετρήσεις. Η υποδομή που χρησιμοποιείται σε αυτήν την ενότητα είναι η Cytera HPC μηχανή που βρίσκεται στην Κύπρο και αποτελεί περιφερειακή υποδομή για την Νοτιοανατολική Μεσόγειο.

Το μηχάνημα αποτελείται από 100 κόμβους, 96 υπολογιστικούς και 4 αποθηκευτικούς. Κάθε κόμβος διαθέτει 12 επεξεργαστικούς πυρήνες, 48 GBs RAM και 15k RPM HDD. Οι κόμβοι διασυνδέονται μεταξύ τους χρησιμοποιώντας δικτυακή υποδομή τύπου QDR (40 Gbit/s) infiniband. Το σύστημα αρχείου χρησιμοποιεί το GPFS και υλοποιείται από τους 4 αποθηκευτικούς κόμβους. Το σύστημα αποθήκευσης διαθέτει 360 TBs σε δίσκους σε μία διάταξη 18 RAID 6 συστοιχιών με 10 σκληρούς δίσκους σε κάθε συστοιχία. Ο μηχανισμός των μεταδεδομένων στο συγκεκριμένο GPFS σύστημα παρέχεται από μια διάταξη 4 RAID 10 συστοιχιών μοιρασμένων στους 4 αποθηκευτικούς κόμβους. Για την υλοποίηση του ΙΚΑΡΟΣ χρησιμοποιήθηκαν οι υπολογιστικούς κόμβους του μηχανήματος και οι τοπικοί δίσκοι που υπάρχουν σε καθένα από αυτούς τους κόμβους (1 HDD/κόμβο).

Για ένα δοσμένο GPFS σύστημα οι πιο σημαντικοί παράγοντες που επηρεάζουν την απόδοση του (εκτός από τα μοτίβα I/O πρόσβασης) είναι ο αριθμός των παράλληλων διεργασιών που συμμετέχουν στην μεταφορά καθώς και το μέγεθος των ανεξάρτητων μεταφορών. Από την εικόνα 29 γίνεται αντιληπτό ότι η μέγιστη απόδοση επιτυγχάνεται όταν ο λόγος των διεργασιών τύπου πελάτη προς τους διαθέσιμους I/O κόμβους είναι κοντά στο 5:1. Αυτός ο λόγος είναι ελάχιστα βελτιωμένος αλλά πολύ κοντά στον λόγο 4:1 που μετρήθηκε στο Lawrence Livermore National Lab το 2000 [83] χρησιμοποιώντας 38 αποθηκευτικούς κόμβους και 152 διεργασίες τύπου πελάτη.



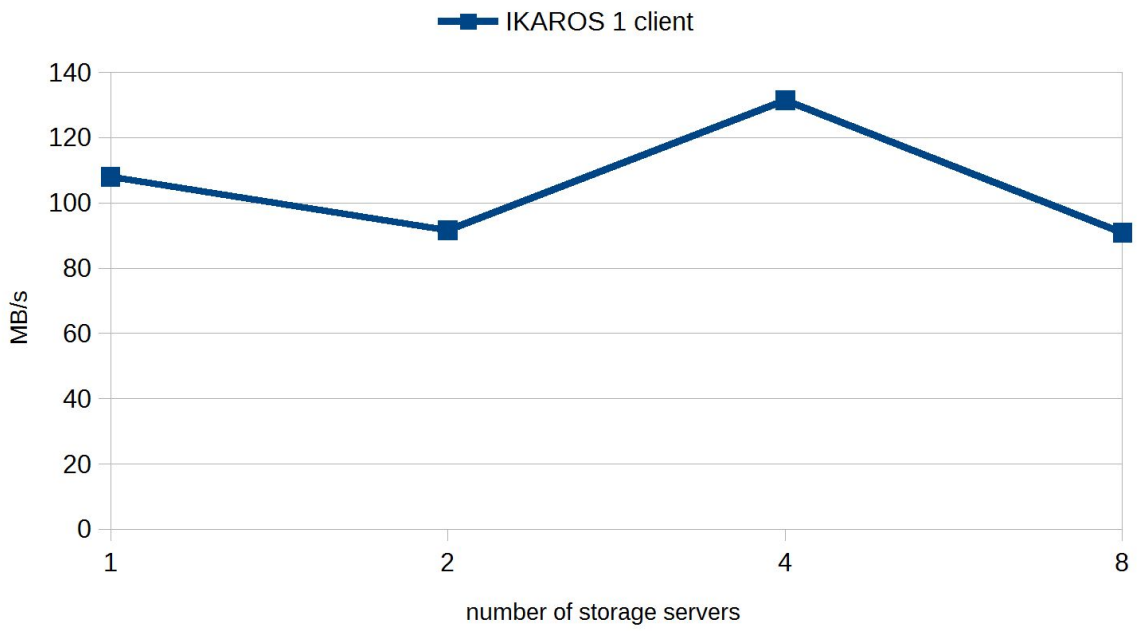
Εικόνα 29: Απόδοση GPFS σε σχέση με τον αριθμό των πελατών

Στην εικόνα 29 [84] παρουσιάζεται η απόδοση του GPFS στη Cytera μηχανή κατά την εγγραφή αρχείου μεγέθους 80GB. Το αρχείο διαμοιράζεται σε επιμέρους ξεχωριστά αρχεία στους διαφορετικούς κόμβους οι οποίοι συμμετέχουν στην μεταφορά. Όπως αναφέρεται στο [83] όταν ο λόγος πελάτη:εξυπηρετητή είναι αρκετά χαμηλός οι αποθηκευτικοί κόμβοι υποχρησιμοποιούνται, όταν πάλι είναι αρκετά υψηλός μπορεί να εμφανιστούν σημαντικά προβλήματα όπως η απώλεια πακέτων με αποτέλεσμα την τελική μείωση της απόδοσης.

Όπως προαναφέρθηκε, το συγκεκριμένο μηχάνημα διαθέτει 180 σκληρούς δίσκους κατανεμημένους στους 4 αποθηκευτικούς κόμβους σε 18 συνολικά RAID 6 διατάξεις. Αυτή η παραμετροποίηση μπορεί θεωρητικά να επιτύχει ρυθμοαπόδοση για τις διεργασίες εγγραφής που να αγγίζει τα 4200 MB/s. Η θεωρητική αυτή τιμή η είναι σημαντικά υψηλότερη από την μέγιστη απόδοση των 1600 MB/s που επιτυγχάνει το μηχάνημα με την χρήση του GPFS.

Στην εικόνα 30 [84] παρουσιάζεται η απόδοση της διεργασίας εγγραφής ενός αρχείου 80 GBs από έναν υπολογιστικό κόμβο (χρήση του ενός τοπικού δίσκου) σε 1,2,4 και 8 υπολογιστικούς κόμβους με αντίστοιχη χρήση σκληρών δίσκων. Για όλες τις μεταφορές χρησιμοποιούνται μόνο οι σκληροί δίσκοι που βρίσκονται τοπικά στους υπολογιστικούς κόμβους και συμμετέχουν στην συγκεκριμένη ενέργεια. Όπως έχει αναφερθεί υπάρχει μόνο ένας τοπικός δίσκος σε κάθε υπολογιστικό κόμβο.

Γίνεται αντιληπτό πως σε αυτό το σενάριο η διεργασία τύπου πελάτη εγγράφει το αρχείο σε ξεχωριστά κομμάτια σε 1-8 κόμβους. Ο σκοπός αυτής της μέτρησης είναι να υπολογιστεί το βέλτιστο φορτίο υπό την μορφή αιτημάτων ανάγνωσης/εγγραφής που οι συγκεκριμένοι σκληροί δίσκοι μπορούν να διαχειριστούν.



Εικόνα 30: Απόδοση τοπικού δίσκου σε σχέση με τα αιτήματα ανάγνωσης/εγγραφής

Από την εικόνα 30 [84] γίνεται φανερό ότι αν το αρχείο βρίσκεται σε ένα σκληρό δίσκο και ξεπερνάει το μέγεθος της RAM, κάτι που συμβαίνει πολύ συχνά στις εφαρμογές που δραστηριοποιείται το ΙΚΑΡΟΣ, τότε για να επιτευχθεί η βέλτιστη κατανομή του αρχείου θα πρέπει να ακολουθηθεί ο λόγος 4:1. Δηλαδή τέσσερις σκληροί δίσκοι για κάθε διεργασία εγγραφής. Στις πειραματικές μετρήσεις που ακολουθούν φαίνεται πώς αν τηρηθεί αυτός ο λόγος είναι εφικτό να γίνει πλήρης εκμετάλλευση του διαθέσιμου εύρους ζώνης, σε επίπεδο I/O και δικτύου.

7.3 Πειραματικές μετρήσεις στη Cytera HPC υποδομή

Τα πειράματα που πραγματοποιούνται σε αυτήν την ενότητα έχουν επιλεγεί έτσι ώστε να καταδειχθούν οι επιπτώσεις που υπάρχουν από τις τυχόν διαφοροποιήσεις στα I/O χαρακτηριστικά των εφαρμογών. Σε αυτά τα πειράματα παρατηρείται πως η συνολική ρυθμοαπόδοση μεταβάλεται σε σχέση με τον αριθμό των διεργασιών τύπου πελάτη και το μέγεθος των ανεξάρτητων μεταφορών. Επίσης φαίνεται πως η κλιμάκωση της απόδοσης του GPFS και του ΙΚΑΡΟΣ εξαρτάται από το μέγεθος του συστήματος καθώς και από την συνολική ρυθμοαπόδοση των παράλληλων διεργασιών που χρησιμοποιούνται για την εγγραφή ενός μεγάλου αρχείου.

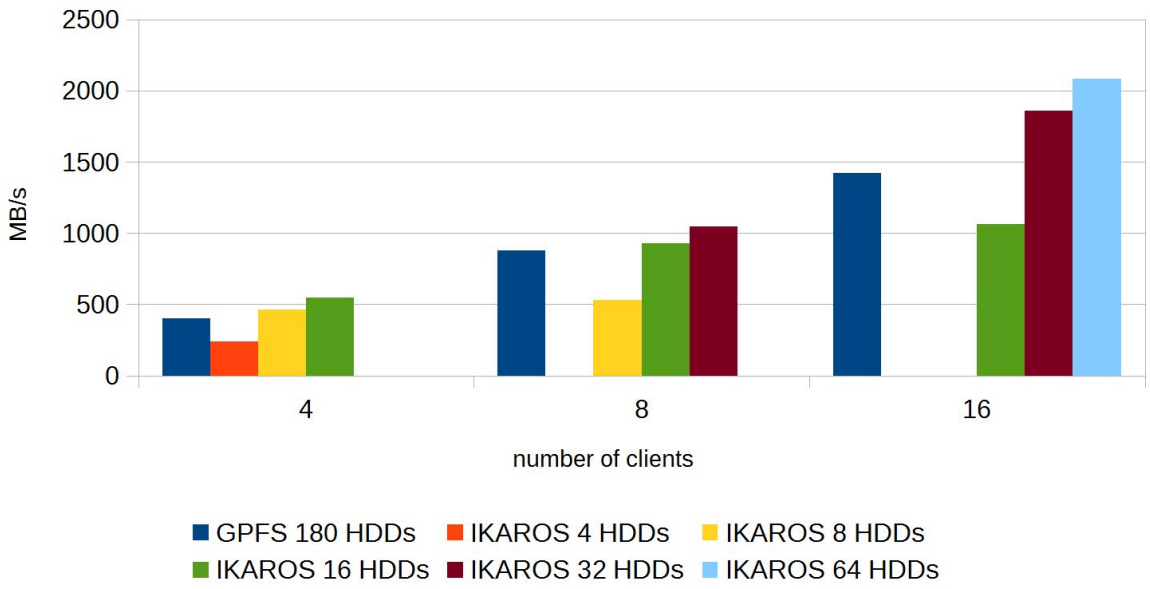
Για τη μέτρηση της ρυθμοαπόδοσης των εγγραφών χρησιμοποιείται ένα “barrier” όπου κάθε διεργασία καταγράφει ένα “wall clock” ως χρόνο έναρξης, στην συνέχεια η διεργασία 0 δημιουργεί το αρχείο και όλες οι άλλες διεργασίες περιμένουν μέχρι το συγκεκριμένο “barrier” πριν αρχίσουν την προσπέλαση του αρχείου. Οι διεργασίες εγγράφουν τα δεδομένα τους σύμφωνα με τα επιλεγμένα χαρακτηριστικά της εφαρμογής. Η κάθε εγγραφή γίνεται ανεξάρτητα η μία από την άλλη χωρίς να δημιουργούνται κενά στο αρχείο. Τελικά όλες οι διεργασίες κλείνουν το αρχείο και καταγράφουν τον χρόνο που διήρκεσε η όλη ενέργεια.

Η ρυθμοαπόδοση υπολογίζεται ως ο συνολικός αριθμός των bytes που εγράφησαν κατά την διάρκεια του συνολικού χρόνου που απαιτήθηκε για την ολοκλήρωση της διαδικασίας. Ο τελευταίος χρόνος που μετρήθηκε μείον τον χρόνο έναρξης. Αυτή η προσέγγιση είναι αρκετά συντηρητική αλλά πλεονεκτεί στο γεγονός ότι περιλαμβάνει το συνολικό overhead από την έναρξη μέχρι και τον τερματισμού της όλης διαδικασίας και

έτσι μετράται η πραγματική συνολική ρυθμοαπόδοση και όχι ο μέσος όρος ανα διεργασία.

Τα αποτελέσματα για το GPFS παρουσιάζουν την μέγιστη πιθανή απόδοση του συστήματος αρχείου και όχι απαραίτητα την πραγματική απόδοση που ένας χρήστης μπορεί να λάβει. Για το GPFS, άλλες διεργασίες που ανταγωνίζονται για τους ίδιους πόρους μπορεί να επηρεάσουν την I/O απόδοση. Το ΙΚΑΡΟΣ επιτρέπει την δημιουργία παράλληλων αιτημάτων τύπου πελάτη προς την αποθηκευτική υποδομή που με τη σειρά τους δεν θα ανταγωνίζονται για τους ίδιους I/O πόρους. Αυτή η λειτουργία μπορεί να οδηγήσει σε μέγιστη I/O απόδοση για κάθε διεργασία.

Στην εικόνα 31 [84] μετράται η απόδοση του GPFS και του ΙΚΑΡΟΣ χρησιμοποιώντας έως και 16 διεργασίες τύπου πελάτη, ταυτόχρονα. Οι μετρήσεις στην εικόνα 29 δείχνουν ξεκάθαρα ότι η απόδοση του GPFS στο συγκεκριμένο μηχάνημα επηρεάζονται σημαντικά μετά από αυτό το όριο. Το ΙΚΑΡΟΣ χρησιμοποιεί τους ακόλουθους λόγους HDD/client 4:4, 4:2 και 4:1. Με αυτόν τον τρόπο δημιουργούνται απομονωμένοι αποκλειστικοί αποθηκευτικοί σχηματισμοί (4:1) ή ημιαποκλειστικοί (4:2, 4:4) για κάθε διεργασία τύπου πελάτη. Έτσι οι διεργασίες εγγραφής δεν ανταγωνίζονται για πόρους, σε αυτήν την περίπτωσή για σκληρούς δίσκους [84].



Εικόνα 31:ΙΚΑΡΟΣ vs GPFS

Στην εικόνα 31 φαίνεται καθαρά ότι το ΙΚΑΡΟΣ υπερτερεί του GPFS όταν χρησιμοποιεί τους λόγους 4:2 και 4:1 στην διαχείριση των διαθέσιμων σκληρών δίσκων. Αυτή η προσέγγιση μπορεί να κλιμακώσει με πάνω όριο το διαθέσιμο εύρος ζώνης σε επίπεδο δικτύου έχοντας εκμεταλλευτεί πλήρως την διαθέσιμη I/O υποδομή σε σκληρούς δίσκους, κάτι που δεν επιτυγχάνει το GPFS.

Εδώ θα πρέπει να σημειωθεί ότι στην παρούσα εργασία η εκάστοτε πολιτική επιλογής του κατάλληλου λόγου HDD:client παρέχεται στο σύστημα με μη αυτοματοποιημένο τρόπο. Μελλοντικά θα μπορούσε να αναπτυχθεί κάποιος αυτόματος μηχανισμός που να παρέχει πληροφορίες όπως η I/O κίνηση, η I/O ικανότητα του μηχανήματος, οι I/O απαιτήσεις της εφαρμογής καθώς και το γενικότερο προφιλ της αποθηκευτικής

υποδομής. Με αυτόν τον τρόπο θα υπάρχει η κατάλληλη πληροφορία στούς schedulers, τα load balancing συστήματα ή ακόμα και στο MDS ώστε να λαμβάνονται πιο ακριβείς αποφάσεις και να τροφοδοτούνται αποδοτικότερα υποκείμενα πλαίσια όπως το ΙΚΑΡΟΣ.

8. MOBILE GRID, ΜΙΑ ΠΛΑΤΦΟΡΜΑ ΔΗΜΙΟΥΡΓΙΑΣ ΕΥΡΥΤΕΡΩΝ ΣΥΝΕΡΓΙΩΝ

Όπως έχει προαναφερθεί, οι απαιτήσεις των εφαρμογών που θα υλοποιηθούν με χρονικό ορίζοντα το 2022, όπως αυτές παρουσιάζονται από τις αναφορές των επιστημονικών ομάδων που δραστηριοποιούνται στους τομείς της κλιματολογικής αλλαγής της φυσικής υψηλών ενεργειών και άλλων, κατατείνουν στο ότι οι υπάρχουσες υποδομές δεν θα μπορούν να ανταποκριθούν αν δεν ανασχεδιαστούν. Τεχνικά, αυτό σημαίνει ότι θα πρέπει να εξελιχθούν τα σημερινά Petascale συστήματα σε eXascale. Η ευρύτερη επιστημονική κοινότητα, όπως αυτή εκφράζεται από αναφορές επιστημονικών ομάδων και τεχνικές μελέτες κρατών [41, 45], συγκλίνει στο ότι είναι αναγκαίο να τοποθετηθεί η διαλειτουργικότητα μεταξύ ετερογενών υποδομών σε ένα ευρύτερο πλαίσιο που θα επιτρέπει τη δημιουργία συνεργιών μεταξύ ευρύτερων κοινοτήτων.

Η λογική που εισάγει το ΙΚΑΡΟΣ φιλοδοξεί να ανταποκριθεί στα παραπάνω ζητήματα ως μία πλατφόρμα που θα επιτρέπει ευρύτερες συνέργειες. Το ΙΚΑΡΟΣ προσπαθεί να αναγνωρίσει τις πιθανές ομοιότητες των λειτουργιών μεταξύ των διαφορετικών επιπέδων με σκοπό να διατηρήσει και να επεκτείνει τη λογική της διαλειτουργικότητας μεταξύ των ετερογενών υποδομών. Γίνεται προσπάθεια σε αυτόν τον κατακερματισμό υπηρεσιών και υποδομών να αναγνωριστούν κοινές λειτουργίες και συμπεριφορές που δυνητικά θα μπορεί να οδηγήσουν στη δημιουργία υποδομών που θα διατρέχονται από κοινές πρακτικές και θα μπορούν να υλοποιηθούν με μειωμένο συνολικό κόστος.

Τα τελευταία χρόνια παρατηρείται μια τεράστια αύξηση στο μέγεθος των δεδομένων που παράγονται από μεγάλης κλίμακας συνεργατικές επιστημονικές εφαρμογές, κάτι που πλέον ισχύει και στις εφαρμογές που αναπτύσσονται σε επίπεδο επιχειρήσεων αλλά και σε αυτό του καθημερινού χρήστη. Αυτού του είδους οι εφαρμογές απαιτούν συνεχή αύξηση των διαθέσιμων πόρων. Οι διαθέσιμοι πόροι για τις υπολογιστικές υποδομές παρέχονται, κυρίως, από HPCs, HTCs, συστοιχίες υπολογιστών και Desktop Grids.

Παρατηρείται επίσης μια έντονη προσπάθεια για συνεργασία και συνύπαρξη των παρόχων δικτύων κινητής τηλεφωνίας με τους πάροχους δικτύων δεδομένων. Η προσπάθεια αυτή εκφράζεται, κυρίως, με την παραγωγή συσκευών όπως τα ευφυή κινητά τηλέφωνα (smartphones) και τους προσωπικούς υπολογιστές τύπου ταμπλέτας (tablet PCs). Είναι φανερό ότι οι πάροχοι υπηρεσιών κινητής τηλεφωνίας προσπαθούν να εισέλθουν στον χώρο των δικτύων δεδομένων χρησιμοποιώντας τα χαρακτηριστικά που προσφέρουν αυτές οι συσκευές. Οι δυο αυτές κατηγορίες συσκευών παρέχουν διεπαφές σύνδεσης σε δίκτυα κινητής τηλεφωνίας (3G/4G) αλλά και σε ασύρματα δίκτυα υπολογιστών (WI-FI). Γίνεται αντιληπτό ότι αυτές οι συσκευές μπορούν να μετατραπούν σε πύλες διαλειτουργικότητας μεταξύ των δύο δικτύων.

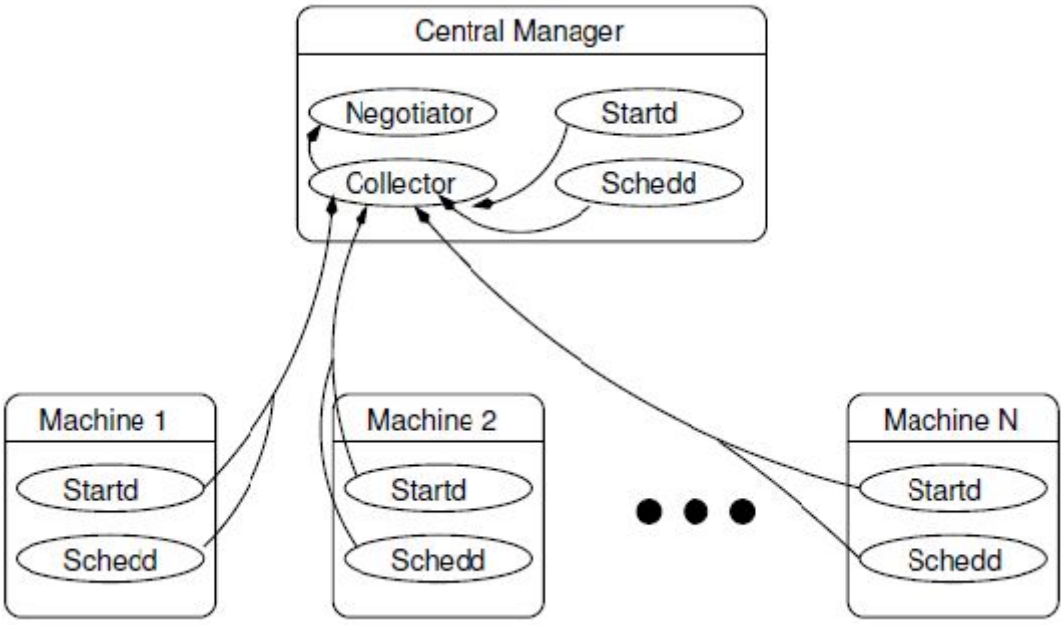
Όμως ένα smartphone ή ένα tablet PC δεν διαθέτει τα κατάλληλα χαρακτηριστικά για να ικανοποιήσει τις ανάγκες μιας διεργασίας που παράγεται από τα επιστημονικά πειράματα που αναφέρονται στην συγκεκριμένη εργασία (CMS-LHC-CERN και KM3NeT). Μια τέτοιου τύπου διεργασία μπορεί να απαιτεί συσκευές που έχουν την δυνατότητα να προσφέρουν τουλάχιστον 2 GB RAM και ρυθμό ρολογιού της CPU στα 2 GHz . Μια τυπική συσκευή κατηγορίας smartphone ή tablet PC μπορεί να διαθέτει CPU με ρυθμό ρολογίου στα 1,2 GHz. Αυτού του τύπου οι συσκευές εισάγουν δύο κατηγορίες περιορισμών:

- Περιορισμούς στις εφαρμογές που έχουν την δυνατότητα να εξυπηρετήσουν. Κυρίως λόγω των χαμηλών τεχνικών δυνατοτήτων και της μικρής αποθηκευτικής ικανότητας που παρέχουν οι μπαταρίες που τροφοδοτούν αυτές τις συσκευές. Η διάρκεια εκτέλεσης των διεργασιών που προέρχονται από τα πιο πάνω πειράματα

ξεπερνά τις 72 ώρες γεγονός απαγορευτικό για χρήση πόρων που έχουν ως βασική πηγή τροφοδοσίας τις μπαταρίες.

- Περιορισμούς στον σχεδιασμό της αρχιτεκτονικής και του μοντέλου αξιοποίησης τους. Κυρίως λόγω της κινητικότητας που εισάγεται από την φύση αυτών των συσκευών. Οι συγκεκριμένοι πόροι αλλάζουν αρκετά συχνά τοπολογία δικτύου (μετάβαση από δίκτυο δεδομένων κινητής τηλεφωνίας σε ασύρματο δίκτυο και αντίστροφά) με αποτέλεσμα να μην μπορούν να υποστηριχθούν με διάφανη τρόπο από τα υπάρχοντα συστήματα δρομολόγησης.

Στην προσπάθειά να επεκταθούν οι υπάρχουσες υπολογιστικές υποδομές ενσωματώνοντας νέες πηγές πόρων όπως συσκευές έξυπνων κινητών τηλεφώνων και προσωπικούς υπολογιστές τύπου ταμπλέτας διαπιστώθηκε ότι δεν θα ήταν αποδοτικό αν απλά επεκτείνονταν μια παραδοσιακή αρχιτεκτονική διαχείρισης πόρων. Τα υπάρχοντα συστήματα, όπως το Condor [6] (σχήμα 23), το Sun Grid Engine (SGE) και το BOINC δεν υποστηρίζουν λειτουργικά συστήματα που χρησιμοποιούνται από τέτοιου είδους συσκευές. Ακόμα όμως και αν συμβεί κάτι τέτοιο στο μέλλον τα συστήματα αυτά δεν θα είναι αποδοτικά στην ενσωμάτωση τέτοιου τύπου συσκευών αν δεν μεταβάλλουν δραστικά την αρχιτεκτονική τους. Τα συστήματα αυτά απαιτούν διαδικασίες, όπως το ταίριασμα των απαιτήσεων με τους διαθέσιμους πόρους (matchmaking), που είναι αρκετά πολύπλοκες και κοστοβόρες για τέτοιου τύπου συσκευές. Την ίδια στιγμή δεν είναι σε θέση να αντιμετωπίσουν την υψηλή κινητικότητα που αυτές εισάγουν.



Σχήμα 23: Condor, μηχανισμοί δρομολόγησης

Η αρχιτεκτονική που εισάγει η έννοια του Mobile Grid (στην παρούσα εργασία) στοχεύει στο να δημιουργήσει μια πύλη διαλειτουργικότητας μεταξύ των υφιστάμενων υπολογιστικών υποδομών, των ασυρμάτων δικτύων και των δικτύων κινητής τηλεφωνίας. Στοχεύοντας στη δημιουργία νέων πιθανών συνεργιών με τους παρόχους δικτύων κινητής τηλεφωνίας μέσω της χρήσης των πόρους που αυτά διαθέτουν. Με αυτό το τρόπο μπορεί να αυξηθεί το εύρος των συνεργιών και πιθανά να ενισχυθεί το όραμα της υλοποίησης των eXascale υποδομών.

Το άμεσο αποτέλεσμα αυτής της συνύπαρξης και συνεργασίας θα είναι η επέκταση των δυνατοτήτων της υπάρχουσας παγκόσμιας υπερυπολογιστικής υποδομής και η ενσωμάτωση σε αυτήν νέου τύπου πόρων, υπηρεσιών και διεπαφών. Μέχρι τώρα, η

έννοια του Mobile Grid περιορίζεται στην χρήση της “πληροφορίας” ως πόρου που διαμοιράζεται (Akogrimo [37]) ή στην χρήση συσκευών τύπου laptop και PDA για την παροχή υπολογιστικών κύκλων [38].

8.1 Μηχανισμοί Διαχείρισης Υπολογιστικών Πόρων στο ΙΚΑΡΟΣ

Τα υπάρχοντα κατανεμημένα συστήματα κλιμακώνονται σε όλο και μεγαλύτερα μεγέθη με αποτέλεσμα να μειώνεται συνεχώς η δυνατότητα αποδοτικού ελέγχου ή πολλές φορές να υπάρχει αδυναμία ακόμα και στη περιγραφή τους. Τα παγκόσμιας κλίμακας κατανεμημένα συστήματα είναι ετερογενή με κάθε έννοια. Αποτελούνται από διαφορετικού τύπου υλικό που παρέχεται από διαφορετικούς κατασκευαστές, χρησιμοποιούν διαφορετικά λειτουργικά συστήματα και εφαρμογές ενώ διασυνδέονται μέσω μη αξιόπιστων δικτύων και αλλάζουν συνεχώς ρυθμίσεις.

Επιπροσθέτως, έχουν πολλούς ιδιοκτήτες και διαχειριστές, διαφορετικές πολιτικές διαχείρισης και απαιτήσεις βάση των οποίων συμμετέχουν στην ευρύτερη κοινότητα. Σε ένα τόσο πολύπλοκο περιβάλλον η δημιουργία ευέλικτων μηχανισμών αποτελεί το στοιχείο που μπορεί να προσφέρει μια αποδοτικότερη αξιοποίηση αυτών των υποδομών [5] [8].

Τα παραδοσιακά συστήματα διαχείρισης πόρων βασιζόμενα σε μηχανισμούς χειρισμού διεργασιών σε επίπεδο δέσμης (batch job systems) υλοποιούν μηχανισμούς αντιστοίχισης και διεκπεραίωσης διεργασιών που μπορούν να σταχυολογηθούν ως ακολούθως [9] :

- Οι πράκτορες (agents) και οι πόροι διαφημίζουν τα χαρακτηριστικά και τις απαιτήσεις τους.
- Ο μηχανισμός αντιστοίχισης αναζητά ανάμεσα στα γνωστά χαρακτηριστικά και τις απαιτήσεις για πόρους και δημιουργούν ζεύγη που ικανοποιούν τους περιορισμούς και τις προτιμήσεις, εκατέρωθεν.
- Ο μηχανισμός αντιστοίχισης ενημερώνει τα δυο μέρη για την απόφαση αντιστοίχισης.
- Ο μηχανισμός αντιστοίχισης παύει την λειτουργία του, οι πράκτορες και οι πόροι που έχουν αμφότεροι αντιστοιχηθεί μεταξύ τους εγκαθιστούν επαφή πιθανόν διαπραγματεύονται περαιτέρω όρους και τέλος συνεργάζονται ώστε να διεκπεραιωθεί η διαδικασία εκτέλεσης της διεργασίας.

Στη δεδομένη περίπτωση, ο βέλτιστος τρόπος για την ένταξη, με διαφανή τρόπο, των πόρων που φιλοξενούνται στα τηλεπικοινωνιακά δίκτυα στην υπάρχουσα υπολογιστική υποδομή θα είναι η δημιουργία μιας πύλης διαλειτουργικότητας μεταξύ των διαφορετικών υποδομών. Για τον παραπάνω λόγο υλοποιήθηκε το σύστημα διαχείρισης υπολογιστικών πόρων του ΙΚΑΡΟΣ που λειτουργεί ως υπηρεσία για τις υπολογιστικές υποδομές και ως πάροχος περιεχομένου για τα δίκτυα κινητής τηλεφωνίας.

Με την δημιουργία του ΙΚΑΡΟΣ προσδοκάται μια αποδοτικότερη χρήση της υπάρχουσας υποδομής δημιουργώντας ένα πλαίσιο διαχωρισμού αρμοδιοτήτων. Για να επιτευχθεί αυτό η υπάρχουσα υποδομή εκτελεί υπηρεσίες και μηχανισμούς που βρίσκονται στον πυρήνα υλοποίησης της αρχιτεκτονικής όπως η δρομολόγηση εργασιών και ανεύρεση των πόρων.

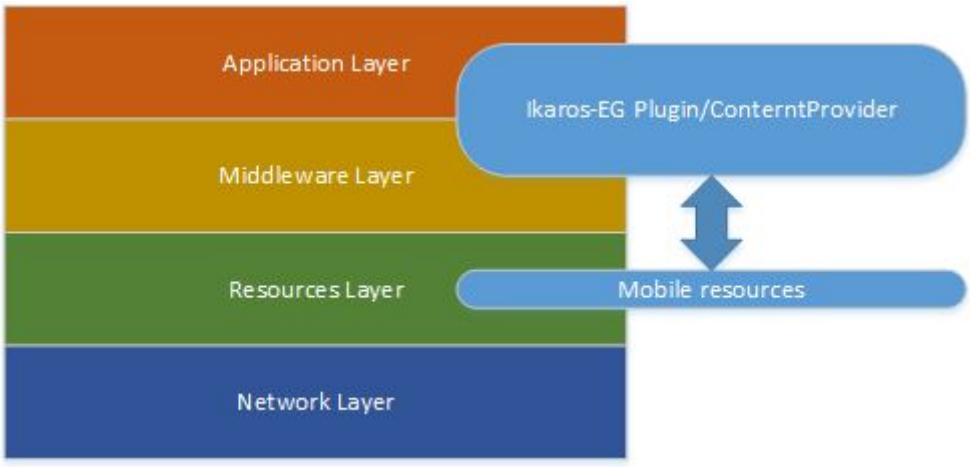
Τα υπόλοιπα χαρακτηριστικά και λειτουργίες θεωρούνται επιπρόσθετες και υλοποιούνται χρησιμοποιώντας τους νέους πόρους που διατίθενται από τα δίκτυα κινητής τηλεφωνίας και ασύρματης δικτύωσης. Όπως γίνεται αντιληπτό οι νέες πηγές πόρων δεν θα χρησιμοποιούνται για την εκτέλεση διεργασιών που υποβάλλονται από εφαρμογές ή πειράματα αλλά για την υλοποίηση των μηχανισμών και των υπηρεσιών

της ίδιας της υποδομής, συνυπολογίζοντας πάντοτε την φύση των συσκευών που καλούνται να διεκπεραιώσουν αυτές τις λειτουργίες.

Γίνεται εύκολα αντιληπτό ότι υπάρχουν διεργασίες που ουσιαστικά υλοποιούν επιπρόσθετες λειτουργίες σε μια υποδομή. Αυτές οι διεργασίες έχουν χαμηλές υπολογιστικές ανάγκες και ταυτόχρονα δεν απαιτούν μεγάλο έλεγχο στους πόρους. Αυτού του τύπου οι διεργασίες μπορούν να θεωρηθούν ως 'αναλώσιμες' και δείχνουν μεγάλη "ανθεκτικότητα" στα ιδιαίτερα χαρακτηριστικά που εισάγουν αυτού του τύπου οι πόροι. Έτσι μπορούν δυνητικά να προέρχονται από μηχανισμούς υλοποίησης στατιστικών στοιχείων, καταγραφής και χρέωσης, αναζήτησης καθώς και περαιτέρω μορφοποίησης δεδομένων και μεταδεδομένων.

Με αυτόν τρόπο μπορεί να αποσυμφορηθεί η υποδομή επιτυγχάνοντας ταυτόχρονα αποδοτικότερη κλιμάκωση. Σε μία υποδομή παγκόσμιας εμβέλειας τέτοιου είδους διεργασίες μπορεί να είναι χιλιάδες μέσα σε πολύ μικρό χρονικό διάστημα. Το σύστημα διαχείρισης υπολογιστικών πόρων του ΙΚΑΡΟΣ δεν στοχεύει στο να υποκαταστήσει το υπάρχον σύστημα διαχείρισης πόρων μιας υποδομής. Σκοπός είναι να ενεργεί ως μια πύλη διαλειτουργικότητας μεταξύ των ξεχωριστών υποδομών (υπερυπολογιστικων υποδομών, δικτύων κινητής τηλεφωνίας και ασυρμάτων δικτύων) παρέχοντας υπηρεσίες διαχείρισης για τις νέες πηγές πόρων που ανακύπτουν προσφέροντας οφέλη σε όλες τις έως τώρα ξεχωριστές υποδομές. Η διαλειτουργικότητα επιτυγχάνεται με την χρήση προτύπων όπως το HTTP.

Μια υπολογιστική υποδομή μπορεί να αναπαρασταθεί ως μια σειρά από επίπεδα. Στην εικόνα 32 διακρίνεται το επίπεδο του δικτύου, των πόρων, του μεσισμικού και τέλος των εφαρμογών. Οι πόροι είναι υπολογιστικές και αποθηκευτικές μονάδες, μνήμες τυχαίας προσπέλασης (RAM) ακόμα και ανιχνευτές. Σε αυτήν τη λογική το σύστημα ΙΚΑΡΟΣ μπορεί να τοποθετηθεί κυρίως στο επίπεδο του μεσισμικού αλλά και σε αυτό των εφαρμογών [14], καθώς λειτουργεί ως υπηρεσία για τις υπερκείμενες εφαρμογές αλλά ταυτόχρονα μπορεί να ενεργεί και ως μία εφαρμογή, ή μέρος εφαρμογής, που χρησιμοποιεί την υποδομή.



Εικόνα 32: Η έννοια του Mobile Grid

Όπως φαίνεται από την εικόνα 32 το σύστημα διαχείρισης υπολογιστικών πόρων του ΙΚΑΡΟΣ λειτουργεί ως πάροχος περιεχομένου για συσκευές τύπου smartphone και tablet PC. Κατά αυτόν τον τρόπο οι συσκευές αυτές επικοινωνούν με το σύστημα ΙΚΑΡΟΣ από όπου λαμβάνουν τις υπάρχουσες, προς εκτέλεση, διεργασίες και επιστρέφουν στο σύστημα τα αποτελέσματα. Έτσι το σύστημα δεν χρειάζεται να

γνωρίζει τις συνεχείς εναλλαγές στην τοπολογία του δικτύου της συσκευής αλλά και ούτε την κατάστασή της κάθε στιγμή.

Υιοθετώντας αυτήν την αρχιτεκτονική δεν χρειάζονται διαδικασίες όπως το ταίριασμα (matchmaking) και η “διαφήμιση” των χαρακτηριστικών που διαθέτουν οι πόροι. Οι διαδικασίες αυτές είναι εξαιρετικά πολύπλοκες, για αυτού του τύπου τις συσκευές, και ταυτόχρονα δεν μπορούν να ανταποκριθούν στην κινητικότητα των συγκεκριμένων πόρων. Επιπροσθέτως, παρατηρείται ότι τα τεχνικά χαρακτηριστικά που παρέχουν αυτές οι συσκευές δεν διαφέρουν πολύ, ανά κατηγορία, που σημαίνει ότι τα τυπικά χαρακτηριστικά (υπολογιστική ισχύς, διαθέσιμη RAM) που διαθέτουν οι συσκευές τύπου smartphone βρίσκονται στην ίδια κλίμακα και δεν διαφοροποιούνται ιδιαίτερα μεταξύ τους. Το ίδιο συμβαίνει και εντός της κατηγορίας των συσκευών τύπου tablet PC.

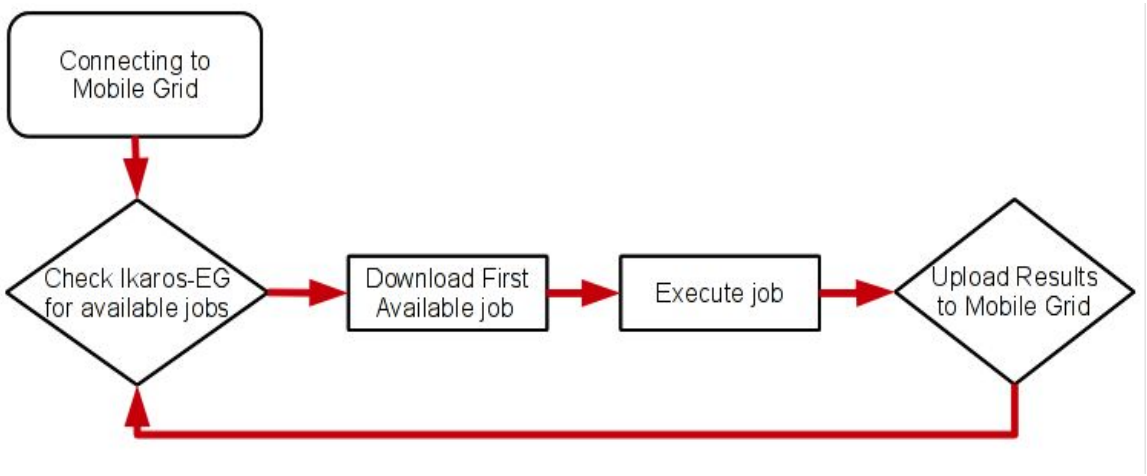
Άρα η μόνη απαίτηση είναι να δημιουργηθεί αντίστοιχος αριθμός διαφορετικών κατηγοριών διεργασιών όσους και οι κατηγορίες των συσκευών που θα χρησιμοποιηθούν. Έτσι παραμερίζονται όλες οι λειτουργίες ενός κλασσικού συστήματος δρομολόγησης (όπως το Condor). Στην συγκεκριμένη περίπτωση αρκούν δύο διαφορετικού τύπου διεργασίες που θα είναι διαθέσιμες από τον πάροχο περιεχομένου και ουσιαστικά θα αποτελούν δυο διαφορετικές εφαρμογές για αυτού του τύπου τις συσκευές ή μια εφαρμογή που θα έχει την δυνατότητα πρόσβασης σε δύο διαφορετικές συλλογές περιεχομένου, ανάλογα με τις ανάγκες υλοποίησης της εκάστοτε αρχιτεκτονικής.

Το άλλο μεγάλο πρόβλημα που πρέπει να επιλυθεί είναι η μεγάλη κινητικότητα αυτών των συσκευών. Αυτό το χαρακτηριστικό θα δημιουργούσε σημαντικά ζητήματα σε παραδοσιακά συστήματα δρομολόγησης που απευθύνονται σε κατανεμημένα περιβάλλοντα καθώς θα έπρεπε να υπάρχουν αλγόριθμοι που να εξασφαλίζουν την προσαρμογή των πόρων στις νέες συνθήκες. Τα συστήματα αυτά απαιτούν πλήρη ή τουλάχιστον μερικό έλεγχο των πόρων. Η χρήση του επεξεργαστή, της RAM αλλά και των άλλων συστημάτων θα πρέπει να ελέγχεται κεντρικά από τον δρομολογητή και όχι από τον ιδιοκτήτη του πόρου.

Οι συνθήκες λειτουργίας αυτών των συσκευών μεταβάλλονται με ταχύτατους ρυθμούς, καθώς υπάρχουν συνεχείς αλλαγές στην τοπολογία του δικτύου αλλά και την συνδεσιμότητα του πόρου γενικότερα. Με την αρχιτεκτονική που υλοποιεί το ΙΚΑΡΟΣ κάτι τέτοιο δεν είναι απαραίτητο, αφού είναι ο πόρος αυτός που καλείται να επικοινωνήσει με το σύστημα, το οποίο είναι αμετάβλητο, και όχι το αντίστροφο.

Τέλος, περιπτώσεις κατά τις οποίες ο πόρος απενεργοποιείται χάνει την σύνδεση για μεγάλο χρονικό διάστημα ή ο χρήστης που έχει στη κατοχή του την συσκευή αποφασίζει να τερματίσει τη λειτουργία της εφαρμογής χωρίς προειδοποίηση ουσιαστικά διευθετούνται από τη φύση των διεργασιών. Αυτές οι διεργασίες θεωρούνται “αναλώσιμες” και διαθέτουν αρκετά ευέλικτα χαρακτηριστικά. Τέτοιου είδους διεργασίες μπορούν να δρομολογηθούν εξαρχής αν για παράδειγμα παρέλθει ο διπλάσιος από τον εκτιμώμενο χρόνο εκτέλεσης, ακολουθώντας το μοντέλο εκτέλεσης διεργασιών.

Στο σχήμα 24 παρουσιάζεται το διάγραμμα ροής της πιλοτικής εφαρμογής ikarosM, η οποία είναι διαθέσιμη για συσκευές που υποστηρίζουν την πλατφόρμα android [15]. Η πλατφόρμα android παρέχεται με μορφή ανοιχτού κώδικα από την Google [16] και την open handset alliance (OHA) [17].



Σχήμα 24: Διάγραμμα ροής εφαρμογής ikarosM

Η εφαρμογή ikarosM λειτουργεί ως μια μηχανή καταστάσεων μία οντότητα που μπορεί να αποθηκεύσει την κατάσταση στην οποία βρίσκεται και μπορεί να τη μεταβάλει ανάλογα με κάποια νέα είσοδο. Έτσι αρχικά ρωτά το πάροχο περιεχομένου για διαθέσιμες διεργασίες, αν υπάρχουν λαμβάνει την πρώτη διαθέσιμη, εν συνεχεία εκτελεί την διεργασία και τέλος επιστρέφει τα αποτελέσματα.

Ο πάροχος επιφορτίζεται με το να δημιουργεί και να διαθέτει κατάλληλες διεργασίες, στην πραγματικότητα διαθέτει διεργασίες με τη μορφή εφαρμογών για τέτοιου είδους συσκευές. Σε αυτήν τη περίπτωσή διαθέτει android εφαρμογές. Κατά αυτόν το τρόπο το έργο της κατανομής των διεργασιών, η αξιοπιστία και ο βαθμός σύνδεσης των πόρων με την υποδομή δεν διαδραματίζουν πρωταρχικό ρόλο, καθώς η διαδικασία εκτέλεσης των διεργασιών δεν επηρεάζεται από το αν η συσκευή αλλάξει περιβάλλον λειτουργίας.

Αυτή η λογική φαίνεται να είναι περισσότερο αποδοτική για αυτού του τύπου τις συσκευές καθώς τα τεχνικά τους χαρακτηριστικά δεν διαφέρουν ιδιαίτερα. Φαίνεται πως για τις συγκεκριμένες συσκευές είναι πιο σημαντικό το να δημιουργηθούν διεργασίες-εφαρμογές που δεν θα καταναλώνουν μεγάλο ποσοστό από τη διαθέσιμη ενέργεια της μπαταρίας και θα έχουν μικρό χρόνο ζωής. Έτσι εξασφαλίζεται η μη ενόχληση του κατόχου της συσκευής και ταυτόχρονα αποφεύγεται η διαδικασία δημιουργίας πολύπλοκων μηχανισμών κατανομής των πόρων.

8.2 Πιλοτική εφαρμογή IkarosM

Στην συνέχεια παρουσιάζεται η υλοποίηση της android εφαρμογής IkarosM, καθώς και το πώς αυτή λειτουργεί στο πλαίσιο που δημιουργείται από το ΙΚΑΡΟΣ. Το σύστημα που χρησιμοποιήθηκε ήταν η συστοιχία υπολογιστών “ZEUS”, όπου και εγκαταστάθηκε το ΙΚΑΡΟΣ. Το ΙΚΑΡΟΣ χρησιμοποίησε δεδομένα στα οποία έχει πρόσβαση η συστοιχία σε τοπικό καθώς και σε απομακρυσμένο επίπεδο. Ο στόχος ήταν να χρησιμοποιηθούν, με διαφανή τρόπο προς την υπολογιστική υποδομή, πόροι τύπου smartphone ή tabletPC που χρησιμοποιούν την ασύρματη διαδικτυακή υποδομή του Ε.Κ.Ε.Φ.Ε Δημόκριτος. Κατά αυτόν τον τρόπο, χρησιμοποιήθηκαν αυτού του τύπου οι πόροι ώστε να μειωθεί το συνολικό φορτίο στην υπάρχουσα υποδομή ενώ ταυτόχρονα υπάρχει η δυνατότητα να προστεθούν νέα χαρακτηριστικά στο όλο σύστημα.

Η android εφαρμογή η οποία και παρουσιάζεται είναι ένας μετατροπέας μεταδεδομένων σε δομή τύπου XML. Στις εικόνες 33, 34, 35, 36, παρουσιάζονται στιγμιότυπα από την εκτέλεση της εφαρμογής σε ένα smartphone δείχνοντας ταυτόχρονα και τα διάφορα ανεξάρτητα βήματα που ακολουθούνται κατά την εκτέλεση της εφαρμογής, σύμφωνα με το διαγράμματος ροής. Πιο συγκεκριμένα, η εφαρμογή αρχικά συνδέεται στο πλαίσιο

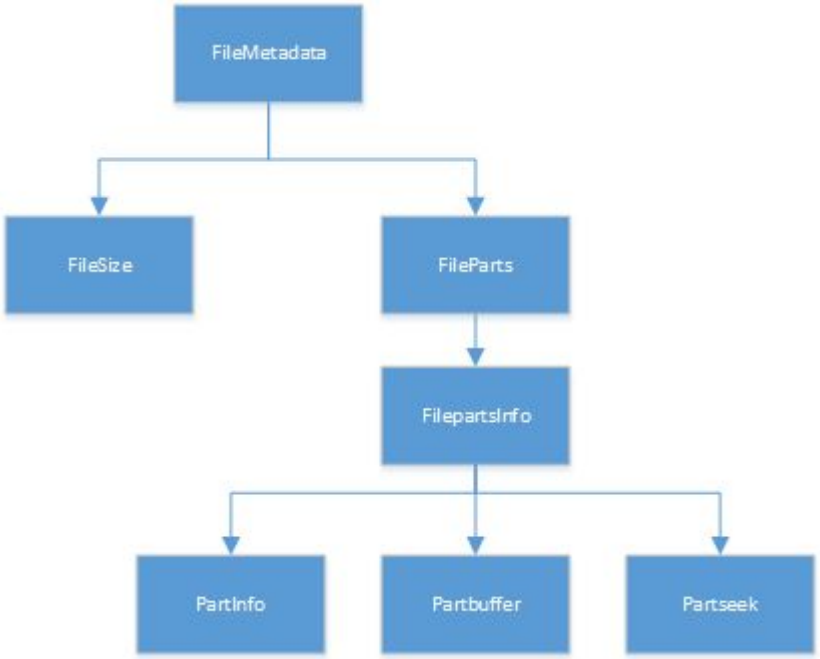
του Mobile Grid που υλοποιεί το ΙΚΑΡΟΣ, στην συνέχεια ανασύρει την πρώτη διαθέσιμη διεργασία, προχωράει στην εκτέλεσή της και στο τέλος επιστρέφει τα αποτελέσματα στην υποδομή αντιμετωπίζοντας το ΙΚΑΡΟΣ σαν ένα πάροχο περιεχομένου. Στην εικόνα 35 φαίνεται ότι η εφαρμογή ανασύρει τα μεταδεδομένα στην αρχική τους μορφή (ένα απλό string) και μετά την εκτέλεση της διεργασίας επιστρέφει στο ΙΚΑΡΟΣ τα δεδομένα με την μορφή ενός XML DOM δέντρου (εικόνα 37).

Εικόνα 33:IkarosM, σύνδεση στο Mobile Grid

Εικόνα 34: IkarosM, ανάσυρση διεργασίας

Εικόνα 35: IkarosM, λήψη αδιαμόρφωτων μεταδεδομένων

Εικόνα 36: IkarosM, επιστροφή αποτελεσμάτων στο ΙΚΑΡΟΣ



Εικόνα 37: XML DOM δέντρο

Αυτού του είδους οι εφαρμογές είναι κατάλληλες για συσκευές τύπου smartphone και table, καθώς δεν απαιτούν μεγάλη επεξεργαστική ισχύ αλλά ούτε και υψηλό ρυθμό διαδικτυακής “κίνησης”, κατά την διάρκεια εκτέλεσης της εφαρμογής. Με αποτέλεσμα να μην αποκλείεται ο ιδιοκτήτης της συσκευής από την ταυτόχρονη χρήση της, για ίδιους σκοπούς. Ταυτόχρονα, η κατανάλωση ηλεκτρικής ενέργειας είναι εξαιρετικά χαμηλή. Η χαμηλή κατανάλωση ενέργειας αποτελεί έναν από τους βασικότερους στόχους καθώς έτσι ο ιδιοκτήτης της συσκευής παραμένει “ευχαριστημένος” με αποτέλεσμα να είναι πρόθυμος να διαθέσει την συσκευή του, ειδικά αν μπορεί να έχει κάποιο επιπλέον όφελος.

Πιο συγκεκριμένα, μπορούν να δημιουργηθούν υβριδικά σχήματα υποδομών στα οποία οι ιδιοκτήτες αυτού του τύπου των συσκευών θα τις διαθέτουν, με διάφανο τρόπο, σε επιστημονικά πειράματα και επιχειρηματικές υπολογιστικές υποδομές μέσω των τηλεπικοινωνιακών παρόχων. Τα αιτήματα για τη χρήση τέτοιου είδους εφαρμογών μπορεί να είναι ακόμα και χιλιάδες σε petascale/eXascale υποδομές, οδηγώντας σε μια συνεχή υπερφόρτωση της υποδομής. Έτσι από την μία θα μπορούν να χρησιμοποιηθούν “ευκαιριακοί” πόροι για να μειωθεί τον συνολικό φόρτο μίας υποδομής καθώς και η συνολική κατανάλωσης ενέργειας αλλά και να προσφερθούν πραγματικά κίνητρα στους ιδιοκτήτες των συσκευών. Για παράδειγμα η μείωση των λογαριασμών κινητής τηλεφωνίας ανάλογα με τη χρήση. Τα οφέλη μπορεί να είναι τεράστια αν υπολογιστεί ότι αυτού του τύπου οι πόροι είναι εκατομμύρια με αυξητική τάση και ανανεώνονται συνεχώς από πλευράς τεχνικών προδιαγραφών.

9. **ΣΥΜΠΕΡΑΣΜΑΤΑ / ΜΕΛΛΟΝΤΙΚΗ ΕΡΓΑΣΙΑ**

Οι επιστημονικοί υπολογισμοί μεγάλης κλίμακας είναι εξαιρετικά απαιτητικοί με αποτέλεσμα να έχουν μεγάλες ανάγκες σε υπολογιστική ισχύ. Οι παράλληλοι υπολογισμοί και τα παράλληλα συστήματα αρχείων αναγνωρίζονται ως η μόνη εφικτή λύση σε αυτού του είδους τα προβλήματα ενώ οι διεργασίες εισόδου/εξόδου (I/O) αποτελούν το σημαντικότερο σημείο συμφόρησης στην απόδοση των εφαρμογών. Τα προβλήματα εντείνονται καθώς οι επιδόσεις των επεξεργαστών αυξάνονται ενώ το λογισμικό και το υλικό που δομεί τις αποθηκευτικές διατάξεις δεν εξελίσσεται με αντίστοιχους ρυθμούς. Οι σημαντικότεροι παράγοντες που επηρεάζουν την απόδοση είναι ο αριθμός των παράλληλων διεργασιών που συμμετέχουν στις μεταφορές των δεδομένων, το μέγεθος της κάθε μεταφοράς καθώς και τα διάφορα I/O μοτίβα πρόσβασης.

Ένας επιπλέον σημαντικός παράγοντας που επηρεάζει την απόδοση είναι η γενικότερη αρχιτεκτονική της αποθηκευτικής υποδομής. Μια τυπική High Performance Computing (HPC) υποδομή χρησιμοποιεί ένα μικρό μέρος των διαθέσιμων κόμβων για αποθηκευτικούς σκοπούς (κόμβοι I/O). Κάθε ένας από τους κόμβους I/O διαθέτει έναν μεγάλο αριθμό σκληρών δίσκων και εφαρμόζει ένα Redundant Array of Independent Disks (RAID) σχηματισμό για την οργάνωση των αποθηκευτικών μέσων. Έτσι, τα διαμοιραζόμενα συστήματα αρχείων έχουν σημαντικούς περιορισμούς όταν εφαρμόζονται σε μεγάλης κλίμακας συστήματα, επειδή: το εύρος ζώνης δεν κλιμακώνει οικονομικά και η I/O κίνηση στην δικτυακή υποδομή και στους αποθηκευτικούς κόμβους μπορεί να επηρεαστεί από άλλες ξένες προς αυτή διεργασίες ή με την σειρά της να επηρεάσει άλλες διεργασίες.

Στοχεύοντας στην επίλυση των πιο πάνω περιορισμών αναπτύχθηκε το πλαίσιο ΙΚΑΡΟΣ ως ένας μηχανισμός που επιτρέπει να δομούνται αποθηκευτικοί σχηματισμοί on demand. Το ΙΚΑΡΟΣ επιτυγχάνει καλύτερο συντονισμό μεταξύ των πολλαπλών στρωμάτων λογισμικού στην συνολική ροή των δεδομένων (τοπική-απομακρυσμένη πρόσβαση), διατηρώντας ταυτόχρονα την αυτονομία των επιπέδων. Επιτρέπει την κλιμάκωση του διαθέσιμου εύρους ζώνης (I/O και δίκτυο) με κόστος ανάλογο με αυτό της κλιμάκωσης της χωρητικότητας των αποθηκευτικών συστημάτων. Στοχεύει στη δημιουργία υποδομών που θα απαιτούν εξαιρετικά μικρότερα ποσά ηλεκτρικής ενέργειας καθώς και στην αντιμετώπιση των προβλημάτων κλιμάκωσης των μηχανισμών μεταδεδομένων.

Το ΙΚΑΡΟΣ έχει δομηθεί ως ένα “λεπτό” στρώμα (thin layer) που έχει την δυνατότητα να προσφέρει υπηρεσίες σε πολλαπλά επίπεδα, μπορεί να χρησιμοποιήσει ένα πολύ μεγάλο αριθμό αποθηκευτικών κόμβων και επιτρέπει την απομόνωσή των λειτουργιών I/O μίας διεργασίας από τις αντίστοιχες των άλλων διεργασιών, στοχεύοντας στην μέγιστη αξιοποίηση των διαθέσιμων πόρων. Οι μετρήσεις στην εικόνα 29 δείχνουν ξεκάθαρα ότι η απόδοση του GPFS επηρεάζεται σημαντικά από τον ανταγωνισμό των διεργασιών σε σχέση με τους αποθηκευτικούς πόρους. Το ΙΚΑΡΟΣ δημιουργεί απομονωμένους, από τις άλλες διεργασίες, αποκλειστικούς αποθηκευτικούς σχηματισμούς με λόγο HDD/client 4:1 ή ημιαποκλειστικούς με λόγο 4:2 και 4:4. Έτσι οι διεργασίες εγγραφής δεν ανταγωνίζονται για πόρους, σε αυτήν την περίπτωσή για σκληρούς δίσκους.

Στην εικόνα 31 φαίνεται καθαρά ότι το ΙΚΑΡΟΣ υπερτερεί του GPFS όταν χρησιμοποιεί τους λόγους 4:2 και 4:1 στην διαχείριση των διαθέσιμων σκληρών δίσκων. Αυτή η προσέγγιση μπορεί να κλιμακώσει με πάνω όριο το διαθέσιμο εύρος ζώνης σε επίπεδο δικτύου έχοντας εκμεταλλευτεί πλήρως τη διαθέσιμη I/O υποδομή σε σκληρούς δίσκους,

κάτι που δεν επιτυγχάνει το GPFS. Με τη χρήση του ΙΚΑΡΟΣ επιτυγχάνεται η βελτίωση της I/O απόδοσης κατά 33% με το 1/3 των διαθέσιμων σκληρών δίσκων.

Το ΙΚΑΡΟΣ προσφέρει άμεση πρόσβαση σε κάθε αποθηκευτικό I/O κόμβο, ανεξάρτητα από την βαθμίδα (Tier) στην οποία ενεργεί. Κάθε βαθμίδα παρέχει πρόσβαση σε πολλαπλά υπολογιστικά κέντρα και υποστηρίζει μια συγκεκριμένη ομάδα υπηρεσιών. Στο οικοσύστημα των εφαρμογών στο οποίο ενεργεί κυριαρχούν οι λειτουργίες εγγραφής. Το ΙΚΑΡΟΣ σε αυτό το περιβάλλον υπερέχει έναντι των άλλων συστημάτων κυρίως λόγω των τεχνικών buffering και caching στην πλευρά του πελάτη, της χρήσης της τεχνικής reverse HTTP καθώς και στην υλοποίηση των διεργασιών write ως “reversed read”.

Ο συνδυασμός του reversed read με την reverse HTTP, ανά περίπτωση, επιτρέπουν στο ΙΚΑΡΟΣ να ενεργεί κυρίως στο επίπεδο του δικτύου κάνοντας κυρίως routing των δεδομένων και αποφεύγοντας την εμπλοκή του λειτουργικού συστήματος, για τη συνολική ροή των δεδομένων (τοπική και απομακρυσμένη πρόσβαση). Όπως φαίνεται από την εικόνα 28 και το σχήμα 21, η χρήση αυτών των τεχνικών του ΙΚΑΡΟΣ παρέχει μέγιστη ευελιξία και επιτρέπεται σε τεχνικό επίπεδο να επιτευχθεί η άμεση πρόσβαση σε οποιουδήποτε αποθηκευτικό κόμβο εισόδου/εξόδου ανεξάρτητα από την βαθμίδα στην οποία αυτός ενεργεί, διασφαλίζοντας παράλληλα την αυτονομία υλοποίησης των επιμέρους υπηρεσιών. Ταυτόχρονα επιτυγχάνεται η μείωση του ανταγωνισμού, για ανεύρεση πόρων, μεταξύ δικτύου και αποθηκευτικών μέσων εφαρμόζοντας συντονισμένες παράλληλες μεταφορές δεδομένων στην συνολική ροή της μεταφοράς [84].

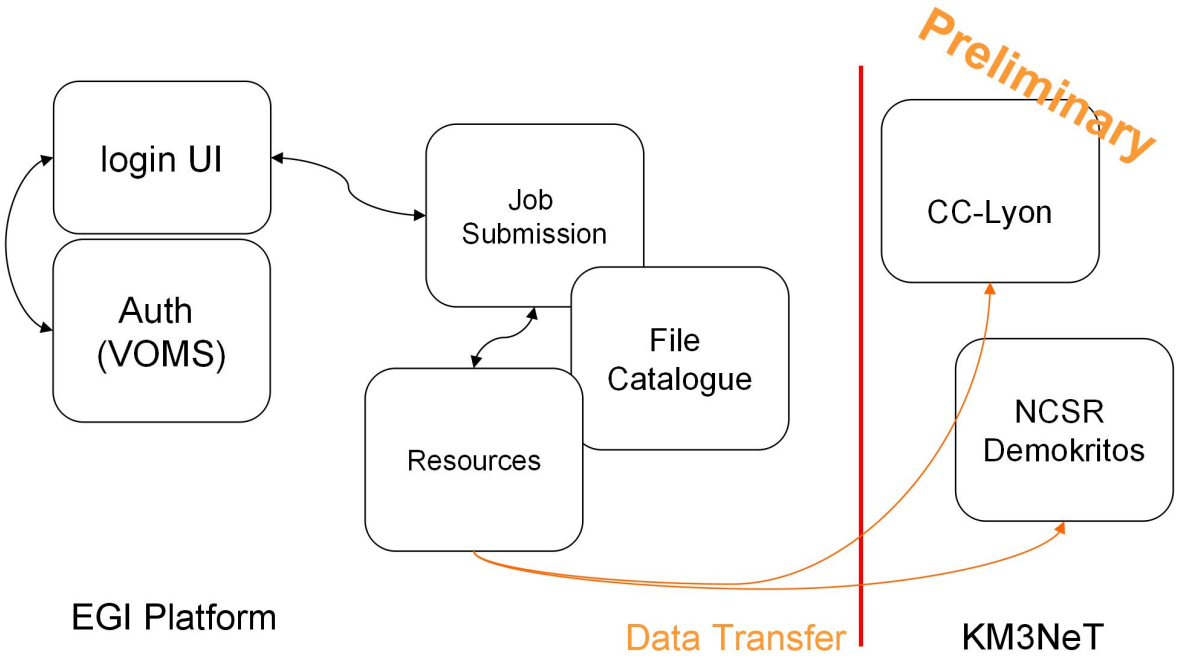
Η προσέγγιση που ακολουθεί το ΙΚΑΡΟΣ επιτυγχάνει να ελαχιστοποιήσει το συνολικό κόστος παρέχοντας ταυτόχρονα μεγαλύτερη ευελιξία στις υποδομές δομώντας στο, επίπεδο του χρήστη έναν ενιαίο αποθηκευτικό σχηματισμό που μπορεί να αποτελείται από όλες τις διαφορετικές υποδομές (Grids, Clouds, HPCs, Data Centers και τοπικές συστοιχίες υπολογιστών) που χρησιμοποιεί το εκάστοτε υπολογιστικό μοντέλο.

Αυτή η λογική δίνει την δυνατότητα να δημιουργηθούν υποδομές στις οποίες οι χρήστες θα έχουν μεγαλύτερη επιρροή στην διακυβέρνηση τους κάτι που θα επιστρέψει να τοποθετηθεί η επιστήμη των υπολογιστών και η εκμετάλλευσή των “Big Data” στο κέντρο της επιστημονικής ανακάλυψης, στοχεύοντας παράλληλα στην ανάπτυξη αποθηκευτικών συστημάτων νέας γενιάς που θα έχουν δυνατότητα κλιμάκωσης σε eXascale περιβάλλοντα.

9.1 Μελλοντική Εργασία

Κάποιες από τις λειτουργίες του ΙΚΑΡΟΣ έχουν αρχίσει να ενσωματώνονται στο υπολογιστικό μοντέλο του πειράματος KM3NeT. Το KM3NeT στοχεύει στην δημιουργία ενός τηλεσκοπίου νετρίνων μεγέθους πολλαπλών κυβικών χιλιομέτρων και αποτελεί European Strategy Forum on Research Infrastructures (ESFRI) υποδομή. Το KM3NeT είναι αναγνωρισμένο πείραμα από το CERN. Η κοινοπραξία αποτελείται από 10 χώρες, 33 Ινστιτούτα και 5 παρατηρητές.

Οι λειτουργίες του ΙΚΑΡΟΣ χρησιμοποιούνται έτσι ώστε να μεταφέρονται τα δεδομένα απευθείας, σε ένα βήμα, από τους υπολογιστικούς πόρους της υποδομής πλέγματος του EGI στο υπολογιστικό κέντρο του IN2P3 στη Lyon της Γαλλίας καθώς και στις υπολογιστικές υποδομές του Δημόκριτου (εικόνα 38). Κατά τα αυτό τον τρόπο αποφεύγεται η καθορισμένη διαδικασία που προσφέρουν οι υπάρχοντες μηχανισμοί μεταφοράς δεδομένων του EGI. Αυτή η διαδικασία επιβάλει τη μεταφορά των δεδομένων, με την χρήση του gridFTP, από το υπολογιστικό πόρο (worker node) σε ένα τοπικό αποθηκευτικό σύστημα (storage element), απο το storage element στην διεπαφή του χρήστη με την υποδομή (gLite UI) και από εκεί στο τελικό προορισμό.



Εικόνα 38: Ροή διεργασίας EGI-KM3NeT

Επίσης σαν μελλοντική εργασία αναγνωρίζεται η αναπτύξει μεσισμικό το οποίο θα επεκτείνει τις υπάρχουσες DaaS λύσεις προς την κατεύθυνσης του “εμπλουτισμού” (enrichment) αυτών κάθε αυτών των μοντέλων δεδομένων, ξεπερνώντας τον απλό διαμοιρασμό τους. Εν προκειμένω, η έννοια του “εμπλουτισμού” αναφέρεται στην ικανότητα του μοντέλου δεδομένων να επεκτείνεται από νέες ιδέες και λειτουργίες που έχουν σχεδιαστεί από τρίτα μέρη. Σε αυτά τα πλαίσια εντάσσονται και οι μελλοντικές προσπάθειες για επέκταση του ΙΚΑΡΟΣ ώστε να λαμβάνει με αυτόματο τρόπο τις απαραίτητες πληροφορίες για την παραμετροποίηση του καθώς και η δυνατότητα άμεσης πρόσβασης του σε αποθηκευτικά μέσα χωρίς την χρήση κάποιου native file system.

Η επέκταση του DaaS μοντέλου είναι απαραίτητη και στον επιστημονικό αλλά και στον επιχειρηματικό τομέα, καθώς είναι ο μόνος τρόπος για να επιτευχθούν συνέργειες μεταξύ ευρύτερων κοινοτήτων και να υπάρχουν περισσότερα οφέλη. Το υπολογιστικό μοντέλο που θα πρέπει να υλοποιηθεί στις μελλοντικές εφαρμογές θα είναι εξαιρετικά πολύπλοκο και θα πρέπει να μπορεί να παρέχει καλύτερο συντονισμό μεταξύ υλικού και λογισμικού με λιγότερα επίπεδα συγχρονισμού. Έτσι το ΙΚΑΡΟΣ και η προσέγγιση που ακολουθεί είναι πολύ πιθανόν ότι θα διαδραματίσει σημαντικό ρόλο στην ανάπτυξη των συστημάτων αποθήκευσης και διαχείρισης δεδομένων επόμενης γενιάς.

ΠΙΝΑΚΑΣ ΟΡΟΛΟΓΙΑΣ

Ξενόγλωσσος όρος	**Ελληνικός Όρος**
3G	τηλεπικοινωνιακά δίκτυα τρίτης γενιάς
ACL	έλεγχος πρόσβασης λίστας
Activity	Δραστηριότητα
agents	αντιπρόσωποι
android package – APK	Πακέτο android εφαρμογής
Apache module	Μονάδα του Apache Hypertext Transfer Protocol (HTTP) εξυπηρετητή
Assimilator	υπηρεσία αφομοίωσης
Reliability	Αξιοπιστία
Application programming interface - API	Διεπαφές προγραμματισμού εφαρμογής
Batch job systems	συστήματα διαχείρισης πόρων βασιζόμενα σε μηχανισμούς χειρισμού διεργασιών σε επίπεδο δέσμης
brokering	υπηρεσίες διαμεσολάβησης
checkpointing	σημείο ελέγχου
chunk of data	μεγάλο “κομμάτι” δεδομένων
Collaboration Grids	Συνεργατικές υποδομές πλέγματος
collision avoidance	αποφυγή της σύγκρουσης
Computing Grid	Υπολογιστικό Πλέγμα
pinning files	Καρφίτσωμα-δέσμευση αρχείων
pool	συλλογή
content generator	γεννήτρια περιεχομένου
cycle scavenging	κυκλική σάρωση των πόρων
.dex	Dalvik εκτελέσιμα
Data Centre Grids	Υποδομές Πλέγματος- υπολογιστικών κέντρων
disk caches	οντότητες δίσκων προσωρινής αποθήκευσης
Distributed Resource Management Application API – DRMAA	Διεπαφή εφαρμογών κατανεμημένης διαχείρισης πόρων
DRM systems	Συστήματα κατανεμημένης διαχείρισης πόρων
Execution Management Services	Υπηρεσίες διαχείρισης εκτέλεσης
Feeder	Υπηρεσία τροφοδότησης
firewall	τείχος προστασίας
flocking	διαδικασία της συγκέντρωσης
fork	Μέθοδος διακλάδωσης
header	επικεφαλίδα
High performance computing – HPC	υπολογιστικό μοντέλο υψηλής διαπερατότητας
High throughput Computing - HTC	υπολογιστικό μοντέλο υψηλής διαπερατότητας
Hierarchical Resource Manger – HRM	ιεραρχικός διαχειριστής πόρων
idle cycles	άεργους κύκλους
intent receivers	Δέκτες ειδικού σκοπού
interpreter	διερμηνευτής
I/O	Διαδικασίες εισόδου/εξόδου
Virtual Organization - VO	Ιδεατός Οργανισμός
GPUs	μονάδες επεξεργασίας γραφικών
Grid Middleware	Μεσισμικό Πλέγματος
Grid Site	τοποθεσίας πλέγματος
Massive parallel processors (MPPS)	Μεγάλης κλίμακας παραλλήλους επεξεργαστές

Mass storage systems	συστήματα μαζικής αποθήκευσης
Matchmaking	Μηχανισμός αντιστοίχισης
Message passing	Ανταλλαγή μηνυμάτων
Master-Worker	επικεφαλής-εργάτη
nearline custodial	Έμμεσης πρόσβασης σύστημα τύπου κηδεμονίας
meta-schedulers	μετά-δρομολογητές/προγραμματιστές
Network Access Management	διαχείριση δικτυακής πρόσβασης
Network Address Translation - NAT	μετάφραση διεύθυνσης δικτύου
online custodial	Απευθείας σύνδεσης σύστημα, τύπου κηδεμονίας
online replica	Αντιγραφή απευθείας σύνδεσης
overhead	επιπλέον κόστος
partitioning	δημιουργία τομέων
Peer to Peer	Ισότιμος με ισότιμο (επικοινωνία μεταξύ ισότιμων)
personally identifiable information - PII	πληροφορίες αναγνώρισης ταυτότητας
Policy module	μονάδα τοπικής πολιτικής
preemptive-multitasking	πολύ-διεργασία που χρησιμοποιεί τεχνικές προγραμματισμού για τον χρόνου εκτέλεσης της κάθε διεργασίας (προ-εκχώρηση)
read-only	διαθέσιμα μόνο για ανάγνωση
redundant array of independent disks - RAID	πλεονασματική συστοιχία ανεξάρτητων δίσκων
Relational database management system – RDBMS	σύστημα διαχείρισης σχεσιακής βάσης δεδομένων
Request Queue Management	διαχείριση ουράς αναμονής αιτημάτων
Remote procedure calls – RPC	Διαδικασία απομακρυσμένης κλήσης
Resource broker	Αναζήτησης – μεσιτείας πόρων
Resource co-allocators	Κατανομής πόρων
resource virtualization	Εικονικοποίηση πόρων
sandbox	Άμμο κιβώτιο
script	δέσμη ενεργειών
semantics	σημασιολογία
semaphores	σηματοφόροι
service level agreements - SLA	συμφωνίες σε επίπεδο υπηρεσιών
service oriented architecture - SOA	αρχιτεκτονική προσανατολισμένη στις υπηρεσίες
Site File Name - SFN	όνομα αρχείου τοποθεσίας πλέγματος
shadow process	διεργασία “σκιά”
smartphones	ευφυή κινητά τηλέφωνα
Storage Element - SE	Αποθηκευτικό στοιχείο
Storage Resource Manager - SRM	διαχειριστής πόρων αποθήκευσης
table PCs	προσωπικούς υπολογιστές τύπου ταμπλέτας
threads	νήματα
tier	βαθμίδα
Tape Resource Manger – TRM	Διαχειριστής πόρων τύπου ταινίας
Wide Area Network -WAN	Δίκτυο ευρείας περιοχής
web portal	Διαδικτυακή πύλη
web services	υπηρεσίες ιστού
Work Generator	Υπηρεσία παραγωγής έργου
world wide web	παγκόσμιου ιστού
Uniform Resource Locator – URL	Ενιαίος Εντοπιστής Πόρων
virtual domain	ιδεατός τομέας

Validator	Υπηρεσία επικύρωσης
Volunteer Computing	Υπολογιστικές υποδομές βασιζόμενες στην εθελοντική προσφορά πόρων
Cloud	Σύννεφο
Content Provider	πάροχος περιεχομένου
stripping servers	χρήση πολλαπλών εξυπηρετητών
Transitioner	Υπηρεσία Μετάβασης

ΣΥΝΤΜΗΣΕΙΣ – ΑΡΚΤΙΚΟΛΕΞΑ – ΑΚΡΩΝΥΜΙΑ

API	Application programming interface
APK	Android package
APR	Apache Portable Runtime
BeStMan	Berkeley Storage Manager
BOINC	Berkeley Open Infrastructure for Network Computing
CASTOR	Cern Advanced Storage system
CGI	Common Gateway Interface
CIM	Common Information Model
CERN	European Organization for Nuclear Research
CMS	Compact Muon Solenoid
CORBA	Common Object Request Broker Architecture
DESY	Deutsches Elektronen-Synchrotron
CIFS	Common Internet File System
DPM	Disk Pool Manager
DRMAA	Distributed Resource Management Application API
EGI	European Grid Infrastructure
EMI	European Middleware Initiative
FNAL	Fermi National Accelerator Laboratory
FTP	File Transfer Protocol
GIIS	Grid Index Information Service
GIS	Grid Information System
GPFS	General Parallel File System
GPU	Graphics processing unit
GRAM	Globus Resource Allocation Managers
GRIS	Grid Resource Information Service
GSI	Grid Security Infrastructure
HPC	High performance computing
HPSS	High Performance Storage System
HTC	High throughput Computing
HTTP	Hypertext Transfer Protocol
HRM	Hierarchical Resource Manger

JME	Java Platform, Micro Edition
JVM	Java Platform, virtual machine
LBNL	Lawrence Berkeley National Laboratory
LSF	Load Sharing Facility
LDAP	Lightweight Directory Access Protocol
MDS	Monitoring and Discovery Service
MPI	Message Passing Interface
MPPS	Massive parallel processors
MPM	Multi-Processing Module
NAT	Network Address Translation
NFS	Network File System
OGF	Open Grid Forum
OGSA	Open Grid Services Architecture
OHA	open handset alliance
P2P	Peer-to-Peer
PheDEx	Physics Experiment Data Export
PII	personally identifiable information
POSIX.	Portable Operating System Interface for Unix
PVFS2	Parallel Virtual File System
PVM	Parallel Virtual Machine
PBS	Portable Batch System
RAID	redundant array of independent disks
RAL	Rutherford Appleton Laboratory
RDBMS	Relational database management system
RFIO	Remote File I/O
RPC	Remote procedure calls
RSL	Resource Specification Language
SE	Storage Element
SFN	Site File Name
SGE	Sun Grid Engine
SURLs	Site URLs
SNMP	Simple Network Management Protocol
SLA	service level agreements
SLURM	Simple Linux Utility for Resource Management

SOA	service oriented architecture
SOHO-NAS	Small Office/Home Office - Network Attach Network
SRB	Storage Resource Broker
SRM	Storage Resource Manager
SSH	Secure Shell
TURL	Transfer URL
UDP	User Datagram Protocol
UIDs	Unix user identifiers
URL	Uniform Resource Locator
VLAN	Virtual Local Area Network
VO	Virtual Organization
WAN	Wide Area Network
WBEM	Web-Based Enterprise Management
WebDav	Web-based Distributed Authoring and Versioning
WLCG	Worldwide LHC Computing Grid
WSDM	Web Services Distributed Management
WSRF	Web Services Resource Framework
TCP/IP	Transmission Control Protocol/ Internet Protocol
TJNAF	Thomas Jefferson National Accelerator Facility
TRM	Tape Resource Manger
UNISIST	Universal System for information in Science and technology
W3C	World Wide Web Consortium
XML	Extensible Markup Language
ΕΚΠΑ	Εθνικό και Καποδιστριακό Πανεπιστήμιο Αθηνών
Ε.Κ.Ε.Φ.Ε “Δημόκριτος”	Εθνικό Κέντρο Έρευνας Φυσικών Επιστημών “Δημόκριτος”
ΙΠΣΦ	Ινστιτούτο Πυρηνικής και Σωματιδιακής Φυσικής

ΠΑΡΑΡΤΗΜΑ Ι (HTTP και WebDav μέθοδοι)

Πίνακας 4:HTTP και WebDav μέθοδοι

HTTP και WebDav μέθοδοι	
Μέθοδος	Ενέργεια
HTTP	
GET	Ανασύρει την αναπαράσταση ενός πόρου
HEAD	Επιστρέφει την HTTP επικεφαλίδα, χωρίς να προσαρτά το περιεχόμενο
PUT	Τοποθετεί/εγγράφει έναν πόρο
DELETE	Καθιστά έναν πόρο μη προσβάσιμο από το ορισθέν URL
POST	Υποβάλει μια φόρμα ιστού/Χρησιμοποιείται από άλλα πρωτόκολλα με τεχνικές tunneling
OPTIONS	Παραθέτει τις διαθέσιμες μεθόδους, για τον εκάστοτε πόρο
TRACE	Ανασύρει τα ληφθέντα διαγνωστικά μηνύματα
CONNECT	Χρησιμοποιείται από τα proxies με σκοπό την δυναμική εναλλαγή σε ένα SSL tunnel
WebDav	
PROPFIND	Ανασύρει τις ιδιότητες των πόρων/παραθέτει τα μέλη των συλλογών
PROPPATCH	Εγγράφει τις ιδιότητες των πόρων
LOCK	Κλειδώνει την χρήση ενός πόρου η μιας συλλογής πόρων μέσω δια μοιραζόμενης ή αποκλειστικής τεχνικής κλειδώματος
UNLOCK	Αφαίρεση κλειδώματος
MKCOL	Δημιουργία νέας συλλογής
COPY	Αντιγραφή ενός πόρου ή μιας συλλογής, ιεραρχικά
MOVE	Μετακίνηση ενός πόρου ή μιας συλλογής, ιεραρχικά
Access Control	
ACL	Εγγράφει την λίστα ελέγχου πρόσβασης ενός πόρου
Ordered Collections	
ORDERPATCH	Μεταβάλλει την ιεράρχηση των πόρων σε μια συλλογή
Bindings	
BIND	Αντιστοιχεί έναν υφιστάμενο πόρο σε μια υφιστάμενη συλλογή
UNBIND	Διαγράφει την αντιστοίχιση ενός πόρου σε μια συλλογή
REBIND	Αυτόματα μετακινεί έναν πόρο από μια συλλογή σε μια άλλη

Redirect References	
MKREDIRECTREF	Δημιουργεί ανακατεύθυνση στην αναφορά ενός πόρου
UPDATEREDIRECTREF	Ανανεώνει την ανακατεύθυνση αναφοράς του πόρου σε διαφορετικό URL
WebDav Search	
SEARCH	Αναζητά τις ιδιότητες και το περιεχόμενο ενός πόρου με ιεραρχική δομή

Πίνακας 5:Υποστήριξη πελατών σε λειτουργίες του HTTP

Υποστήριξη πελατών σε λειτουργίες του HTTP		
	CURL	**BROWSER**
OS	οποιοδήποτε	οποιοδήποτε
GUI	ΟΧΙ	ΝΑΙ
CLI	ΝΑΙ	ΟΧΙ
X509	ΝΑΙ	ΝΑΙ
Proxies	ΝΑΙ	Μόνο ο IE
Redirect	ΝΑΙ	ΝΑΙ
PUT	ΝΑΙ	ΟΧΙ

Πίνακας 6:Υποστήριξη πελατών σε λειτουργίες του WebDav

Υποστήριξη πελατών σε λειτουργίες του WebDav							
	TrailMix	Cadaver	DavLib	Shared Folders	DavFS2	Nautilus	Dolphin
OS	Firefox < 4	*nix	MacOS X	Windows	*nix	Gnome	KDE
GUI	ΝΑΙ	ΟΧΙ	ΝΑΙ	ΝΑΙ	-	ΝΑΙ	ΝΑΙ
CLI	ΟΧΙ	ΝΑΙ	ΟΧΙ	ΟΧΙ	-	ΟΧΙ	ΟΧΙ
X509	ΝΑΙ	ΝΑΙ	ΟΧΙ	ΝΑΙ	ΝΑΙ	ΟΧΙ	ΟΧΙ
Proxies	-	ΟΧΙ	ΟΧΙ	ΝΑΙ	ΟΧΙ	ΟΧΙ	ΟΧΙ
Redirect	ΝΑΙ	ΟΧΙ	ΝΑΙ	Δεν είναι διαθέσιμο για το PUT / Στα Windows 7 δεν υποστηρίζεται ούτε για τον GET	ΟΧΙ	ΟΧΙ	ΝΑΙ

ΠΑΡΑΡΤΗΜΑ ΙΙ (Επεκτάσεις του FTP)

Πίνακας 7:Επεκτάσεις του FTP, που χρησιμοποιούνται για την υλοποίηση του GridFTP, προσαρτήθηκε από [36]

Επεκτάσεις του FTP, που χρησιμοποιούνται για την υλοποίηση του GridFTP	
SPAS (Striped Passive)	Επιτρέπει την επιστροφή συνδέσεων με μια συστοιχία από εξυπηρετητές/θύρες. Ουσιαστικά ενεργοποιεί πολλαπλούς δια συνδεδεμένους εξυπηρετητές στο να συμμετάσχουν σε μια μεταφορά δεδομένων
SPOR (Striped Port)	Επιτρέπει την αποστολή μιας συστοιχίας από εξυπηρετητές/θύρες. Ουσιαστικά ενεργοποιεί πολλαπλούς δια συνδεδεμένους εξυπηρετητές στο να συμμετάσχουν σε μια μεταφορά δεδομένων
ERET (Extended Retrieve)	Επιτρέπει τον χειρισμό των δεδομένων πριν αυτά αποσταλούν.
ESTO (Extended Store)	Επιτρέπει τον χειρισμό των δεδομένων πριν αυτά αποθηκευτούν.
SBUF (Set TCP Buffer Size)	Επιτρέπει τον ρητό ορισμό του μεγέθους του TCP Buffer
ABUF (Auto-negotiate TCP Buffer Size)	Επιτρέπει την επιλογή ενός αλγορίθμου, με σκοπό τον αυτόματο προσδιορισμό του κατάλληλου μεγέθους για το TCP Buffer
DCAU (Data Channel Authentication)	Καθιερώνει έναν τρόπο για την χρήση τεχνικών gss στο κανάλι ελέγχου, αλλά όχι και στο κανάλι μεταφοράς των δεδομένων
RETR (TCP Streams Options)	Παρέχει επιπλέον επιλογές, σχετικά με τον προσδιορισμό του αριθμού των TCP καναλιών, για την διενέργεια μιας μεταφοράς δεδομένων, καθώς και για την μορφή των δεδομένων όταν στην μεταφορά συμμετέχουν πολλαπλοί διαδικτυακά διασκορπισμένοι εξυπηρετητές
FEAT (Appropriate Feature Responses)	Χρησιμοποιείται ώστε να μπορεί ο πελάτης να καθορίσει τις λειτουργίες που υποστηρίζει ο συγκεκριμένος εξυπηρετητής
EBLOCK (Extended Block)	Αποστέλλει τα δεδομένα με την μορφή block,στο οποίο έχουμε: 8 bit flag, 64 bit μήκος δεδομένων και 64 bit offset. Κατά αυτόν τον τρόπο επιτυγχάνετε η υποδοχή των δεδομένων, χωρίς την απαίτηση της αποστολής τους με συγκεκριμένη σειρά (μια λειτουργία απαραίτητη για μεταφορές δεδομένων που υποστηρίζουν PARALLEL και STRIPED τρόπους μεταφοράς)

ΑΝΑΦΟΡΕΣ

[1] I. Foster, C. Kesselman and S Tuecke, The Anatomy of the Grid: Enabling Scalable Virtual Organizations, International Journal of Supercomputer Applications, May 10, 2001.

[2] I. Foster, What is the Grid? A Three Point Checklist, GRIDToday, July 20, 2002.

[3] P.Dini, W.Gentzsch, M.Potts, A.Clemm, M.Yousif, A.Polze, “Internet, Grid, Self-Adaptability and Beyond: Are We Ready?”, Proc. 2nd Intl. Workshop on Self-Adaptive & Autonomic Computing Systems, Zaragoza, Spain, Aug. 30 – Sept 03,2004.

[4] W. Allcock, J. Bester, J. Bresnahan, A. Chervenak, L. Liming, S. Meder,and S. Tuecke. GridFTP Protocol Specification.

[5] N. Coleman, R. Raman, M. Livny, and M. Solomon. Distributed policy management and comprehension with classified advertisements. Technical Report UW-CS-TR-1481, University of Wisconsin - Madison Computer Sciences Department, April 2003.

[6] D. Epema, M. Livny, R. van Dantzig, X. Evers, and J. Pruyne. A worldwide flock of Condors: Load sharing among workstation clusters. Future Generation Computer Systems, 12:53–65, 1996.

[7] Ιστοχώρος WLCG, http://lcg.web.cern.ch/lcg/

[8] D. Thain, T. Tannenbaum, M. Livny: ”Distributed Computing in Practice: The Condor Experience” Concurrency and Computation: Practice and Experience, Vol. 17, No. 2-4, pages 323-356, February-April, 2005.

[9] M. Livny and R. Raman: High-throughput resource management. In I. Foster and C. Kesselman, editors, The Grid: Blueprint for a New Computing Infrastructure. Morgan Kaufmann, 1998.

[10] Ιστοχώρος Apache HTTP εξυπηρετητή, http://httpd.apache.org/

[11] N. Kew, Apache Modules Book, The: Application Development with Apache, Jan 26 2007, Prentice Hall, Part of the Prentice Hall Open Source Software Development Series series ISBN-10: 0-13-240967-4.

[12] HTTP Extensions for Web Distributed Authoring and Versioning (WebDAV), rfc4918, http://tools.ietf.org/html/rfc4918

[13] http://www.gridsite.org/talks/

[14] C. Filippidis, Y. Cotronis, C. Markou, DESIGN AND IMPLEMENTATION OF THE MOBILE GRID RESOURCE MANAGEMENT SYSTEM, Computer Science (2012), ISSN 1508-2806 (In Press)

[15] Ιστοχώρος android, http://www.android.com/

[16] Ιστοχώρος google, http://www.google.com

[17] Ιστοχώρος OHA, http://www.openhandsetalliance.com/

[18] Ιστοχώρος GPFS, http://www-03.ibm.com/systems/software/gpfs/

[19] P. H. Carns, W. B. Ligon III, R. B. Ross, and R. Thakur. PVFS: A parallel fille system for Linux clusters. In Proceedings of the 4th Annual Linux Show-case and Conference, pages 317327, Atlanta, GA, October 2000. USENIX Association.

[20] Ιστοχώρος Lustre, http://wiki.lustre.org/index.php/Main_Page

[21] F. Schmuck and R. Haskin. GPFS: A shared-disk _le system for large computing clusters. In Proceedings of the FAST 2002 Conference on File and Storage Technologies, San Jose, CA, January 2002. IBM Almaden Research Center.

[22] D. Hildebrand, P. Honeyman Exporting Storage Systems in a Scalable Manner with pNFS,MSST '05 Proceedings of the 22nd IEEE / 13th NASA Goddard Conference on Mass Storage Systems and Technologies

[23] D. Hildebrand, P. Honeyman Direct-pNFS: Scalable, transparent, and versatile access to parallel _le systems,HPDC '07 Proceedings of the 16th international symposium on High performance distributed computing

[24] C. Filippidis, Y. Cotronis, C. Markou, IKAROS: an HTTP-based distributed File System, for low consumption & low specification devices, 2013, Journal of Grid Computing (JOGC)

[25] Ιστοχώρος wget, http://www.gnu.org/software/wget/

[26] Ιστοχώρος curl, http://curl.haxx.se/

[27] Ιστοχώρος NFS, http://en.wikipedia.org/wiki/Network_File_System

[28] Ιστοχώρος CIFS, http://www.samba.org/cifs/

[29] J. Postel and J. Reynolds, “File Transfer Protocol,” IETF, RFC 959, 1985.

[30] T. Ylonen and C. Lonvick, eds., “The Secure Shell (SSH) Authentication Protocol,” IETF, RFC 4252, 2006.

[31] Ιστοχώρος WebDav, http://www.ietf.org/rfc/rfc4918.txt

[32] J. Whitehead, WebDAV: Versatile Collaboration Multiprotocol, 1089-7801/05/, 2005 IEEE INTERNET COMPUTING

[33] Ιστοχώρος CMS, http://cms.web.cern.ch/

[34] Ιστοχώρος KM3NeT, http://www.km3net.org
[35] L. TUURA et al, PhEDEx, high-throughput data transfer management system. Computing in High Energy and Nuclear Physics 13-17 February 2006, T.I.F.R. Mumbai, India. http://indico.cern.ch/contributionDisplay.py?contribId=389&confId=048
[36] W. Allcock, GridFTP: Protocol Extensions to FTP for the Grid, April 2003, GFD-20, http://www.ogf.org/documents/GFD.20.pdf
[37] Ιστοχώρος Akogrimo, http://www.akogrimo.org/
[38] Katsaros, K., Polyzos, G.C, Optimizing Operation of a Hierarchical Campus-wide Mobile Grid for Intermittent Wireless Connectivity, August 2007, Local & Metropolitan Area Networks, 2007. LANMAN 2007. 15th IEEE Workshop
[39] Foster I.,Zhao Y., Raicu I., Lu S.,Cloud Computing and Grid Computing 360-Degree Compared, Grid Computing Environments Workshop, 2008, GCE '08
[40] I. Raicu, Z. Zhang, M. Wilde, I. Foster, P. Beckman, K. Iskra, B. Clifford.,Toward Loosely Coupled Programming on Petascale Systems, IEEE SC 2008
[41] The International Exascale Software Roadmap," Dongarra, J., Beckman, P. et al., Volume 25, Number 1, 2011, International Journal of High Performance Computer Applications, ISSN 1094-3420
[42] The International Exascale Software Roadmap," Dongarra, J., Beckman, P. et al., Volume 25, Number 1, 2011, International Journal of High Performance Computer Applications, ISSN 1094-3420. Exascale Nearby Storage, Cray Position paper
[43] C. Filippidis, Y. Cotronis, C. Markou, IKAROS: Building an ad-hoc nearby storage based on IKAROS and social networking services, 2013, CHEP2013/Open Access Journal of Physics (JPCS)
[44] Ιστοχώρος Facebook API, https://developers.facebook.com/docs/reference/apis/, 2013
[45] Summary Report of the DOE Advanced Scientific Computing Advisory Commitee (ASCAC), March 30, 2013
[46] Peter Kogge et Al. Exascale computing study: Technology challenges in achieving exascale systems. Technical Report, AFRL contract Number FA8650-07-C-7724, 2008.
[47] Carl Shapiro, HalR.Varian, Information Rules: A strategic guide to network economy, Harvard business school press Boston, Massachusetts
[48] I. Raicu, I. Foster, P. Beckman. Making a Case for Distributed File Systems at Exascale, LSAP11, June 8, 2011.
[49] Maria D. Paraskevopoulou, Georgios Georgakilas, Nikos Kostoulas, Ioannis S. Vlachos, Thanasis Vergoulis, Martin Reczko, Christos Filippidis, Theodore Dalamagas and A.G. Hatzigeorgiou. (2013). DIANA-microT web server v5.0: service integration into miRNA functional analysis workflows. Nucleic Acids Research.
[50] Synergistic Challenges in Data-Intensive Science and Exascale Computing, Summary Report of the Advanced Scientic Computing Advisory Committee (ASCAC), March 30, 2013
[51] The Scientific Case for HPC in Europe 2012-2020, PRACE, October 2012
[52] Development of a Software Framework for the ANTARES Acoustic Data and Simulations within the Framework, Diplomarbeit aus der Physik, Alexander Wurstlein, 2010
[53] Ιστοχώρος ROOT http://root.cern.ch
[54] H. Stern. "Managing NFS and NIS". OReilly and Associates, Inc., 1991
[55] P.J. Braam. "The Coda distributed file system", Linux Journal, 50, 1998
[56] D. Nagle, D. Serenyi, A. Matthews. "The panasas activescale storage cluster: Delivering scalable high bandwidth storage". In SC 04: Proceedings of the 2004 ACM/IEEE conference on Supercomputing, 2004
[57] Microsoft Inc. "Distributed File System", http://www.microsoft.com/windowsserversystem/dfs/default.mspx/, 2011
[58] GlusterFS, http://www.gluster.com/, 2011
[59] Isilon Systems. "OneFS", http://www.isilon.com/, 2012
[60] "POHMELFS: Parallel Optimized Host Message Exchange Layered File System", http://www.ioremap.net/projects/pohmelfs/, 2012
[61] F. Hupfeld, T. Cortes, B. Kolbeck, E. Focht, M. Hess, J. Malo, J. Marti, J. Stender, E. Cesario. "XtreemFS - a case for object-based storage in Grid data management". VLDB Workshop on Data Management in Grids, 2007
[62] S. Ghemawat, H. Gobioff, S.T. Leung. The Google file system, 19th ACM SOSP, 2003
[63] H. Yang Z. Wenjun L. Qian. MapReduce Workload Modeling with Statistical Approach, J Grid Computing (2012) 10:279310
[64] Y. Gu, R. Grossman, A. Szalay, A. Thakar. Distributing the Sloan Digital Sky Survey Using UDT and Sector, e-Science 2006
[65] CloudStore, http://code.google.com/p/kosmosfs/, 2012

[66] S.A. Weil, S.A. Brandt, E.L. Miller, D.D.E. Long, C. Maltzahn. ”Ceph: A scalable, high-performance distributed file system”. In Proceedings of the 7th OSDI, 2006
[67] O. Tatebe et al.: Gfarm V2: A Grid file system that supports High-Performance distributed and parallel data computing, CHEP 04
[68] W. Xiaohui, W.W. Li, O. Tatebe, X. Gaochao, H. Liang, J. Jiubin. Implementing Data Aware Scheduling in Gfarm Using LSF Scheduler Plugin Mechanism, GCA05, 2005
[69] X. Wei, Li Wilfred W., T. Osamu, G. Xu, L. Hu, J. Ju. Integrating Local Job Scheduler LSF with Gfarm, ISPA05, vol. 3758/2005, 2005
[70] MooseFS, http://www.moosefs.org/, 2012
[71] D. Thain, C. Moretti, J. Hemmes. Chirp: A Practical Global Filesystem for Cluster and Grid Computing, JGC, Springer, 2008
[72] S. Al-Kiswany, A. Gharaibeh, M. Ripeanu. ”The Case for a Versatile Storage System”, Workshop on Hot Topics in Storage and File Systems (HotStorage09), 2009
[73] P. Druschel, A. Rowstron. ”Past: Persistent and anonymous storage in a peer-to-peer networking environment”. In Proceedings of the 8th IEEE Workshop on Hot Topics in Operating Systems (HotOS), 2001
[74] Circle, http://savannah.nongnu.org/projects/circle/, 2012
[75] J. Ousterhout, et al. The case for RAMclouds: Scalable high-performance storage entirely in DRAM. In Operating system review, 2009
[76] https://twiki.cern.ch/twiki/bin/view/CMSPublic/WorkBookCMSSWFramework
[77] Ν. Μισυρλής, Εισαγωγή στον Προγραμματισμό με την C, Παράρτημα Γ, 2002
[78] Martí J, Queralt A, Gasull D, Cortes T. Living Objects: Towards Flexible Big Data Sharing. Journal of Computer Science & Technology. 2013 ;13:56-63. Available from: http://hpc.ac.upc.edu/PDFs/dir20/file004283.pdf
[79] Franck Cappello, Al Geist, Bill Gropp, Sanjay Kale, Bill Kramer, Marc Snir, Toward Exascale Resilience, Technical Report of the INRIA-Illinois Joint Laboratory on PetaScale Computing TR-JLPC-09-01.
[80] Dengpan Yin, Esma Yildirim, and Tevfik Kosar,A Data Throughput Prediction and Optimization Service for Widely Distributed Many-Task Computing, IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, 2010.
[81] W. Allcock, J. Bresnahan, R. Kettimuthu, M. Link, C. Dumitrescu, I. Raicu, and I. Foster, “The globus striped gridftp framework and server,” in *SC ’05: Proceedings of the 2005 ACM/IEEE conference on Supercomputing*. Washington, DC, USA: IEEE Computer Society, 2005, p. 54.
[82] http://tools.ietf.org/html/rfc6749
[83] Terry Jones, Alice Koniges, and R. Kim Yates, *Performance of the IBM General Parallel File System*, Proceedings of the International Parallel and Distributed Processing Symposium, Cancun, Mexico, May 2000.
[84] Christos Filippidis, Panayiotis Tsanakas, Yiannis Cotronis, IKAROS: a Scalable I/O Framework for High-Performance Computing Systems,The Journal of Systems & Software (2016),Volume 118, pp. 277-287, DOI:http://dx.doi.org/10.1016/j.jss.2016.05.027