

---

ΧΑΡΟΚΟΠΕΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ  
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΜΑΤΙΚΗΣ

---



# ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ

«ΠΛΗΡΟΦΟΡΙΚΗ ΚΑΙ ΤΗΛΕΜΑΤΙΚΗ»

Διπλωματική Εργασία

*Ανοχή  
σε Σφάλματα σε Υπολογιστικό  
Περιβάλλον Νέφους*

ΑΘΑΝΑΣΙΟΥ Γ. ΚΩΝΣΤΑΝΤΙΝΟΣ



**Πρόγραμμα Μεταπτυχιακών Σπουδών  
«Πληροφορική και Τηλεματική»**

**Διπλωματική Εργασία**

**Ανοχή σε Σφάλματα  
σε  
Υπολογιστικό Περιβάλλον Νέφους**

Κωνσταντίνος Αθανασίου

---



Κωνσταντίνος Αθανασίου , 'Ανοχή σε σφάλματα σε υπολογιστικό περιβάλλον νέφους'

## Ανοχή σε Σφάλματα σε Υπολογιστικό Περιβάλλον Νέφους

Κωνσταντίνος Αθανασίου

Κωνσταντίνος Τσερπές

Νικολαΐδου Μάρα

Μιχαήλ Δημήτριος

Επιβλέπων Καθηγητής

Μέλος Τριμελούς Επιτροπής

Μέλος Τριμελούς Επιτροπής

**Περίληψη :** Το υπολογιστικό νέφος αποτελεί μια τεχνολογική υποδομή, η οποία έχει στόχο τη διαμοίραση υπολογιστικών πόρων όπως δίκτυο, εξυπηρετητές, εφαρμογές και υπηρεσίες, με υψηλή ευελιξία και ανάπτυξη σε σύντομο χρονικό διάστημα και χωρίς εξεζητημένες γνώσεις από πλευράς χρηστών. Το νέφος παρέχεται από εταιρίες πάροχοι, οι οποίες προσφέρουν με ελάχιστο αντίτιμο πολύ σημαντικές υπηρεσίες προς τους χρήστες. Αναφερόμαστε σε ελάχιστο αντίτιμο, καθώς η ίδια ανάπτυξη θα απαιτούσε πολλαπλάσια τιμή αν αναπτύσσονταν με τη παραδοσιακή μορφή, με άμεσο αποτέλεσμα αύξησης κόστους.

Οι χρήστες του νέφους δεν χρειάζεται να αγοράζουν λογισμικά ή φυσικά μηχανήματα, καθώς αυτή είναι η εργασία των παρόχων, παρέχουν δηλαδή αντί αυτών εικονικά μηχανήματα. Πλέον, οι χρήστες εκτελούν όλες τις διεργασίες που θα εκτελούσαν και από τα φυσικά τους μηχανήματα. Η διαφορά είναι ότι πλέον τα αρχεία τους αποθηκεύονται στο νέφος και όχι στο τοπικό μηχανήμα τους. Επιπλέον, με τη χρήση του νέφους μειώθηκε ο χώρος εγκαταστάσεων των εταιριών καθώς επίσης και οι δαπάνες αυτών.



*Κωνσταντίνος Αθανασίου , 'Ανοχή σε σφάλματα σε υπολογιστικό περιβάλλον νέφους'*

Το σημαντικό και απαραίτητο στο νέφος είναι τα δεδομένα. Η διασφάλιση ύπαρξης των δεδομένων γίνεται από τους παρόχους, εξασφαλίζοντας στους χρήστες, ότι όποτε χρειαστούν τα δεδομένα τους, αυτά θα είναι διαθέσιμα προς χρήση. Το γεγονός ότι το νέφος χρησιμοποιεί το διαδίκτυο δυσχεραίνει την αξιοπιστία ασφάλειας αυτών. Οι πάροχοι οφείλουν να πείσουν τους πελάτες τους ότι τα δεδομένα είναι ασφαλή και υπαρκτά.

Στη μέχρι τώρα πορεία του νέφους έχουν πραγματοποιηθεί μελέτες και αναλύσεις για τα δεδομένα σε αυτό. Αναπτύχθηκαν τεχνικές οι οποίες εξασφαλίζουν την ύπαρξη των δεδομένων σε περιπτώσεις ατυχημάτων σε κέντρα δεδομένων ή σφαλμάτων σε εικονικά μηχανήματα. Οι τεχνικές αυτές συνεχίζουν ακόμη και σήμερα την ανάπτυξή τους προσπαθώντας να δώσουν λύση στο πρόβλημα απώλειας δεδομένων.

Για τη καλύτερη κατανόηση του νέφους αναπτύχθηκαν από ερευνητές αντίστοιχα εργαλεία προσομοίωσης, τα οποία έχουν στόχο να παρουσιάσουν το νέφος στους ενδιαφερόμενους, προσπαθώντας να δώσουν απαντήσεις για το κόστος ανάπτυξης των μηχανημάτων, τη κατανάλωση που έχουν κατά την εκτέλεση των εργασιών και τέλος, να μπορέσουν να εφαρμόσουν τεχνικές που θα εξασφαλίζουν τα δεδομένα στο νέφος.

**Λέξεις-κλειδιά:** Υπολογιστικό Νέφος, Ανοχή σε Σφάλματα, Τεχνικές Ανοχής, Αλγόριθμος, Προσομοίωσης Νέφους.

**Περιεχόμενο:** Κείμενο, Εικόνες, Πίνακες, Εφαρμογή Εργαλείων Προσομοίωσης



*Κωνσταντίνος Αθανασίου , 'Ανοχή σε σφάλματα σε υπολογιστικό περιβάλλον νέφους'*

## **Fault Tolerance in Cloud Computing**

Constantinos Athanasiou

Konstantinos Tserpes

Nikolaidou Mara

Michail Dimitris

**Supervisor**

**Trilateral Commission Member**

**Trilateral Commission Member**

**Abstract:** Cloud computing is the central online available computing resources such as network servers, applications and services, with high flexibility and growth in a short time and without sophisticated knowledge from users. The cloud is provided by companies' - providers, which offer very important services to users to a minimum price. The reference to the minimum price is that they buy the same tools and development, which they require, multiple price, and as a result the cost of these direct increase.

Users of the cloud do not need to buy software or physical machines, this is the work of providers, provide that instead virtual machines. Plus, users perform all processes to be executed by their physical machines. The difference is that now their files are being stored in the cloud and not on the local machine. In addition, using the Cloud decreased the space facilities of the companies as well as the costs of these.



*Κωνσταντίνος Αθανασίου , 'Ανοχή σε σφάλματα σε υπολογιστικό περιβάλλον νέφους'*

The important and necessary in cloud is the data. Ensuring the existence of the data is done by the providers, ensuring users that whenever they need, these data are available for use. The fact that the cloud is using the Internet complicates the credibility of security. The providers have to convince their customers that the data are safe and existent.

Until now the cloud studies and analyzes of the data have taken place in the cloud. Developed techniques ensure that the data exist in cases of accidents in data centers or errors in virtual machines. These techniques continue even today their development by trying to solve the data loss problem.

For better understanding of the cloud, researchers have developed respective simulation tools, which are intended to illustrate the cloud to stakeholders, trying to give answers to the development cost of machinery, the consumption of which have in the performance of work and finally how they can implement techniques to ensure the data in the cloud.

**Key Words:** Cloud Computing, Fault Tolerance, Fault Tolerance Techniques, Algorithm, Cloud Simulation

**Content:** Text, Images, Tables, Cloud Computing Simulators



## **Summary**

Cloud Computing has become a new paradigm for hosting and delivering services over the internet. Nowadays, users of the internet create accounts to providers in order to create virtual machines or upload files.

Cloud computing has recently emerged as a new paradigm for hosting and delivering services over the Internet. Business are taken into account the use of cloud computing because is very attractive to their owners as it eliminates the requirement for users to plan ahead for provisioning, and allows enterprises to start from the small and increase resources only when there is a rise in service demand. Using the cloud, you can easily scale up or down as fast as you want resources, in order to meet your demand. Also, using cloud, you pay as you use. There is always a metered service to show your consuming recourses. On the other hand, cloud computing is still at its early, because it has to manage many issues, such that failures or security of data.

In this post graduate thesis, there is an effort in order to expound the problem of failures in the cloud. This issue, called fault tolerance in cloud computing, is difficult since cloud computing relies by nature on a complex splitting into many layers, which are hosts, virtual machines and applications or services. This paper presents the meaning of fault tolerance in cloud computing, providing the types of faults, which may be done when using cloud services. Furthermore, in order to avoid data loss, there is a reference to algorithms, which the scopes of these are to prevent data loss. The right way to do that is to implement techniques during the use of cloud by the providers. Fault tolerance provides the ability of a system in the cloud to respond effectively to an unexpected hardware or software failure. This paper provide to readers a better understanding of fault tolerance along with some existing and published algorithms, which refers to the saving of data loss during the use of the cloud, when something goes wrong.

Moreover, an objective of this thesis it to provide a generalized and extensible simulation framework that enables seamless modeling, simulation, and experimentation of emerging Cloud computing infrastructures and application services. Researchers and industry-based developers can focus on specific system design issues that they want to investigate, without getting concerned about the low level details related to cloud-based infrastructures and services.

As a result, using simulators, researchers have the opportunity to implement their studies to them, in order to find solutions to the problem of failures and data loss.

In this paper, there is a full presentation of two simulators, which main work are to simulate cloud, creating hosts, virtual machines and running jobs or applications.

Whilst a lot of development and progress has already been made in cloud technologies, there still remains a wide range of concerns that need to be addressed in future cloud iterations in order to reach its full potentials.



## Περιεχόμενα

1. Εισαγωγή.....	14
1.1. Γενικά.....	14
1.2. Βασική κατεύθυνση.....	14
1.3. Περιγραφή της σημερινής κατάστασης.....	15
1.4. Στόχος διπλωματικής εργασίας.....	17
1.5. Συνοπτική περιγραφή των κεφαλαίων της εργασίας.....	17
1.6. Λέξεις κλειδιά.....	18
2. Cloud Computing .....	19
2.1. Ορισμός.....	19
2.2. Αρχιτεκτονική του «Cloud» και οι Προσφερόμενες Υπηρεσίες.....	23
2.3. Πλεονεκτήματα-Κίνδυνοι.....	28
2.3.1. Πλεονεκτήματα και Ανησυχίες .....	28
2.3.2. Κίνδυνοι.....	30
2.3.3. Μετεγκατάσταση Εταιριών στο Νέφος .....	30
3. Ανοχή σε Σφάλματα (Fault Tolerance) .....	33
3.1. Έννοια Ανοχής Σφαλμάτων (Fault Tolerance).....	33
3.2. Τύποι Σφαλμάτων.....	34
3.3. Τεχνικές.....	35
3.4. Αλγόριθμοι .....	36
3.4.1. Αλγοριθμική Προσέγγιση.....	36
3.4.2. «Corryset Replication».....	37
3.4.3. Αλγόριθμος QADR .....	42
3.5. Προσομοίωση στο Νέφος.....	46
4. Εργαλείο Προσομοίωσης «CloudSim» .....	47
4.1. Περιγραφή .....	47
4.2. Αρχιτεκτονική .....	47
4.3. Σχεδιασμός-Βασικές Δομές.....	51
4.4. Εγκατάσταση - Εκτέλεση του «CloudSim» .....	55
5. Εργαλείο Προσομοίωσης «DCSim» .....	63
5.1. Περιγραφή .....	63
5.2. Αρχιτεκτονική .....	63
5.3. Σχεδιασμός-Βασικές Δομές.....	64
5.4. Εγκατάσταση – Εκτέλεση του «DCSim».....	68





*Κωνσταντίνος Αθανασίου , 'Ανοχή σε σφάλματα σε υπολογιστικό περιβάλλον νέφους'*

6. Συμπέρασμα - Επέκταση.....	69
6.1. Συμπέρασμα .....	69
6.2. Επέκταση.....	70
Βιβλιογραφία.....	71



## Εικόνες

Εικόνα 1. "Gartner Hype Cycle" .....	20
Εικόνα 2. Εξέλιξη του νέφους 2008[11] .....	21
Εικόνα 3. Εξέλιξη του νέφους 2010 .....	21
Εικόνα 4. Εξέλιξη του νέφους 2012 .....	22
Εικόνα 5. Εξέλιξη του νέφους 2014 .....	23
Εικόνα 6. Αρχιτεκτονική Cloud .....	24
Εικόνα 7. Σχέση Χρήστη-Παρόχου .....	26
Εικόνα 8. SaaS, PaaS, IaaS .....	26
Εικόνα 9. Σχέση μεταξύ χρηστών και παρόχων .....	28
Εικόνα 10. Τυχαία Αντιγραφή (α) .....	38
Εικόνα 11. Τυχαία Αντιγραφή (β) .....	38
Εικόνα 12. Τυχαία Αντιγραφή (γ) .....	38
Εικόνα 13. Τυχαία Αντιγραφή (δ) .....	38
Εικόνα 14. Φάση Μετάθεσης .....	39
Εικόνα 15. Φάση Αντιγραφής .....	40
Εικόνα 16.(α) Πιθανότητα Απώλειας Δεδομένων με Σύγκριση Τεχνικών Random και Copyset Replication .....	40
Εικόνα 17.(β) Πιθανότητα Απώλειας Δεδομένων με Σύγκριση Τεχνικών Random και Copyset Replication .....	40
Εικόνα 18.(γ) Πιθανότητα Απώλειας Δεδομένων με Σύγκριση Τεχνικών Random και Copyset Replication .....	41
Εικόνα 19. Αρχιτεκτονική Hadoop .....	43
Εικόνα 20. Δομή QADR Αλγορίθμου .....	44
Εικόνα 21. Χρόνος πρόσβασης αντιγραφής μεταξύ κόμβων .....	44
Εικόνα 22. Γράφημα Περιγραφής Κόμβων .....	44
Εικόνα 23. Συνολικό κόστος αντιγραφής .....	45
Εικόνα 24. Αναλογία Κόστους .....	45
Εικόνα 25. Δομή Νέφους .....	48
Εικόνα 26. Δομή CloudSim προηγούμενης έκδοσης .....	49
Εικόνα 27. Αρχιτεκτονική CloudSim .....	50
Εικόνα 28. Διάγραμμα Κλάσης CloudSim .....	53
Εικόνα 29. Κύκλος Ζωής CloudSim [28] .....	55
Εικόνα 30. Λήψη CloudSim .....	56
Εικόνα 31. Αποθήκευση σε τοπικό ή φορητό μηχάνημα .....	56
Εικόνα 32. Λειτουργία IDE .....	57
Εικόνα 33. Δημιουργία Νέου Έργου(α) .....	57
Εικόνα 34. Δημιουργία Νέου Έργου(β) .....	58
Εικόνα 35. Δημιουργία Νέου Έργου(γ) .....	58
Εικόνα 36. Εισαγωγή Βιβλιοθηκών (α) .....	59
Εικόνα 37. Εισαγωγή Βιβλιοθηκών (β) .....	59
Εικόνα 38. Προβολή Οντοτήτων - Κλάσεων .....	60
Εικόνα 39. Εκτέλεση Προσομοίωσης .....	61
Εικόνα 40. Προβολή Αποτελεσμάτων .....	62
Εικόνα 41. DCSim Μοντέλο .....	63
Εικόνα 42. Λήψη του "DCSim" .....	68



*Κωνσταντίνος Αθανασίου , 'Ανοχή σε σφάλματα σε υπολογιστικό περιβάλλον νέφους'*



## Πίνακες

Πίνακας 1. Απόψεις Στελεχών Επιχειρήσεων για το Νέφος.....	16
Πίνακας 2. Οδηγίες Ασφαλείας Χρήσης Νέφους.....	31
Πίνακας 3. Απόφασης Μεταβολής στο Νέφος.....	31
Πίνακας 4. Πλεονεκτήματα Νέφους .....	32
Πίνακας 5. Είδη Σφαλμάτων .....	34
Πίνακας 6. Χαρακτηρισμός Σφαλμάτων.....	35



*Κωνσταντίνος Αθανασίου , 'Ανοχή σε σφάλματα σε υπολογιστικό περιβάλλον νέφους'*

© Χαροκόπειο Πανεπιστήμιο Αθηνών, 2015

Η παρούσα διατριβή, η οποία εκπονήθηκε στα πλαίσια Διπλωματικής Εργασίας, και τα λοιπά αποτελέσματα αυτής αποτελούν συνιδιοκτησία του Χαροκοπείου και του φοιτητή, ο καθένας από τους οποίους έχει το δικαίωμα ανεξάρτητης χρήσης και αναπαραγωγής τους (στο σύνολο ή τμηματικά) για διδακτικούς και ερευνητικούς σκοπούς, σε κάθε περίπτωση αναφέροντας τον τίτλο και το συγγραφέα και το Χαροκόπειο Πανεπιστήμιο όπου εκπονήθηκε η Διπλωματική Εργασία καθώς και τον επιβλέποντα και την επιτροπή κρίσης.



## 1. Εισαγωγή

### 1.1. Γενικά

Η πληροφορική ή επιστήμη υπολογιστών αποτελεί μια θετική και εφαρμοσμένη επιστήμη, ο ρόλος της οποίας είναι η έρευνα των πληροφοριών, των αλγορίθμων και των υπολογισμών. Η έρευνα αυτή, στηρίζεται σε τεχνολογίες ή τεχνολογικές εφαρμογές, οι οποίες αποτελούν υπολογιστικά συστήματα. Η σχεδίαση, η ανάπτυξη, η υλοποίηση, η διερεύνηση, η ανάλυση αυτών των συστημάτων έχει ως πυλώνα τη χρήση υπολογιστών, στους οποίους οι χρήστες εισάγουν δεδομένα εισόδου και δέχονται ως αποτελέσματα μια πιθανή έξοδο, η οποία αποτελεί κωδικοποιημένη πληροφορία, ικανή να διαχειριστεί και να επεξεργαστεί όχι μόνο από τους επιστήμονες που έχουν ως βασικό επιστημονικό πεδίο την πληροφορική, αλλά και από διάφορους ανθρώπους και εταιρίες σε διάφορους τομείς, όπως οικονομία, κοινωνικές υπηρεσίες, διασκέδαση. [1]

Στην αγορά εργασίας υπάρχουν εταιρίες διαφόρων τύπων, οι οποίες κάνουν χρήση των υπολογιστών και ιδιαίτερα των προγραμμάτων που αναπτύσσονται με ραγδαίο ρυθμό. Εταιρίες, οι οποίες σχεδιάζουν, αναπτύσσουν και υλοποιούν προϊόντα πληροφορικής είτε πρόκειται για λογισμικό είτε για υλικό υπολογιστών, όπως λειτουργικά συστήματα, βάσεις δεδομένων, πληροφοριακά συστήματα, συστήματα διαχείρισης αποφάσεων ή βιομηχανικά συστήματα έχουν ανάγκη τη χρήση υπολογιστών, η οποία θα εξασφαλιστεί με όσο το δυνατό μικρότερο κόστος. Στην ίδια κατάσταση, εξαιτίας της σημερινής οικονομικής κρίσης έχουν βρεθεί πληθώρα των εταιριών που δραστηριοποιούνται στο ηλεκτρονικό εμπόριο ή σε μεγάλα επιστημονικά και ερευνητικά κέντρα. Ο συνεχόμενος ρυθμός χρήσης του διαδικτύου αυξάνει την επιθυμία των εταιριών να μεγαλώσουν ακόμη περισσότερο τον αριθμό και τη δυναμική των υπολογιστών που έχουν στη διάθεσή τους, ώστε να μπορέσουν και αυτές με τη σειρά τους να αποτελέσουν τον καλύτερο ανταγωνιστή σε σχέση με τα προϊόντα ή τα έργα που διοχετεύουν στην αγορά.

### 1.2. Βασική κατεύθυνση

Η ραγδαία εξέλιξη της πληροφορικής, τόσο σε θεωρητικό αλλά και σε τεχνολογικό πλαίσιο έρχεται να προσθέσει ακόμη μια δυσκολία, στην ήδη υπάρχουσα, δηλαδή στο κόστος, στη ποιότητα και στο χώρο που αυτοί λαμβάνουν. Τα αποτελέσματα της εξελεγκτικής πορείας της τεχνολογίας οδήγησε στη δημιουργία του περιβάλλοντος νέφους. Το νέφος αποτελεί έναν όρο που συνηθίζεται να προσφωνείται και να αναγράφεται τόσο από ομιλητές όσο και σε επιστημονικά άρθρα ως «Cloud Computing». Το «Cloud Computing» συνήθως ορίζεται ως ένα είδος υπολογιστή που βασίζεται σε κοινή χρήση υπολογιστικών πόρων και όχι σε τοπικούς διανομείς («Servers») ή προσωπικές συσκευές με σκοπό να χειριστεί εφαρμογές ή προγράμματα. [2]

Το «Cloud Computing», αποτελεί ένα «μακρινό όνειρο» των υπολογιστών, ως ένα βοηθητικό πρόγραμμα, το οποίο έχει τη δυνατότητα να μετατρέψει ένα μεγάλο μέρος της βιομηχανίας πληροφορικής, καθιστώντας το λογισμικό ακόμη περισσότερο ελκυστικό ως υπηρεσία και να διαμορφώσει το υλικό της πληροφορικής με κατάλληλο σχεδιασμό ώστε να μπορέσει να αγοραστεί από όλους. Οι επιστήμονες ή προγραμματιστές που διαθέτουν καινοτόμες ιδέες αλλά μικρό ποσό διαθέσιμο, για δαπάνες, για υλικό, μπορούν πλέον να



αναπτύξουν τις υπηρεσίες τους πιο γρήγορα και πιο εύκολα με λιγότερο κόστος σε σχέση με τη διάθεση υλικού και λογισμικού με τη παραδοσιακή μορφή, δηλαδή με την εγκατάσταση διανομέων («Servers») και λογισμικού σε πραγματικούς χώρους. [2]

Η Google [3], η Yahoo [4], Amazon [5], και άλλοι έχουν δημιουργήσει μεγάλες, ειδικά κατασκευασμένες αρχιτεκτονικές για την υποστήριξη των εφαρμογών τους, καθώς επίσης παρέχουν την αναγκαία εκπαίδευση στον υπόλοιπο κόσμο στη κατηγορία δημιουργίας αρχιτεκτονικών για την υποστήριξη υπηρεσιών που έχουν σχέση με την υπολογιστική ισχύ, την αποθήκευση και την εφαρμογή σε περιβάλλον «Cloud». [6]

Το «Cloud Computing» δημιουργεί σχετική ευκολία στη μετακίνηση των υπηρεσιών, εγγράφων, υπολογισμών ή στοιχεία κόστους προς όφελος των επιχειρήσεων σε διαφανή θέση σε συνεργασία με ένα πάροχο, με αρκετά μικρό αντίτιμο. Αναπτυγμένες εταιρίες μπορούν να δημιουργήσουν το δικό τους «Cloud» κάνοντας χρήση των δικών τους υπηρεσιών, χωρίς ενδιάμεσο πάροχο. [6] Κάνοντας διαθέσιμα τα δεδομένα στο «Cloud», υπάρχει η δυνατότητα εύκολης πρόσβασης και διαχείρισης αυτών από οπουδήποτε, επιτρέποντας με αυτό τον τρόπο ευκαιρίες για ενισχυμένη συνεργασία μεταξύ ερευνητικών ομάδων ή στη σχέση πελατών-εταιρίας.

### 1.3. Περιγραφή της σημερινής κατάστασης

Το «Cloud Computing» αποτελεί μία τάση των ερευνητών, χρηστών και εταιριών στο τομέα της πληροφορικής και γενικότερα στο τομέα των προσφερόμενων υπηρεσιών μέσω διαδικτύου να διαχειρίζονται τα δεδομένα από απόσταση. Διαχειρίζονται επίσης υπηρεσίες και προγράμματα τα οποία είναι εγκατεστημένα σε εικονικά μηχανήματα, πάντα όμως με τη χρήση του διαδικτύου. [7] Η τάση αυτή, η οποία έχει υιοθετηθεί ευρέως από τη βιομηχανία, παρά το γεγονός ότι είναι γρήγορη και διαθέτει αρκετά πλεονεκτήματα σε σχέση με τη παραδοσιακή εγκατάσταση υλικού και λογισμικού διαθέτει και προβλήματα. Τα προβλήματα προκύπτουν από το κενό σε ερευνητικά ερωτήματα που θα πρέπει να αντιμετωπιστούν από τη διεθνή κοινότητα. Ερωτήματα όπως η ανοχή σφαλμάτων («Fault Tolerance»), προγραμματιστική ροή εργασίας («Workflow Scheduling»), διαχείρισης ροής εργασίας («Workflow Management»), ασφάλεια («Security») χρίζουν αντιμετώπισης, ώστε οι χρήστες του «Cloud» να αισθάνονται ακόμη περισσότερο ασφάλεια για τα δεδομένα τους. [8] Η ανοχή σφαλμάτων είναι ένα από τα βασικά ζητήματα μεταξύ όλων, καθώς ασχολείται με ένα πολύ ευαίσθητο κομμάτι επεξεργασίας, τα δεδομένα. Αυτά μπορεί να είναι από απλό κείμενο, χωρίς ιδιαίτερη λεκτική σημασία έως και πολύ προσωπικά δεδομένα συναλλαγών, κειμένων ή πολυμέσων. Η ανοχή σε σφάλματα αναφέρεται σε τεχνικές που αναπτύσσονται από επιστήμονες ή ομάδες επιστημόνων σε Πανεπιστήμια και ερευνητικά κέντρα, με σκοπό ένα σύστημα σε περιβάλλον «Cloud» να μπορέσει να αμυνθεί σε ενδεχόμενα προβλήματα που μπορεί να προκύψουν είτε από το υλικό είτε από το λογισμικό. Οι τεχνικές αυτές δηλαδή, αποτρέπουν στο να χαθούν δεδομένα και μέσω διαφόρων αλγορίθμων ανακτούν τα δεδομένα αυτά και τα προωθούν στο προορισμό τους. Στα κύρια οφέλη της εφαρμογής της ανοχής σε σφάλματα περιλαμβάνονται η αποκατάσταση βλάβης, το χαμηλό κόστος ανάκτησης δεδομένων και η καλύτερη απόδοση του συστήματος. Σε περίπτωση που δεν υπήρχαν οι μηχανισμοί ανάκτησης, τότε όλα τα δεδομένα θα έπρεπε να υπήρχαν ως αντίγραφα σε δίσκους, γεγονός που αποτελεί ιδιαίτερα μεγάλο πρόβλημα από πλευράς κόστους και μεγέθους εξαιτίας του μεγάλου όγκου των δεδομένων. Ένα επιπλέον χαρακτηριστικό της



Κωνσταντίνος Αθανασίου , 'Ανοχή σε σφάλματα σε υπολογιστικό περιβάλλον νέφους'

ανοχής σε σφάλματα είναι ότι αποτελεί μια πολύ σημαντική ανησυχία για τη διασφάλιση της διαθεσιμότητας και της αξιοπιστίας των κρίσιμων υπηρεσιών και των εφαρμογών που εκτελούνται κάθε φορά στο περιβάλλον «Cloud».

Υπάρχει μεγάλη ποικιλία άρθρων, μελετών σε συνέδρια και σε ημερίδες σχετικά με το περιβάλλον νέφους. Παρόλα αυτά παραμένει ακόμη σε σύγχυση το γεγονός αν είναι αποδοτικό μέσο για την διασφάλιση των δεδομένων, αν είναι χρήσιμο, ποιος ο ακριβής ορισμός του και τι ακριβώς κάνει. [9]

Στον παρακάτω πίνακα παραθέτουμε απόψεις στελεχών επιχειρήσεων, πως αυτοί αντιλαμβάνονται το περιβάλλον νέφους και πως ενστερνίζονται τη μετάβαση σε υπηρεσίες «Cloud». [9]

**Πίνακας 1. Απόψεις Στελεχών Επιχειρήσεων για το Νέφος**

Ελεύθερη Μετάφραση	Αγγλική Αποτύπωση
<p><i>Το ενδιαφέρον σχετικά με το περιβάλλον νέφους είναι ότι έχουμε επαναπροσδιορίσει ότι το περιβάλλον νέφους περιλαμβάνει ότι μέχρι σήμερα κάνουμε... Δεν κατανοώ τι διαφορετικό θα έπρεπε να κάνουμε εκτός από τη διαφορετική διατύπωση των διαφημίσεων.</i></p> <p><i>Larry Ellison, quoted in the Wall Street Journal, September 26, 2008</i></p>	<p><i>The interesting thing about Cloud Computing is that we've redefined Cloud Computing to include everything that we already do. . . . I don't understand what we would do differently in the light of Cloud Computing other than change the wording of some of our ads.</i></p>
<p><i>Πολλοί άνθρωποι μεταβαίνουν σε περιβάλλον νέφους, διότι είναι μόδα, αλλά δεν έχω ακούσει δύο άτομα να λένε το ίδιο πράγμα για το ίδιο γεγονός. Υπάρχουν πολλαπλοί ορισμοί για το «νέφος»</i></p> <p><i>Andy Isherwood, quoted in ZDnet News, December 11, 2008</i></p>	<p><i>A lot of people are jumping on the [cloud] bandwagon, but I have not heard two people say the same thing about it. There are multiple definitions out there of "the cloud."</i></p>
<p><i>Είναι ηλιθιότητα. Είναι χειρότερο από ηλιθιότητα: είναι διαφημιστική εκστρατεία. Κάποιος λέει ότι είναι αναπόφευκτο – και οπουδήποτε αν ακούσεις κάποιον να λέει ότι, είναι πιθανό να αποτελεί επιχειρηματική εκστρατεία ώστε να επιτύχει.</i></p> <p><i>Richard Stallman, quoted in The Guardian, September 29, 2008</i></p>	<p><i>It's stupidity. It's worse than stupidity: it's a marketing hype campaign. Somebody is saying this is inevitable — and whenever you hear somebody saying that, it's very likely to be a set of businesses campaigning to make it true.</i></p>





#### 1.4. Στόχος διπλωματικής εργασίας

Η ανοχή σε σφάλματα σε περιβάλλον νέφους πρέπει να συνεχίσει να εκτελείται σωστά και αποδοτικά με την παρουσία ρηγμάτων. Η ανοχή σε σφάλματα διαθέτει όλες εκείνες τις τεχνικές που αναπόφευκτα επιτρέπουν την ευρωστία και αξιοπιστία του συστήματος. Τα κύρια οφέλη εκτός των όσων προαναφέρθηκαν (1.3) συγκαταλέγεται η ανθεκτικότητα της υπηρεσίας σε δυσμενή κατάσταση λόγω προβλημάτων του συστήματος (π.χ. πτώση τάσης). Η εξάρτηση της υπηρεσίας είναι άμεση με την ποιότητα αυτής («QoS-Quality of Services»), οπότε συμπεραίνουμε ότι η ανοχή σε σφάλματα είναι εξαρτώμενη από τη αξιοπιστία και τη διαθεσιμότητα της υπηρεσίας του συστήματος.

Οι τεχνικές ανοχής σε σφάλματα αποτελούνται από δύο φάσεις. Η πρώτη αναφέρεται «Αποτελεσματική διαδικασία εύρεσης σφάλματος», η οποία στοχεύει στο να βρίσκει το σφάλμα και να το επαναφέρει στην αρχική του κατάσταση. Η δεύτερη φάση αναφέρεται «Διαδικασία διόρθωσης σφάλματος», η οποία έχει στόχο να εξασφαλίσει ότι το σφάλμα δεν θα επαναληφθεί. [10]. Για τη καλύτερη και ποιο αποδοτική εφαρμογή των τεχνικών ανοχής σε σφάλματα, έχουν αναπτυχθεί αλγόριθμοι με βασικό αντικείμενο την εξάλειψη των δυσμενών αποτελεσμάτων από πιθανά σφάλματα στο σύστημα που διαθέτει δεδομένα ή εκτελεί υπηρεσίες. Στόχος της διπλωματικής εργασίας είναι η μελέτη, ανάλυση, υλοποίηση και σύγκριση αλγορίθμων ανοχής σε σφάλματα σε περιβάλλον υπολογιστικού νέφους. Η μελέτη, ανάλυση, σύγκριση των αλγορίθμων θα πραγματοποιηθεί με την ήδη υπάρχουσα βιβλιογραφία των αντίστοιχων άρθρων. Η ολοκλήρωση της εργασίας θα γίνει με την παρουσίαση εργαλείων προσομοίωσης περιβάλλον υπολογιστικού νέφους, τα οποία θα υπολογίζουν το χρόνο εκτέλεσης των εργασιών σε εικονικά μηχανήματα, τα οποία διαθέτουν υπολογιστικούς πόρους.

#### 1.5. Συνοπτική περιγραφή των κεφαλαίων της εργασίας

Για την αποτελεσματικότερη κατανόηση της παρούσας διπλωματικής εργασίας, έγινε προσπάθεια διαχωρισμού των κεφαλαίων με τέτοιο τρόπο ώστε η ροή αυτής να δίνει στον αναγνώστη τη δυνατότητα να κατανοήσει σε βάθος τις έννοιες υπολογιστικό νέφος και ανοχή σε σφάλματα. Αναλυτικότερα:

- Κεφάλαιο 1: Παρουσιάζεται η γενική εικόνα της διπλωματικής, η κατεύθυνση στην οποία κινείται, ο στόχος αυτής, καθώς και το παρών υπό-κεφάλαιο.
- Κεφάλαιο 2: Παρουσιάζεται ο ορισμός του υπολογιστικού νέφους («Cloud Computing»), τα πλεονεκτήματα και μειονεκτήματα προς τους χρήστες και εταιρίες.
- Κεφάλαιο 3: Παρουσιάζεται η έννοια της ανοχής σε σφάλματα σε περιβάλλον υπολογιστικού νέφους. Επιπλέον, αναφέρονται οι τύποι σφαλμάτων και οι τεχνικές που χρησιμοποιούνται, ώστε να αποφεύγονται τα διάφορα σφάλματα που προκύπτουν κατά τη εκτέλεση εργασιών σε ένα κέντρο δεδομένων.
- Κεφάλαιο 4: Σε αυτά τα κεφάλαια, παρουσιάζεται το εργαλείο προσομοίωσης του νέφους, με ονομασία «CloudSim». Αναγράφονται η αρχιτεκτονική του, οι βασικές δομές που περιέχει και πραγματοποιείται εκτέλεση αυτού, εμφανίζοντας σε πίνακες τα αποτελέσματα αυτής.
- Κεφάλαιο **Ошибка! Источник ссылки не найден.**: Αντίστοιχα με το κεφάλαιο 4, παρουσιάζεται το εργαλείο προσομοίωσης με ονομασία «DCSim.»



*Κωνσταντίνος Αθανασίου , 'Ανοχή σε σφάλματα σε υπολογιστικό περιβάλλον νέφους'*

- Κεφάλαιο 5: Αναγράφεται το συμπέρασμα στο οποίο καταλήγει η διπλωματική και παρουσιάζεται μελλοντική εργασία.

### **1.6. Λέξεις κλειδιά**

Περιβάλλον νέφους, «Cloud», «Cloud Computing», Ανοχή σε σφάλματα, «Fault Tolerance», αλγόριθμος.



## 2. Cloud Computing

### 2.1. Ορισμός

Στο «Cloud Computing», το «Cloud» αποτελεί τη λέξη σύννεφο, χρησιμοποιείται ως μεταφορά για το διαδίκτυο, έτσι ώστε η φράση «Cloud Computing» να αποτελεί είναι είδος που βασίζεται στο προγραμματιστικό διαδίκτυο, «Internet Computing», όπου διάφορες υπηρεσίες και διακομιστές, εκτελούν προγράμματα και αποθηκεύουν δεδομένα μέσω του διαδικτύου. Το «Cloud Computing» αποτελεί μία από τις πιο ασαφείς τεχνικές ορολογίες στην ιστορία. [11] Ένας λόγος είναι ότι μπορεί να χρησιμοποιηθεί σε πολλές εφαρμογές, σενάρια. Επιπλέον, υπάρχει φόβος και ανησυχία από τις εταιρίες να υιοθετήσουν το «καινούργιο», οπότε υπάρχει η προκατάληψη από πολλές εταιρίες για τη χρήση του νέφους.

Η εξέλιξη του «Cloud Computing» αποτυπώνεται και αναλύεται από το διάγραμμα «Gartner», όπως απεικονίζεται και στις παρακάτω εικόνες.. Το διάγραμμα, «Gartner» γενικότερα, αποτελεί μια γραφική αναπαράσταση της ωριμότητας και της υιοθέτησης των τεχνολογιών και εφαρμογών με σκοπό την επίλυση των πραγματικών προβλημάτων και δυσχερειών των επιχειρήσεων για την αξιοποίηση των νέων ευκαιριών. Αποτελεί μία μεθοδολογία που παρουσιάζει την εξέλιξη μια τεχνολογίας ή εφαρμογής σε σχέση με τη πάροδο του χρόνου. Οι χρήστες μια τεχνολογίας χρησιμοποιούν το διάγραμμα, ώστε να εκπαιδεύονται σε σχέση με την εξέλιξη μιας αναδυόμενης τεχνολογίας στο πλαίσιο της βιομηχανίας και της διάθεσης ανάληψης κινδύνων που μπορεί να προκύψουν με τη χρήση αυτής. Η έγκαιρη ανάλυση του διαγράμματος σε πρώιμο στάδιο της τεχνολογίας μπορεί να οδηγήσει τις εταιρίες σε επωφελή κέρδη, χωρίς ωστόσο η ανάλυση αυτή να αποτελεί κανόνα, καθώς πρόκειται για ριψοκίνδυνες ενέργειες.

Πιο συγκεκριμένα, το διάγραμμα αναλύεται σε περιόδους εξέλιξης και αποδοχής της υπηρεσίας ή της τεχνολογίας. Η πρώτη περίοδος αναφέρεται στην ύπαρξη της καινοτομίας ή νέας ιδέας. Προσπαθεί να αποδώσει ένα απλό επιχείρημα έγκαιρης επένδυσης, αλλά με επιμονή στην ανάλυση σχέσης οφέλους - κόστους, η οποία θα πρέπει να γίνεται με νέους τρόπους και διαδικασίες που ακόμη, στη παρούσα περίοδο εξέλιξης της τεχνολογίας δεν έχει ακόμη αποδειχθεί. Στην Εικόνα 1. "Gartner Hype Cycle" παρουσιάζεται η γενική μορφή του κύκλου του διαγράμματος του «Gartner», το οποίο αναλύεται σε πέντε βασικά στάδια εξέλιξης μιας καινοτόμας τεχνολογίας στη πορεία του χρόνου.



Εικόνα 1. "Gartner Hype Cycle"

Το στάδιο ύπαρξης καινοτόμας ιδέας και τεχνολογίας («Technology Trigger»), η οποία είναι σε πρώιμο στάδιο, χωρίς σημαντική δημοσιότητα και χωρίς εύχρηστα προϊόντα ή υπηρεσίες.

Το στάδιο προσδοκιών («Peak of Inflated Expectations»), στο οποίο δίνεται μια πρώτη δημοσιότητα της νέας ιδέας – τεχνολογίας και δίνονται στο κοινό τα πρώτα αποτελέσματα εφαρμογής στους ενδιαφερόμενους. Επίσης, παρέχεται στο στάδιο αυτό και το ιστορικό τόσο της επιτυχίας όσο και της αποτυχίας.

Το επόμενο στάδιο, το στάδιο της απογοήτευσης («Trough of Disillusionment»), στο οποίο εξασθενεί το ενδιαφέρον του κοινού για τα αποτελέσματα της χρήσης της νέας τεχνολογίας. Οι επενδύσεις σε αυτό το στάδιο γίνονται για τη συνέχισης της ιδέας και τη βελτίωση αυτής, ώστε οι ενδιαφερόμενοι να απολαμβάνουν πλήρως και χωρίς σφάλματα τη νέα αυτή τεχνολογία.

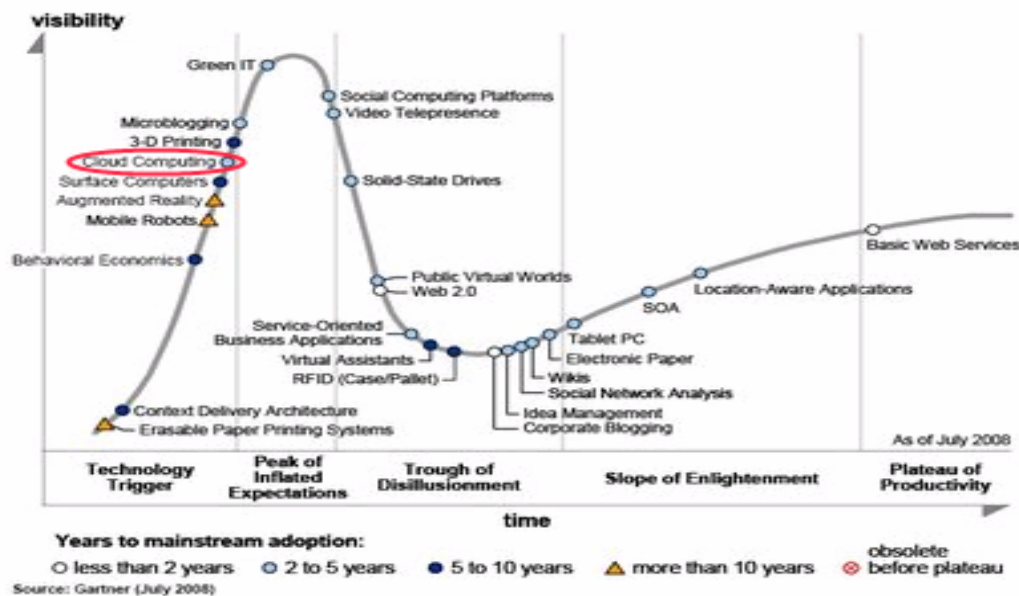
Στο στάδιο κλίσης διαφωτισμού («Slope of Enlightenment») πραγματοποιείται προσπάθεια των ενδιαφερομένων να παρουσιάσουν τη τεχνολογία στις επιχειρήσεις και πως αυτές μπορούν να επωφεληθούν από τα θετικά στοιχεία της τεχνολογίας. Στο στάδιο αυτό παρουσιάζονται πλέον τα νέα προϊόντα που έχουν αναπτυχθεί από τη χρήση της νέας τεχνολογίας.

Το τελευταίο στάδιο αποτελεί το στάδιο της απόλυτης παραγωγής («Plateau of Productivity»), στο οποίο η νέα τεχνολογία απογειώνεται και εφαρμόζεται χωρίς φόβο σε όλες τις διαδικασίες των εφαρμογών ή προσφερόμενων υπηρεσιών. Στο στάδιο αυτό, η νέα τεχνολογία βρίσκεται σε παραγωγικό επίπεδο και αποδίδει κέρδη.

Στην Εικόνα 2. Εξέλιξη του νέφους 2008, παρουσιάζεται το νέφος ανάμεσα στο πρώτο και δεύτερο στάδιο εξέλιξης του. Το νέφος πλέον προσπαθεί να γίνει μια δημοσιευμένη καινοτόμα ιδέα και να παρουσιάσει τα πρώτα στάδια εκτέλεσής του.

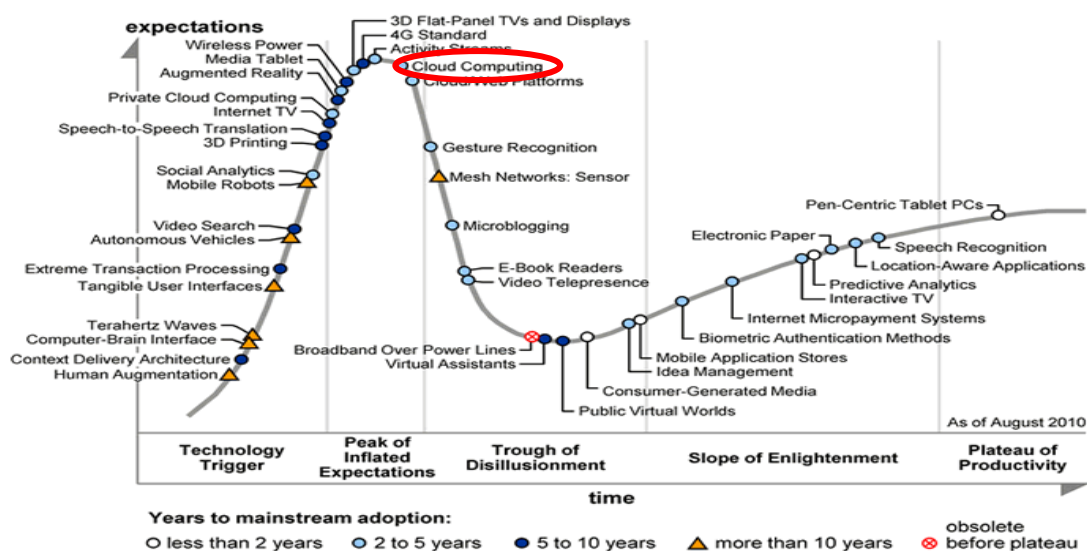


Κωνσταντίνος Αθανασίου, 'Ανοχή σε σφάλματα σε υπολογιστικό περιβάλλον νέφους'



Εικόνα 2. Εξέλιξη του νέφους 2008[11]

Στην Εικόνα 3. Εξέλιξη του νέφους 2010, παρουσιάζεται η εξέλιξη του νέφους τον Αύγουστο 2010. Το νέφος, παρατηρείται να επεκτείνεται συνεχώς βγαίνοντας από το στάδιο προσδοκιών και κάνοντας την είσοδό του στο επόμενο στάδιο, δηλαδή του «Trough of Disillusionment». Οι ενδιαφερόμενοι πλέον μπορούν να εμπιστευτούν την τεχνολογία του νέφους και να στηρίξουν τις επενδυτικές τους πολιτικές σε αυτό. Το ενδιαφέρον για την τεχνολογία του νέφους δρα αυξητικά και η τεχνολογία αποδίδει αποτελεσματικά.



Εικόνα 3. Εξέλιξη του νέφους 2010

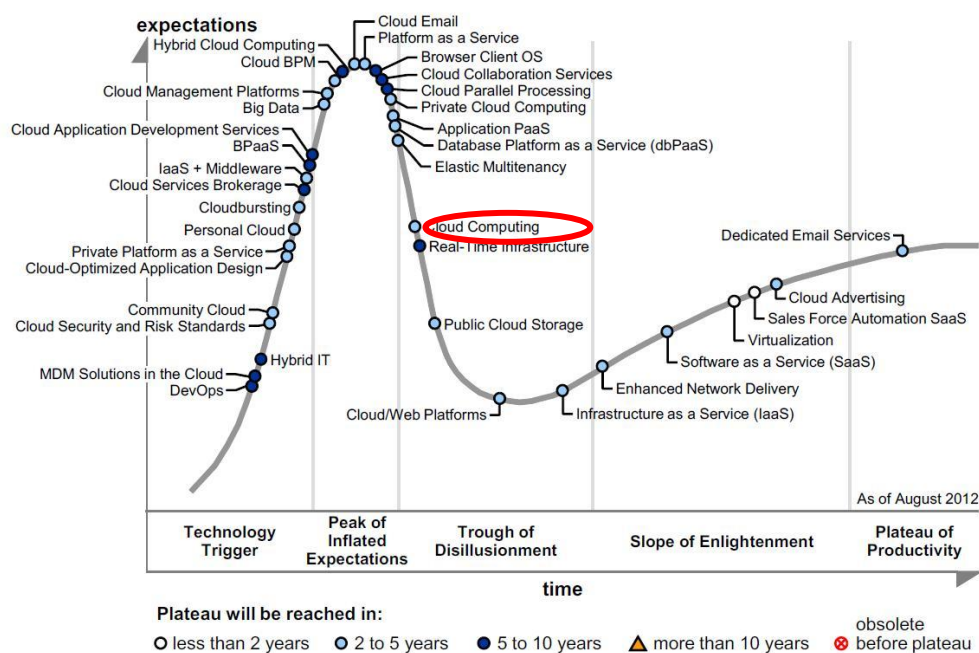
Η επόμενη Εικόνα 4.Εξέλιξη του νέφους 2012, παρουσιάζει την εξέλιξη του νέφους τον Αύγουστο 2012. Πλέον, το νέφος ξεπερνά οριστικά το δεύτερο στάδιο και οδηγείται στο επόμενο. Οι ενδιαφερόμενοι, πελάτες – χρήστες ή επιχειρήσεις αρχίζουν πλέον να αλλάζουν αγοραστική συμπεριφορά με βάση την ανάπτυξη της οικονομίας και την προσαρμογή αυτών στη νέα τεχνολογία. Παρά την ύπαρξη κερδοφόρας νέας τεχνολογίας, οι επιχειρήσεις είναι



Κωνσταντίνος Αθανασίου , 'Ανοχή σε σφάλματα σε υπολογιστικό περιβάλλον νέφους'

δύσκολο να εγκαταλείψουν τα δοκιμασμένα μοντέλα τεχνολογικών υποδομών και να μετατραπούν σε κάτι καινούργιο, δηλαδή να κάνουν χρήση του νέφους. Με βάση την ανάλυση του κύκλου Gartner για το Cloud Computing, το 2012, τα καλύτερα αποτελέσματα επιτυγχάνονται με τις επιχειρήσεις που εστιάζουν σε μια πολύ συγκεκριμένη στρατηγική και να αποδέχονται μερικώς τις τεχνολογίες νέφους, ώστε να επιτευχθεί η απόδοσή τους. Αποτέλεσμα της εφαρμογής του νέφους οδηγεί τις επιχειρήσεις στη παρούσα κατάσταση του νέφους σε μείωση κόστους, αύξηση ταχύτητας συναλλαγών και αποτελεσματική διαχείριση υπηρεσιών και εφαρμογών

Figure 1. Hype Cycle for Cloud Computing, 2012



Source: Gartner (August 2012)

#### Εικόνα 4.Εξέλιξη του νέφους 2012

Η τελευταία ανάλυση του νέφους παρουσιάζεται στην Εικόνα 5. Εξέλιξη του νέφους 2014. Εμφανίζει το νέφος στο ίδιο στάδιο εξέλιξής του, πιο αναπτυγμένο και ώριμο να περάσει στο επόμενο στάδιο. Οι περισσότεροι των χρηστών και υπηρεσιών εκτελούν υπηρεσίες και εφαρμογές στο νέφος, χωρίς ιδιαίτερα προβλήματα. Οι ενδιαφερόμενοι έχουν πλέον ενσωματώσει τις υπηρεσίες τους στο νέφος, και έχουν αποκτήσει άμεση εξάρτηση με αυτό. Πολλές από τις εταιρίες έχουν προσπαθήσει και έχουν καταφέρει τη δημιουργία νέφους με σκοπό να προωθήσουν τις δικές τους υπηρεσίες και προγράμματα. Η εφαρμογή του νέφους είναι σχεδόν καθολική, οι χρήστες κάνουν χρήση του νέφους και η τεχνολογία πλέον είναι αποδεκτή.



Κωνσταντίνος Αθανασίου , 'Ανοχή σε σφάλματα σε υπολογιστικό περιβάλλον νέφους'



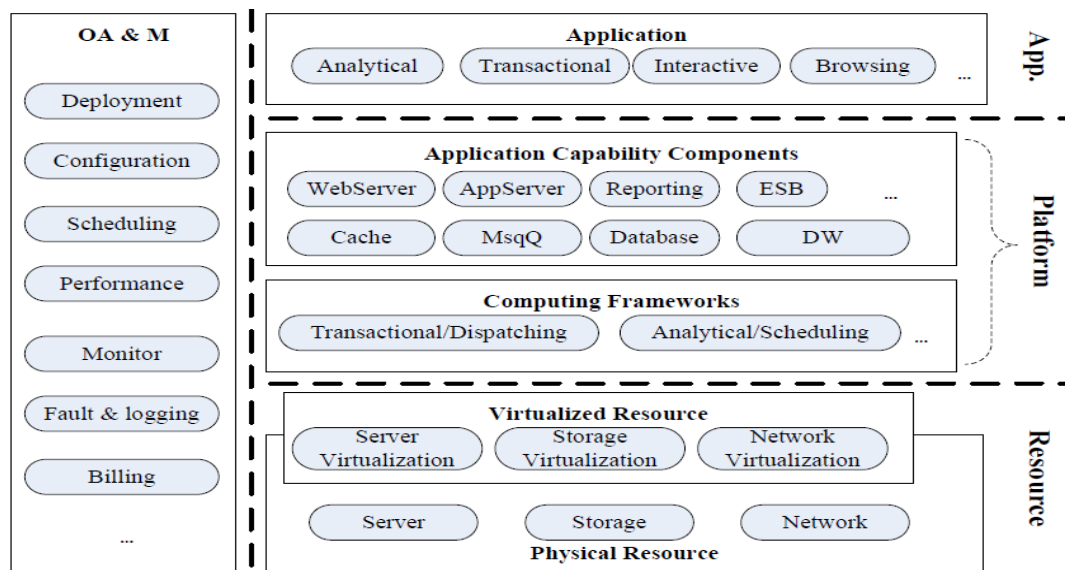
Εικόνα 5. Εξέλιξη του νέφους 2014

Οι παρακάτω εικόνες μας παρουσιάζουν την εξελεγκτική πορεία του νέφους από το 2008 έως και το 2014. Η πορεία της καινοτόμας ιδέας είναι ακόμη σε ανάπτυξη καθώς δεν βρίσκεται στο τελευταίο στάδιο, σύμφωνα με το διάγραμμα., αλλά παραμένει στο τρίτο στάδιο, εξελίσσεται αλλά παρουσιάζει στο ευρύ κοινό τα πλεονεκτήματα χρήσης των υπηρεσιών που προσφέρει. Παρά της ανάπτυξης του νέφους δεν είναι σαφές ο χρονικός προσανατολισμός της ανάπτυξης αυτής. Για να μπορέσει να αποτελέσει απαραίτητο εργαλείο ανάπτυξης νέων προϊόντων θα πρέπει οι ερευνητές του νέφους να εξηγήσουν πως μπορεί ο χρήστης να αξιολογήσει ένα νέφος, ποιος είναι ο κατάλληλος πάροχος, πως αλληλεπιδρά το ιδιωτικό με το δημόσιο νέφος ή με διαφορετικού είδους νέφη και άλλες τέτοιου είδους ερωτήσεις, ώστε το νέφος να μπορεί να χρησιμοποιηθεί πλήρως και χωρίς ασάφειες και ερωτήματα για την ορθή και αποτελεσματική χρήση του.

## 2.2. Αρχιτεκτονική του «Cloud» και οι Προσφερόμενες Υπηρεσίες

Σε πολλά ερευνητικά κέντρα και σε οργανώσεις έχει υπάρξει προσπάθεια απόδοσης της αρχιτεκτονικής του «Cloud». Το αποτέλεσμα των ερευνών αποτυπώνεται στην παρακάτω Εικόνα 6. Αρχιτεκτονική Cloud. [11].





Εικόνα 6. Αρχιτεκτονική Cloud

Διαχωρίζεται ο πυρήνας στοίβας από τη διαχείριση. Στον πυρήνα στοίβας υπάρχουν τρία στρώματα, τα οποία είναι οι πόροι (1), η πλατφόρμα (2) και οι εφαρμογές (3). Στο στρώμα των πόρων συγκαταλέγεται το στρώμα των υποδομών, όπου σε αυτό το σημείο συγκαταλέγονται η φυσική και η εικονική υπολογιστική ισχύ, ο χώρος αποθήκευσης των δεδομένων καθώς επίσης και οι πόροι του δικτύου. Εξαιτίας της ύπαρξης και χρήσης του διαδικτύου εκτελούνται μεγάλες και κατακευματισμένες συναλλαγές δεδομένων καθώς επίσης και επικοινωνία με άλλες υπηρεσίες που λειτουργούν στο διαδίκτυο.

Οι υπηρεσίες που προσφέρονται με τη χρήση του «Cloud Computing» διακρίνονται ανάλογα το σκοπό χρήσης τους. Διακρίνονται, με βάση τη βιβλιογραφία, δύο είδη υπηρεσιών. [11] Το πρώτο είδος αναφέρεται στο όριο υπηρεσίας «Service Boundary» και το δεύτερο στο τύπο της υπηρεσίας που παρέχεται «Service Type».

Οι υπολογιστικοί πόροι του παρόχου χρησιμοποιούνται για να εξυπηρετήσουν πολλαπλούς καταναλωτές με τη χρήση του μοντέλου πολλαπλών μισθωτών («multi-tenant»), με τους διάφορους φυσικούς και εικονικούς πόρους να ανατίθενται δυναμικά και εκ νέου ανάλογα με τη ζήτηση των καταναλωτών. Υπάρχει μια αίσθηση ανεξαρτησίας από τον τόπο στο γεγονός ότι ο πελάτης δεν έχει γενικά κανέναν έλεγχο ή γνώση σχετικά με την ακριβή τοποθεσία των παρεχόμενων πόρων, αλλά μπορεί να είναι σε θέση να προσδιορίζει την τοποθεσία σε ένα υψηλότερο επίπεδο αφαίρεσης (πχ. χώρα, κράτος, ή «datacenter»). Παραδείγματα πόρων αποτελούν οι αποθηκευτικοί χώροι, η επεξεργασία, η μνήμη, το «bandwidth» του δικτύου, καθώς και οι εικονικές μηχανές. Ένας καταναλωτής μπορεί να δεσμεύσει από μόνος του τους υπολογιστικούς πόρους που χρειάζεται, όπως χρόνο στον «server» και αποθηκευτικό χώρο στο δίκτυο, ανάλογα με τις ανάγκες του αυτόματα, χωρίς να απαιτείται ανθρώπινη αλληλεπίδραση με το φορέα παροχής κάθε υπηρεσίας. Οι δυνατότητες είναι διαθέσιμες μέσω του δικτύου και προσβάσιμες μέσω τυποποιημένων μηχανισμών που προωθούν την χρήση από ετερογενείς «thin» ή «thick client» πλατφόρμες (π.χ. κινητά τηλέφωνα, φορητούς υπολογιστές και «PDAs»).

Το άλλο και σημαντικό που κάνει την τεράστια διαφορά, είναι πως το «Cloud» διαθέτει «virtualization», τ' οποίο τα τελευταία χρόνια κάλπασε σε εξέλιξη. Με την χρήση





*Κωνσταντίνος Αθανασίου , 'Ανοχή σε σφάλματα σε υπολογιστικό περιβάλλον νέφους'*

επιτυγχάνεται μια ισότητα και εξοικονόμηση πόρων, καθώς ο καθένας χρησιμοποιεί μόνο αυτό που θέλει (και όποτε το θέλει). ενώ το υπόλοιπο, κάποιοι άλλοι και αντίστοιχα.

Οι πόροι μπορούν να δεσμευτούν προς χρήση γρήγορα και ελαστικά, σε ορισμένες περιπτώσεις αυτόματα, έτσι ώστε να εμφανιστούν άμεσα ως μη διαθέσιμοι («scale out») και επίσης να αποδεσμευτούν γρήγορα για να εμφανιστούν ξανά ως διαθέσιμοι («scale in»). Για τον καταναλωτή, οι διαθέσιμες δυνατότητες για δέσμευση και χρήση συχνά φαίνεται να είναι απεριόριστες και μπορούν να αγοραστούν ανά πάσα στιγμή και σε οποιαδήποτε ποσότητα.

Το «Cloud Computing» είναι ένα μοντέλο που επιτρέπει ευέλικτη, «on-demand» δικτυακή πρόσβαση σε ένα κοινόχρηστο σύνολο παραμετροποιήσεων υπολογιστικών πόρων (όπως δίκτυα, «servers», αποθηκευτικοί χώροι, εφαρμογές και υπηρεσίες), το οποίο μπορεί να τροφοδοτηθεί γρήγορα και να διατεθεί με ελάχιστη προσπάθεια διαχείρισης ή αλληλεπίδραση με τον πάροχο της υπηρεσίας. Αυτό το «cloud» μοντέλο προωθεί την διαθεσιμότητα και αποτελείται από πέντε βασικά χαρακτηριστικά, τρία μοντέλα παροχής υπηρεσιών, και τέσσερα μοντέλα ανάπτυξης.

Το «Cloud Computing» από την πλευρά του ορίου υπηρεσίας, μπορεί να χαρακτηριστεί ως δημόσιο «Public Cloud», ιδιωτικό «Private Cloud», κοινότητας «Community Cloud» και υβριδικό «Hybrid Cloud».

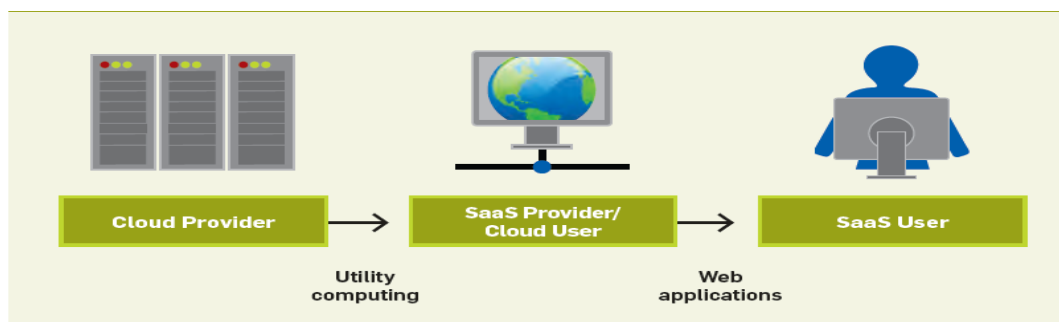
**Private cloud:** Η «Cloud» υποδομή λειτουργεί αποκλειστικά και μόνο για έναν. Η διαχείρισή της μπορεί να γίνεται από τον ίδιο τον οργανισμό ή από τρίτους και μπορεί να βρίσκεται εντός ή εκτός των εγκαταστάσεων του οργανισμού.

**Community cloud:** Η «Cloud» υποδομή μοιράζεται μεταξύ πολλών οργανισμών και υποστηρίζει μια συγκεκριμένη κοινότητα που έχει κοινές ανησυχίες (π.χ. αποστολή, απαιτήσεις ασφαλείας, πολιτική και θέματα συμμόρφωσης). Η διαχείρισή της μπορεί να γίνεται από τον ίδιο τον οργανισμό ή από τρίτους και μπορεί να βρίσκεται εντός ή εκτός των εγκαταστάσεων του οργανισμού.

**Public cloud:** Η «Cloud» υποδομή διατίθεται στο ευρύ κοινό ή σε μια μεγάλη ομάδα εταιρειών και ανήκει σε έναν οργανισμό που πουλά υπηρεσίες «cloud».

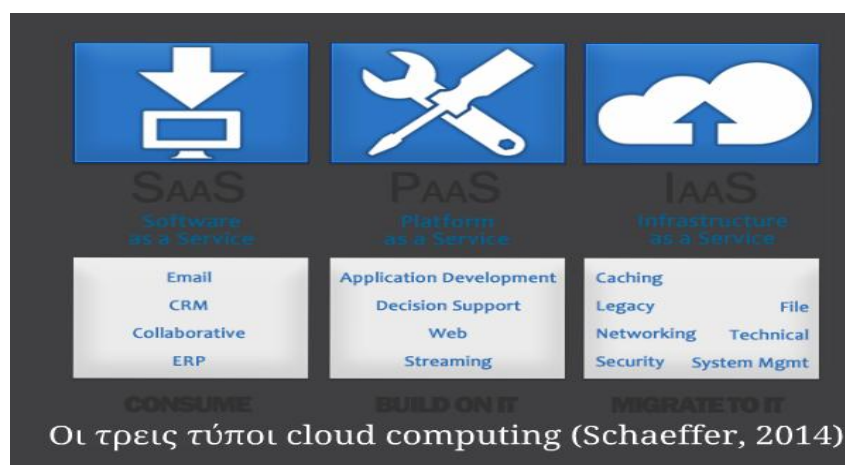
**Hybrid cloud:** Η «Cloud» υποδομή είναι μια σύνθεση από δύο ή περισσότερα «clouds» («private», «community», «public») τα οποία παραμένουν μοναδικές οντότητες, αλλά συνδέονται μεταξύ τους με τυποποιημένη ή αποκλειστική τεχνολογία που επιτρέπει τη φορητότητα δεδομένων και εφαρμογών (π.χ. εξισορρόπηση φόρτου εργασίας μεταξύ των «clouds»).

Το δημόσιο αναφέρεται σε υπηρεσίες που παρέχονται σε εξωτερικούς φορείς, ενώ το ιδιωτικό κατασκευάζεται από εταιρίες οι οποίες θέλουν να καλύψουν τις ανάγκες τους χτίζοντας το δικό τους περιβάλλον νέφους, πάνω στο οποίο εκτελούνται και οι προσφερόμενες από αυτές υπηρεσίες. Το υβριδικό νέφος αποτελεί ένα συνδυασμό των παραπάνω, μοιράζοντας πόρους μεταξύ του δημόσιου και του ιδιωτικού νέφους μέσω ενός ασφαλούς δικτύου.



Εικόνα 7. Σχέση Χρήστη-Παρόχου

Ο δεύτερος τύπος υπηρεσίας του νέφους, από την πλευρά του τύπου δηλαδή, μπορεί να χαρακτηριστεί ως υπηρεσία υποδομών «Infrastructure as a Service (IaaS)», υπηρεσία ως πλατφόρμα «Platform as a Service (PaaS)» και υπηρεσία ως λογισμικό «Software as a Service (SaaS)», όπως παρουσιάζεται και στην Εικόνα 8. SaaS, PaaS, IaaS Στους τελικούς χρήστες παρέχονται υπηρεσίες ως λογισμικό, ενώ υποδομών και πλατφόρμας παρέχονται σε προγραμματιστές ως πακέτα εφαρμογών.



Εικόνα 8. SaaS, PaaS, IaaS

Ο χρήστης σε μια υπηρεσία «IaaS» μπορεί αυτόνομα να σχεδιάσει και να διαχειριστεί τη δική του υπολογιστική υποδομή. Μπορεί για παράδειγμα να δημιουργήσει εικονικά δίκτυα χρηστών, να προσθέσει εικονικό εξοπλισμό («servers», βάσεις δεδομένων) και να εγκαταστήσει πολλαπλά λειτουργικά συστήματα και εφαρμογές της επιλογής του. Σε μια υπηρεσία «PaaS» ο πάροχος υπηρεσιών «Cloud» παραχωρεί στο χρήστη πρόσβαση σε λειτουργικά συστήματα διαχείρισης βάσεων δεδομένων, εργαλεία ανάπτυξης λογισμικού. Όλη η πληροφοριακή υποδομή (δίκτυα, «servers», συστήματα αποθήκευσης) ελέγχεται από τον πάροχο. Η σχέση παρόχου και χρήστη φαίνεται και στην Εικόνα 7. Σχέση Χρήστη-Παρόχου και Εικόνα 9. Σχέση μεταξύ χρηστών και παρόχων. Στο επίπεδο «SaaS» ο πάροχος παρέχει στο χρήστη λογισμικό και υπηρεσίες. Όλα τα δεδομένα αποθηκεύονται στο νέφος και προκειμένου ο χρήστης να τα προσπελάσει, χρειάζεται μόνο ένα φυλλομετρητή («web browser»). Θεωρείται η πιο άμεσα εκμεταλλεύσιμη υπηρεσία για τα εκπαιδευτικά ιδρύματα, καθώς δεν απαιτεί έξοδα και πόρους για την αγορά, εγκατάσταση και συντήρηση εξοπλισμού



και λογισμικού το οποίο απαιτείται όταν κάποιος επιλέξει να χρησιμοποιήσει τις υπηρεσίες «IaaS» και «PaaS».

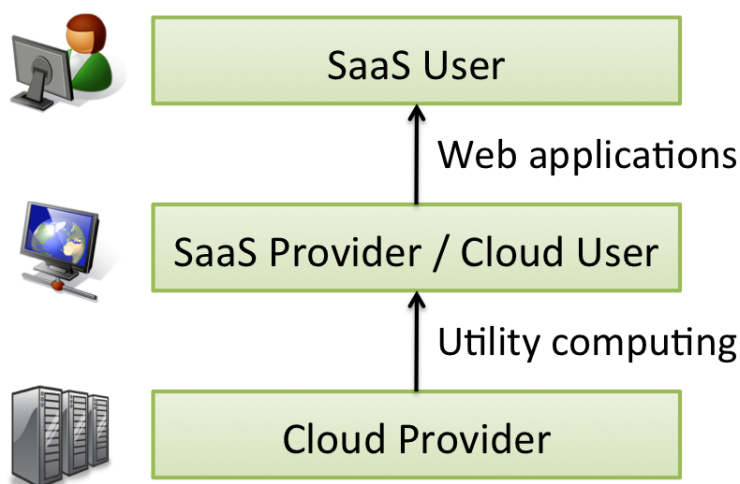
**Cloud Software as a Service (SaaS):** Η δυνατότητα που παρέχεται στον καταναλωτή είναι να χρησιμοποιεί τις εφαρμογές του παρόχου που τρέχουν σε μια «cloud» υποδομή. Οι εφαρμογές είναι προσβάσιμες από διάφορες client συσκευές μέσω ενός thin client interface, όπως ένα πρόγραμμα περιήγησης στο «Web» (π.χ. «web-based email»). Ένα καλό παράδειγμα, ώστε να γίνει αντιληπτό αυτό είναι το «GoogleDrive» και οι εφαρμογές που μπορούν να τρέξουν απ' ευθείας σε αυτό. Ο καταναλωτής δεν έχει τη διαχείριση ή τον έλεγχο της χρησιμοποιούμενης «cloud» υποδομής συμπεριλαμβανομένων των δικτύων, των «servers», των λειτουργικών συστημάτων, των αποθηκευτικών μονάδων, ή ακόμα και μεμονωμένων δυνατοτήτων της εφαρμογής, με την πιθανή εξαίρεση κάποιων περιορισμένων «user-specific» ρυθμίσεων παραμετροποίησης των εφαρμογών.

**Cloud Platform as a Service (PaaS):** Η δυνατότητα που παρέχεται στον καταναλωτή είναι να αναπτύσσει πάνω στην «cloud» υποδομή εφαρμογές που έχει δημιουργήσει ή εφαρμογές που έχει αποκτήσει, οι οποίες έχουν δημιουργηθεί με χρήση γλωσσών προγραμματισμού και εργαλείων που υποστηρίζονται από τον πάροχο. Ο καταναλωτής δεν διαχειρίζεται ούτε ελέγχει τη σχετική «cloud» υποδομή που συμπεριλαμβάνει τα δίκτυα, τους servers, τα λειτουργικά συστήματα ή τα αποθηκευτικά μέσα, αλλά έχει τον έλεγχο των εφαρμογών που έχουν αναπτυχθεί, και ενδεχομένως, των παραμετροποιήσεων του περιβάλλοντος φιλοξενίας των εφαρμογών.

**Cloud Infrastructure as a Service (IaaS):** Η δυνατότητα που παρέχεται στον καταναλωτή είναι να μπορεί να δεσμεύσει προς χρήση επεξεργαστική ισχύ, αποθηκευτικά μέσα, δίκτυα, και άλλους θεμελιώδεις υπολογιστικούς πόρους, όπου ο καταναλωτής είναι σε θέση να αναπτύξει και να εκτελέσει αυθαίρετο λογισμικό, το οποίο μπορεί να περιλαμβάνει λειτουργικά συστήματα και εφαρμογές. Ο καταναλωτής δεν έχει τη διαχείριση ή τον έλεγχο της χρησιμοποιούμενης cloud υποδομής, αλλά έχει τον έλεγχο των λειτουργικών συστημάτων, των αποθηκευτικών μέσων, των εφαρμογών που έχουν αναπτυχθεί και πιθανόν κάποιον περιορισμένο έλεγχο επιλεγμένου εξοπλισμού δικτύωσης (π.χ. «firewalls»).

Παραδείγματα παρόχων που προσφέρουν υπηρεσίες «Cloud Computing» είναι οι AmazonWeb Services, Google, AppEngine, Microsoft Azure.

Οι προσφερόμενες υπηρεσίες από την εταιρία «AmazonWeb» για παράδειγμα, είναι μια συλλογή απομακρυσμένων υπηρεσιών στη πλατφόρμα Amazon.com. Η εταιρία διαφημίζεται παρέχοντας μεγάλη υπολογιστική ικανότητα, με πολύ γρήγορη και φθηνή ανάπτυξη των μηχανημάτων ή σύμπλεγμα αυτών. Παρόμοιες υπηρεσίες προσφέρουν και οι υπόλοιπες εταιρίες.



Εικόνα 9. Σχέση μεταξύ χρηστών και παρόχων

## 2.3. Πλεονεκτήματα-Κίνδυνοι

### 2.3.1. Πλεονεκτήματα και Ανησυχίες

Το «Cloud Computing» αποτελεί μια «win-win» στρατηγική για τον πάροχο και τις υπηρεσίες που παρέχει. Τόσο οι πελάτες-καταναλωτές όσο και οι εταιρίες-πάροχοι των υπηρεσιών νέφους, έχουν τη δυνατότητα να ωφεληθούν από τη ραγδαία εξέλιξη της τεχνολογίας και της προσπάθειας αυτών να μεταβούν ή μετεγκατασταθούν στο νέφος. Στον παρακάτω λίστα αναγράφονται τα πλεονεκτήματα όσων χρησιμοποιούν ή πρόκειται να χρησιμοποιήσουν το περιβάλλον νέφους. [12]

- Ικανοποίηση απαιτήσεων των επιχειρήσεων για τη ζήτηση στην αγορά εργασίας, μεταβάλλοντας τους διαθέσιμους πόρους όπου αυτό είναι αναγκαίο, ανάλογα με τις εφαρμογές και υπηρεσίες που εκτελούνται.

- Χαμηλότερο κόστος και κατανάλωση ενέργειας σε σχέση με τη κανονική εγκατάσταση των υλικών και πόρων. Πιο αναλυτικά, όταν οι υπολογιστές μιας εταιρίας εκτελούν τις προσφερόμενες υπηρεσίες συνεχόμενα σε περιβάλλον νέφους, σπαταλούν λιγότερη ενέργεια, καθώς οι υπολογιστές αυτοί είναι εικονικά μηχανήματα και όχι πραγματικά. Επιπλέον, το χαμηλότερο κόστος μπορεί να γίνει αντιληπτό από το γεγονός ότι στο νέφος οι εταιρίες/καταναλωτές πληρώνουν με βάση την απαιτούμενη χρήση τους. Η απαίτηση αυτή μπορεί εύκολα να μεταβληθεί αυξάνοντας ή μειώνοντας τους πόρους του συστήματος. Αν η εταιρία είχε πραγματική εγκατάσταση, τότε τα μηχανήματά της θα έπρεπε να είναι ενεργά συνεχώς καταναλώνοντας ρεύμα, χώρο, χρήματα συντήρησης, επιβλέπων προσωπικό. Το «Cloud Computing» είναι ίσως η πιο αποτελεσματική, από πλευράς κόστους, μέθοδος για τη χρήση, διατήρηση και αναβάθμιση των υπολογιστών και υπηρεσιών. Τα παραδοσιακά «Desktop» διαθέτουν λογισμικό, το οποίο κοστίζει στις εταιρείες πολλά, από πλευράς χρηματοδότησης. Προσθέτοντας στο παραπάνω κόστος και τα τέλη αδειοδότησης για πολλούς χρήστες μπορεί να αποδειχθεί πολύ ακριβό για την εν λόγω εγκατάσταση. Το νέφος, από την άλλη πλευρά, είναι διαθέσιμο σε πολύ χαμηλότερες τιμές και, ως εκ τούτου,



μπορεί να μειώσει σημαντικά τις δαπάνες πληροφορικής της εταιρείας. Εκτός αυτού, υπάρχουν πολλοί «one-time-πληρωμές», «pay-as-you-go» και άλλες διαθέσιμες επιλογές επέκτασης, γεγονός που καθιστά πολύ λογικές για την εν λόγω εταιρεία.

- Βελτίωση της αποτελεσματικότητας της διαχείρισης πόρων μέσω δυναμικών αλλαγών στη παροχή πόρων.

- Σχεδόν Απεριόριστη αποθήκευση. Η αποθήκευση πληροφοριών στο νέφος, δίνει σχεδόν απεριόριστη ικανότητα αποθήκευσης. Ως εκ τούτου, δεν χρειάζεται πλέον να υπάρχει η ανησυχία για τη διακοπή του χώρου αποθήκευσης ή την αύξηση της τρέχουσας διαθεσιμότητας στον αποθηκευτικό χώρο των χρηστών.

- Αντίγραφα Ασφαλείας και Ανάκτηση Πληροφοριών. Δεδομένου ότι όλα τα δεδομένα είναι αποθηκευμένα στο νέφος, η δημιουργία αντιγράφων ασφαλείας και η επαναφορά του ιδίου είναι σχετικά πολύ πιο εύκολη από την αποθήκευση το ίδιο σε μια φυσική συσκευή. Επιπλέον, οι περισσότεροι πάροχοι υπηρεσιών «Cloud» είναι συνήθως αρκετά ικανοί να χειριστούν την ανάκτηση των πληροφοριών. Ως εκ τούτου, αυτό καθιστά την όλη διαδικασία της δημιουργίας αντιγράφων ασφαλείας και ανάκτησης πολύ πιο απλή από ό, τι άλλες παραδοσιακές μεθόδους αποθήκευσης δεδομένων

- Αυτόματη ολοκλήρωση λογισμικού («Automatic Software Integration»). Στο νέφος, η ενσωμάτωση του λογισμικού συνήθως είναι κάτι που συμβαίνει αυτόματα. Αυτό σημαίνει ότι δεν χρειάζεται να λάβει ο χρήστης πρόσθετες προσπάθειες για να προσαρμόσει και να ενσωματώσει τις εφαρμογές σύμφωνα με τις προτιμήσεις αυτού. Με το πλεονέκτημα αυτό, ο χρήστης έχει τη δυνατότητα να προσαρμόσει το λογιστικό στις δικές του ανάγκες.

- Εύκολη πρόσβαση σε πληροφορίες.

- Γρήγορη Ανάπτυξη.

Η μετάβαση στο «Cloud Computing» μπορεί να αποτελεί σημαντικό ωφέλιμο γεγονός τόσο για μια εταιρία όσο και έναν απλό χρήστη του διαδικτύου, όμως ακόμη και σήμερα οι ερευνητές σε ιδρύματα και επιστημονικά κέντρα, προσπαθούν να δώσουν ολοκληρωμένες λύσεις σε μια σειρά από προβλήματα. Τα πιο σημαντικά από αυτά περιλαμβάνονται στη παρακάτω λίστα:

- Απόρρητο δεδομένων - Ασφάλεια. Οι χρήστες του νέφους και οι πελάτες της εταιρίας, αλλά και η ίδια η εταιρία έχει την ανησυχία για την προστασία των δεδομένων τους και της ιδιωτικής ζωής.

- Φόβος συνέχισης της υπηρεσίας. Επειδή το «Cloud Computing» είναι λειτουργία που απαιτεί χρήση διαδικτύου, υπάρχουν στους ενδιαφερόμενους ο φόβος διακοπής και η ύπαρξη σφαλμάτων. Μια ενδεχόμενη διακοπή ρεύματος, για παράδειγμα, θα ήταν καταστροφή για την εταιρία, καθώς σε αυτό το διάστημα θα σταματήσουν και οι λειτουργίες της. Τις ανησυχίες αυτές, μόνο ο πάροχος μπορεί να τις απαντήσει και να δώσει άμεσες και αποτελεσματικές οδηγίες.

- Τεχνικά Θέματα. Αν και είναι αλήθεια ότι μπορούν να ληφθούν οι πληροφορίες και τα δεδομένα σχετικά εύκολα οποτεδήποτε και από οπουδήποτε, υπάρχουν φορές που το σύστημα μπορεί να έχει κάποια σοβαρή δυσλειτουργία. Θα πρέπει να υπάρχει η επίγνωση του γεγονότος ότι αυτή η τεχνολογία είναι πάντα επιρρεπείς σε διακοπές και άλλα τεχνικά θέματα. Ακόμα και οι καλύτεροι των παρόχων υπηρεσιών «Cloud» ερευνούν τέτοιου είδους



προβλήματα. Εκτός αυτού, απαιτείται τις περισσότερες φορές μια πολύ καλή σύνδεση με το Διαδίκτυο ώστε να υπάρχει σύνδεση στο διακομιστή, ανά πάσα στιγμή.

- Επιρρεπείς στην επίθεση. Η αποθήκευση πληροφοριών στο νέφος είτε αφορά το χρήστη είτε μια εταιρεία, είναι ευάλωτη σε εξωτερικές επιθέσεις «Hack και απειλές. Θα πρέπει να αναφέρουμε ότι τίποτα στο διαδίκτυο είναι απολύτως ασφαλές και, ως εκ τούτου, υπάρχει πάντα η πιθανότητα επιθέσεων και απόκτησης μυστικών και ευαίσθητων δεδομένων.

### 2.3.2. Κίνδυνοι

Οι κίνδυνοι που αντιμετωπίζουν όσοι χρησιμοποιούν το «Cloud Computing» εξακολουθούν να είναι περίπου όμοιοι με τα ήδη υπάρχουσα συστήματα. Κίνδυνοι όπως ιδιωτικότητα των δεδομένων παραμένουν ενεργοί και γίνεται αρκετά μεγάλη προσπάθεια καταπολέμησης του κινδύνου αυτού. Όμως στο περιβάλλον νέφος, ο κίνδυνος μπορεί να χαρακτηριστεί διαφορετικός με τη έννοια ότι τα δεδομένα δε βρίσκονται σε ένα μόνο μηχάνημα, αλλά σε διαφορετικούς κεντρικούς χώρους δεδομένων («Data Center»), χωρίς ο χρήστης να γνωρίζει πως αυτά μοιράζονται και κατανέμονται στο μηχάνημά του. Ο πάροχος του νέφους είναι αυτός που εξασφαλίζει με τεχνικές και νόμιμες διαδικασίες την ιδιωτικότητα των δεδομένων. Επίσης, η διαθεσιμότητα αποτελεί εξίσου σημαντικός κίνδυνος του περιβάλλοντος νέφους και συνδέεται έμμεσα με την ιδιωτικότητα του νέφους. Ο χρήστης, είτε πρόκειται για απλό χρήστη είτε για εταιρία απαιτεί τα δεδομένα ή οι υπηρεσίες του να είναι διαθέσιμα όποτε αυτός χρειαστεί. Ακόμη υπάρχουν κίνδυνοι που προέρχονται από κακόβουλο λογισμικό («Malicious Attacks») καθώς και από τη μη τήρηση κανόνων συμμόρφωσης μεταξύ των παρόχων και των χρηστών νέφους. Είναι σε μεγάλο βαθμό σημαντική η καταπολέμηση των κινδύνων, ώστε να υπάρχει εμπιστοσύνη μεταξύ των συνεργατών του νέφους. Ο καθορισμός των απαιτήσεων ασφαλείας των συνεργατών πραγματοποιείται προτού η εταιρία ή ο χρήστης εγκατασταθεί σε περιβάλλον νέφους. [13]

### 2.3.3. Μετεγκατάσταση Εταιριών στο Νέφος

Το περιβάλλον νέφους, «Cloud Computing» αποτελεί μια εναλλακτική προσέγγιση για τις επιχειρήσεις, οι οποίες θέλουν να μειώσουν το κόστος, τόσο της ανάπτυξης όσο και της συντήρησής τους, διατηρώντας ωστόσο τους πελάτες τους ευχαριστημένους με τις υπηρεσίες που προσφέρουν, καθώς επίσης και να διατηρήσουν ή να αυξήσουν τα κέρδη τους. Η προσπάθεια των συγγραφέων έγκειται στην απάντηση που πρέπει να δώσουν σχετικά με τη μεταβολή ή μετεγκατάσταση μιας επιχείρησης σε περιβάλλον νέφους. Η μεταβολή αυτή, αν και φαίνεται απλή απαιτεί σχέδιο μετάβασης, κατανόηση των κινδύνων και των απειλών, κατανόηση των διαφόρων τρωτών σημείων του νέφους και ενδεχόμενα αντίμετρα που θα πρέπει να ληφθούν. Οι πάροχοι του νέφους είναι εταιρίες, όπως Amazon, Google, Microsoft, οι οποίες βασικό στόχο έχουν να παρέχουν τις απαιτούμενες υπηρεσίες στον πελάτη (επιχείρηση). Τέτοιες υπηρεσίες αποτελούν για παράδειγμα, ο χώρος αποθήκευσης, μνήμη, επεξεργαστική ισχύ σε εικονικά μηχανήματα. Παρά το γεγονός ότι η μείωση του κόστους ζωής μιας επιχείρησης είναι ελκυστική πρόταση, οι επιχειρήσεις θα πρέπει να σκεφτούν πιο "καθαρά" και να αναλύσουν το κίνδυνο της ασφάλειας σε σχέση με τη μεταβολή και εγκατάσταση της επιχείρησης στο περιβάλλον νέφους. Μία ακόμη ανησυχία που θα πρέπει να προβληματίζει τους επικεφαλής των επιχειρήσεων αποτελεί το γεγονός ότι οι απειλές των "hackers" στρέφονται στο νέφος, διότι γνωρίζουν ότι αποτελεί "το καινούργιο", οπότε η ασφάλεια θα πρέπει να αυξηθεί σε μέγιστο βαθμό, ώστε να αποφευχθεί η διαρροή





προσωπικών δεδομένων των πελατών της επιχείρησης ή οι συναλλαγές της. Σύμφωνα με τους συγγραφείς, [12] διακρίνουμε επτά (7) κύριες απειλές. Αυτές είναι κατάχρηση του νέφους, μη σωστά αναπτυγμένες εφαρμογές, κακόβουλο λογισμικό, ευπάθειες σε τεχνολογίες διαμοιρασμού αρχείων, διαρροή δεδομένων, «hijacking» λογαριασμών, άγνωστη απειλή. Στους παρακάτω πίνακες παραθέτουμε οδηγίες, οι οποίες θα βοηθήσουν τις επιχειρήσεις να μετασταθμεύσουν το υλικό τους σε περιβάλλον νέφους όσον αφορά την ασφάλεια, σε θέματα προτού αποφασιστεί η μετάβαση στο νέφος, και τέλος, τα πλεονεκτήματα της μεταβολής αυτής.

**Πίνακας 2. Οδηγίες Ασφαλείας Χρήσης Νέφους**

Οδηγίες Ασφάλειας	Μερική Ανάλυση
<b>Κατανόηση Νέφους</b>	Κατανόηση λειτουργίας του νέφους, πως μεταφέρονται και διαχειρίζονται τα δεδομένα.
<b>Διαφάνεια</b>	Αναλυτική εξήγηση του παρόχου όσον αφορά τις υπηρεσίες που παρέχει, την ασφάλεια και την αρχιτεκτονική της ασφάλειας.
<b>Ενίσχυση Εσωτερικής Ασφάλειας</b>	Εξασφαλίζοντας το γεγονός ότι ο πάροχος διαθέτει ισχυρούς μηχανισμούς ασφάλειας (ελέγχου χρηστών, τειχών προστασίας)
<b>Νομικές Συνέπειες</b>	Διασφαλίζοντας τη γνώση των νόμων και περιορισμών των δεδομένων που "ανεβάζεις" στο νέφος.
<b>Ιδιαίτερη Προσοχή</b>	Παρακολούθηση της ανάπτυξης της τεχνολογίας και των αλλαγών που μπορεί να έχει στη λειτουργία του νέφους επηρεάζοντας αρνητικά τις υπηρεσίες και τα δεδομένα

**Πίνακας 3. Απόφασης Μεταβολής στο Νέφος**

Απόφαση Μεταβολής	Μερική Ανάλυση
<b>Πρόσβαση Χρηστών</b>	Ζητήστε από τον πάροχο συγκεκριμένες πληροφορίες σχετικά με τη πρόσληψη και την εποπτεία του προσωπικού στη διαχείριση της πληροφορίας. Οι εταιρίες είναι υποχρεωμένες να επιβάλουν κριτήρια στο προσωπικό που εργάζεται σε περιβάλλον νέφους.
<b>Συμμόρφωση</b>	Βεβαιωθείτε ότι ο πάροχος είναι έτοιμος να υποστεί εξωτερικούς ελέγχους και πιστοποιήσεις ασφαλείας.
<b>Τοποθεσία Δεδομένων</b>	Οι επιχειρήσεις θα πρέπει να απαιτήσουν από τον πάροχο ώστε τα δεδομένα τους να είναι αποθηκευμένα σε χώρες που σέβονται την ιδιωτικότητα.
<b>Διαχωρισμός Δεδομένων</b>	Ζητήστε να μάθετε τί γίνεται με το διαχωρισμό δεδομένων και ζητήστε απόδειξη σωστής κρυπτογράφησης.
<b>Αδυναμία Ανάκτησης</b>	Αποκτήστε γνώμη αντιμετώπισης ανάκτησης δεδομένων, καθώς και πόσο χρόνο θα χρειαστεί ο πάροχος να ανακτήσει τα δεδομένα.
<b>Βιωσιμότητα</b>	Ρωτήστε το πάροχο πως θα πάρετε πίσω τα δεδομένα σας σε περίπτωση προβλήματος και σε ποιά μορφή.



#### Πίνακας 4. Πλεονεκτήματα Νέφους

Πλεονεκτήματα Νέφους	Μερική Ανάλυση
<b>Τοποθεσίες Δεδομένων, Δίκτυα σε Μορφή Ακμών, Χρονική Περίοδος</b>	Η ασφάλεια των συστημάτων είναι φθηνότερη σε σχέση με τα "σταθερά" συστήματα. Στο νέφος υπάρχουν ποικιλία σε τοποθεσίες δεδομένων, δίκτυα σε μορφή ακμών, χρονική περίοδος αιτήσεων και απαντήσεων.
<b>Φήμη</b>	Όσο πιο γνωστός είναι ένας πάροχος, συνήθως είναι και ο πιο αποδοτικός σε θέματα ασφάλειας.
<b>Σταθερό και Τυποποιημένο interface</b>	Υπάρχει σταθερό και τυποποιημένο interface για όλους τους χρήστες
<b>Αποφυγή επιθέσεων</b>	Υπάρχει ανακατανομή των δεδομένων και πόρων με τη διαμόρφωση της κυκλοφορίας, τον έλεγχο ταυτότητας, την κρυπτογράφηση , με αποτέλεσμα να αποφεύγονται οι επιθέσεις
<b>Καταγραφή, έλεγχος</b>	Έλεγχος των αποδεικτικών στοιχείων με ολοκληρωτική καταγραφή των πράξεων του παρόχου προς τον πελάτη.





### 3. Ανοχή σε Σφάλματα (Fault Tolerance)

#### 3.1. Έννοια Ανοχής Σφαλμάτων (Fault Tolerance)

Οι υπηρεσίες που παρέχονται στο «Cloud Computing» διέπονται από υψηλές προσδοκίες, διότι περιέχουν πλεονεκτήματα τα οποία είναι αρκετά ωφέλιμα. Το κόστος υπηρεσιών αποτελεί πρωτεύον όφελος, διότι αποτελεί ένα οικονομικό στοιχείο, άμεσα δηλαδή συνδεδεμένο με την ανάπτυξη της χρήσης της υπηρεσίας που προσφέρεται από τους παρόχους. Ο σκοπός του «Cloud Computing» είναι να παρέχει αξιόπιστες υπηρεσίες μέσα από αξιόπιστα κέντρα δεδομένων, μέσα στα οποία λειτουργούν με ασφάλεια οι διακομιστές, οι σκληροί δίσκοι και η υποδομή του δικτύου.

Το «Cloud Computing» αποτελεί μία νέα τεχνολογία με καινούργια αρχιτεκτονική και διαχείριση τόσο των πόρων όσο και των υλικών. Οι μεγάλες εταιρίες επενδύουν πάνω σε αυτή τη τεχνολογία, ώστε να μπορέσουν να αξιοποιήσουν σε μεγάλο βαθμό το διαδίκτυο και τις παροχές ενός μεταβλητού εικονικού μηχανήματος. Η διαχείριση των πόρων μειώνει το κόστος ζωής αυτών, με αποτέλεσμα να ανοίγονται νέοι ορίζοντες για νέες έρευνες πάνω στην ήδη υπάρχουσα δομή του «Cloud Computing».

Όμως, παρά το γεγονός ότι είναι αποτελεσματικό και κερδοφόρο, έχει και σημαντικές αδυναμίες. Η πιο σοβαρή από αυτές είναι η ανοχή σε σφάλματα «Fault Tolerance». Κατά τη καταγραφή των απαιτήσεων ενός συστήματος, η ομάδα ανάπτυξης θα πρέπει να εξασφαλίσει το γεγονός αποτυχίας του συστήματος. Θα πρέπει να κατασκευάσει-ερευνήσει τεχνικές, οι οποίες θα εκτελούνται στο σύστημα και θα αποτελούν τη προστασία απέναντι σε αστοχίες, είτε αυτές είναι υλικού είτε λογισμικού. Στο περιβάλλον νέφους τα κύρια οφέλη των τεχνικών ανοχής σε σφάλματα αποτελούν η αποκατάσταση βλάβης, το χαμηλότερο κόστος αποκατάστασης και οι βελτιωμένες μετρήσεις στην απόδοση αυτών. [8]

Η ανοχή σε σφάλματα αναφέρεται στη συνεχή και ορθή λειτουργία των υπηρεσιών ακόμη και με την ύπαρξη σφαλμάτων στο λογισμικό και υλικό. Η ύπαρξη των σφαλμάτων, δηλαδή δε θα πρέπει να αποτρέπει τη λειτουργία μιας υπηρεσίας. Η ύπαρξη αυτών καθιστά απαραίτητη την απαίτηση υποδομής μηχανισμών που θα αντιμετωπίζουν τα σφάλματα αυτά. Η υποδομή θα βοηθήσει στη συνέχεια και την αύξηση της αξιοπιστίας των υπηρεσιών του παρόχου με πολλαπλά οφέλη τόσο για τον ίδιο όσο και για τους πελάτες. Η σωστή, και με αντιμετωπίσιμα σφάλματα, υπηρεσία, μπορεί να προσφέρει κέρδος για τον πάροχο, διότι θα εξυπηρετεί πολλούς πελάτες και από την άλλη πλευρά οι πελάτες θα απολαμβάνουν μια σωστή, εκτελέσιμη και αξιόπιστη υπηρεσία. [10]

Για την καλύτερη κατανόηση του όρου «Ανοχή σε σφάλματα», είναι θεμιτή η διάκριση της έννοιας σε τρεις (3) πολιτικές, σύμφωνα με τα σφάλματα που μπορεί να προκαλέσει. [10]

- Προληπτική Ανοχή σε Σφάλματα. «Proactive Fault Tolerance». Η πολιτική αυτή έχει στόχο τη πρόβλεψη και αποφυγή ανάκτησης δεδομένων και υλικού που μπορεί να προκληθεί από ένα ή περισσότερα σφάλματα. Η πολιτική στοχεύει στην εύρεση με μεθοδικό τρόπο των σφαλμάτων που μπορεί να γίνουν προτού αυτά πραγματοποιηθούν. Αποτελεί, δηλαδή, μία έννοια που αποτρέπει τις αποτυχίες (σφάλματα) υπολογίζοντας τον κόμβο, στον οποίο μπορεί να εκτελεστεί ένα σφάλμα μέσα από μια σειρά παράλληλων ενεργειών σε εικονικό περιβάλλον.



- Αντιδραστική Ανοχή σε Σφάλματα. «Reactive Fault Tolerance». Η πολιτική αυτή μειώνει τη προσπάθεια αποτυχίας, όποτε αυτή συμβεί. Η τεχνική έχει στόχο να προστατεύσει το σύστημα και να το καταστήσει πιο ισχυρό.

- Προσαρμοστική Ανοχή σε Σφάλματα. «Adaptive Fault Tolerance». Η πολιτική αυτή εξαρτάται άμεσα από το στάδιο εκτέλεσης μιας υπηρεσίας και πόσο έχει επηρεαστεί από ένα σφάλμα. Εξασφαλίζει επαρκή αξιοπιστία των κρίσιμων δεδομένων, σύμφωνα με χρονικούς περιορισμούς και πόρους. Η πολιτική σκοπεύει να προσαρμόσει τους πόρους ώστε αν προκύπτει ένα σφάλμα, άμεσα να περιορίζεται και η επίδραση που μπορεί να έχει στα δεδομένα και στο υλικό.

### 3.2. Τύποι Σφαλμάτων

Τα σφάλματα που μπορεί να προκύψουν διακρίνονται ανάλογα με τον παράγοντα που επηρεάζουν. [14] Διακρίνονται όπως αναγράφονται στο παρακάτω Πίνακας 5. Είδη Σφαλμάτων.

Πίνακας 5. Είδη Σφαλμάτων

Ονομασία	Περιγραφή
<b>Σφάλματα Δικτύου («Network Fault»).</b>	Ένα σφάλμα μπορεί να συμβεί σε ένα δίκτυο κατά τη διάρκεια διαχωρισμού δίσκων, χαμένων πακέτων, αποτυχίας αποστολής αιτημάτων, κτλ.
<b>Φυσικά Σφάλματα («Physical Fault»).</b>	Αναφέρονται τα σφάλματα που μπορεί να συμβούν στο υλικό όπως επεξεργαστές, μνήμη, αποθηκευτικό χώρο, κτλ.
<b>Σφάλματα μέσων ενημέρωσης («Media Fault»).</b>	Αποτελούν τα σφάλματα που πραγματοποιούνται όταν καταστρέφονται ή χάνονται οι επικεφαλίδες των αιτημάτων ενός δικτύου.
<b>Σφάλματα Επεξεργαστή («Processor Faults»).</b>	Σφάλματα που μπορεί να συμβούν στο λειτουργικό σύστημα.
<b>Σφάλματα Διαδικασίας («Process Faults»).</b>	Σφάλματα που εμφανίζονται στο λογισμικό με την ύπαρξη bugs και έλλειψη πόρων.
<b>Σφάλματα Λήξης Υπηρεσιών («Service Expiry Fault»).</b>	Σφάλματα που εμφανίζονται όταν οι υπηρεσίες δεν αναβαθμίζονται και emπίπτουν σε ημερομηνία λήξης.



Η αποτυχία που συμβαίνει σε ένα υπολογιστικό σύστημα μπορεί να διακριθεί, επίσης, όπως παρακάτω

- αποτυχία από παράλειψη «Omission Failure»
- αποτυχία συγχρονισμού «Timing Failure»
- αποτυχία ανταπόκρισης «Response Failure»
- αποτυχία από συντριβή «Crash Failure»

Σύμφωνα με τη παραπάνω διάκριση των αποτυχιών-σφαλμάτων, η διάκριση αυτών μπορεί να χαρακτηριστεί όπως περιγράφεται στον παρακάτω πίνακα:

**Πίνακας 6. Χαρακτηρισμός Σφαλμάτων**

Ονομασία	Περιγραφή
<b>Μόνιμο Σφάλμα («Permanent»).</b>	Αυτό συμβαίνει κατά τη διάρκεια διακοπής ρεύματος ή τάσης
<b>Διακοπτόμενο Σφάλμα («Intermittent»).</b>	Αναφέρεται σε σφάλματα που γίνονται περιστασιακά. Εμφανίζονται κατά τη διάρκεια δοκιμών λειτουργικών συστημάτων. Σε αυτή τη κατηγορία είναι δύσκολο να προβλέψεις τη βλάβη που έχει προκληθεί από το σφάλμα.
<b>Παροδικό Σφάλμα («Transient»).</b>	Το είδος αυτό των σφαλμάτων εμφανίζονται κατά την προσπάθεια ανάκτησης ή επανεκκίνησης του λογισμικού σε προηγούμενη κατάσταση.

### 3.3. Τεχνικές

Πραγματοποιείται διάκριση δύο επιπέδων ανοχής σε σφάλματα. Η πρώτη αναφέρεται σε επίπεδο εργασίας (πολιτική αντίδρασης – «Reactive Fault Tolerance») και η δεύτερη σε επίπεδο προληπτικής ροής εργασίας «Proactive Fault Tolerance». [8] Η πολιτική αντίδρασης αφορά την ανοχή σε σφάλματα που μειώνουν την επίδραση της αποτυχίας στην εκτέλεση της εφαρμογής, όταν αυτή συμβαίνει αποτελεσματικά.

Οι τεχνικές που αντικατοπτρίζουν το πρώτο επίπεδο είναι οι πολιτικές σημείου ελέγχου/επανεκκίνησης («Checkpoint/Restart»), Επανάληψη και επανα-προσπάθεια («Replay and Retry»). Στη πρώτη τεχνική, όταν πραγματοποιείται ένα σφάλμα τότε το σύστημα ξεκινά από το τελευταίο σημείο ελέγχου και όχι από την αρχή. Αυτό είναι σημαντικό καθώς το σύστημα με αυτό το τρόπο επανέρχεται γρήγορα στην τωρινή του κατάσταση. Αφορά κυρίως



εφαρμογές που εκτελούνται για μεγάλο χρονικό διάστημα. Η δεύτερη αφορά τη προσπάθεια του συστήματος να εκκινήσει την ίδια εφαρμογή από την αρχή. [15]

Επιπλέον, διακρίνεται η τεχνική της αντιγραφής («Replication»), η οποία εκτελείται με χρήση διαφορετικών πόρων. Μπορεί να πραγματοποιηθεί με τη χρήση εργαλείων όπως «HAProxy», «Hadoop», «AmazonEC2», στα οποία εκτελούνται αλγόριθμοι ανοχής σε σφάλματα. Ακόμη, η τεχνική διανομής εργασίας («Migration Job») αποτελεί μία αξιόλογη προσπάθεια εύρεσης λύσης στο πρόβλημα ανοχής σε σφάλματα, όπου οι διαδικασίες έχουν τη δυνατότητα να διανεμηθούν σε άλλο μηχανήματα, αποφεύγοντας το σφάλμα και συνεχίζοντας την εκτέλεση της υπηρεσίας ή προγράμματος. Τέλος, η τεχνική διάσωσης («Rescue Workflow») επιτρέπει στη ροή των εργασιών να συνεχίσουν ακόμη και με την ύπαρξη σφάλματος. [16] [14]

Στο δεύτερο επίπεδο ανοχής σε σφάλματα, δηλαδή στο επίπεδο προληπτικής ροής εργασίας, αφορά την αποφυγή ανάκτησης δεδομένων που προέρχονται από σφάλματα και αποτυχίες, καθώς αυτά θα οδηγήσουν στο ίδιο αποτέλεσμα, δηλαδή στο σφάλμα του συστήματος. Ο τρόπος για να πραγματοποιηθεί η τεχνική αυτή είναι η πρόβλεψη των σφαλμάτων και η ενεργή αντικατάσταση των υποπτων εξαρτημάτων ή γραμμών αν αναφερόμαστε σε πρόγραμμα, με αποτέλεσμα την αποτροπή του σφάλματος. Οι τεχνικές που βασίζονται στη πολιτική αυτή είναι τρεις (3) σε αριθμό. [8]

Η πρώτη αποτελεί τη τεχνική αναζωογόνησης («Software Rejuvenation»), όπου το σύστημα είναι σχεδιασμένο να επανακινείται σε καθορισμένο σημείο, αποφεύγοντας με αυτό το τρόπο τη μόλυνση. Η τεχνική αυτή δίνει τη δυνατότητα στο λογισμικό να εκτελεί καθορισμένη επανεκκίνηση, οπότε αν σε ένα σημείο έχει υπάρξει κάποιο σφάλμα, τότε αυτό θα επαναλάβει την εκτέλεσή του, έχοντας ανακτήσει την ορθή ύπαρξη τόσο των δεδομένων όσο και του υλικού. Η δεύτερη τεχνική αναφέρεται ως προληπτική ανοχή με χρήση αυτοάνοσων προγραμμάτων («Proactive Fault Tolerance using Self-Healing») και τέλος, η τρίτη τεχνική αναφέρεται ως προληπτική μετανάστευση («Proactive Fault Tolerance using Preemptive Migration»), η οποία βασίζεται σε μηχανισμό ελέγχου, στον οποίο οι εφαρμογές αναλύονται και παρακολουθούνται συνεχώς.

### 3.4. Αλγόριθμοι

#### 3.4.1. Αλγοριθμική Προσέγγιση

Η ανοχή σε σφάλματα απαιτεί από τους επιστήμονες και τα ερευνητικά κέντρα να μελετήσουν τις τεχνικές και τις μεθόδους αυτής, ώστε η χρήση του νέφους να αποτελέσει μια ασφαλή διαδικασία για όλους. Σήμερα, έχει γίνει αρκετά μεγάλη προσπάθεια, ώστε να αποφευχθεί οποιοδήποτε σφάλμα στο νέφος. Αν αυτό προκύψει, τότε οι τεχνικές που προαναφέρθηκαν, αποθαρρύνουν την διαρροή ή τη διαγραφή δεδομένων των χρηστών. Στη παρούσα εργασία, πραγματοποιείται προσπάθεια εύρεσης και μελέτης αλγορίθμων όπου με τη χρήση τεχνικών ανοχής σε σφάλματα θα μπορέσει να απαντήσει σε ερωτήματα για το αν η ανοχή σε σφάλματα μπορεί να διασφαλίσει τα δεδομένα με ασφάλεια. Η τεχνική που θα αξιολογηθεί είναι η αντιγραφή δεδομένων («Data Replication»). Η τυχαία αντιγραφή («Random Replication») εφαρμόζεται στη τεχνική αυτή και αποτελεί ένα σενάριο διαδεδομένο. Όμως αν ένα σφάλμα προκύψει κατά την εκτέλεση των εφαρμογών σε μια



υπηρεσία, όπως για παράδειγμα μια απώλεια τάσης ή πτώση ρεύματος, τότε το σύστημα ενδέχεται να παρουσιάσει απώλεια δεδομένων. Η αλγοριθμική προσέγγιση σε αυτή τη περίπτωση αποτελεί η εύρεση αλγορίθμων και κατ' επέκταση τεχνικών που θα μειώσουν τη πιθανότητα να συμβεί η απώλεια των δεδομένων.

### 3.4.2. «Coryset Replication»

Ο σκοπός της τεχνικής αντιγραφής «Coryset Replication» έχει σκοπό να ελαχιστοποιήσει τη πιθανότητα απώλειας δεδομένων. Αποτελεί ένα γενικής χρήσης σχήμα μεθόδου αντιγραφής με την ίδια απόδοση όπως της μεθόδου τυχαίας αντιγραφής «Random Replication». [17]

Η τεχνική «Coryset Replication» έχει ως κεντρική ιδέα μείωσης της συχνότητας χαμένων δεδομένων μετά από μια αποτυχία και πως αυτό το ποσοστό των δεδομένων μπορεί να μειωθεί. Αποτελεί ένα γενικής χρήσης σχήμα αντιγραφής, το οποίο διαχωρίζει τους κόμβους των δεδομένων σε «corysets». Η αντιγραφή των δεδομένων που βρίσκονται σε τεμάχια ενός κόμβου που ονομάζεται «chunk», αποθηκεύονται σε ένα «Coryset». Με τη παραπάνω διαδικασία συμπεραίνουμε ότι υπάρχει πιθανότητα να χαθούν τα δεδομένα ενός συστήματος όταν σε όλους τους κόμβους προκληθεί σφάλμα. Στη τεχνική «Coryset Replication» πραγματοποιείται αντιγραφή των «Coryset» σε τρεις διαφορετικούς κόμβους, με αποτέλεσμα η πιθανότητα απώλειας δεδομένων να γίνει ακόμη μικρότερη, καθώς τα δεδομένα θα βρίσκονται σε διαφορετικό κόμβο. Αποτελούν, δηλαδή τα διαχωρισθέντα στοιχεία των κόμβων.

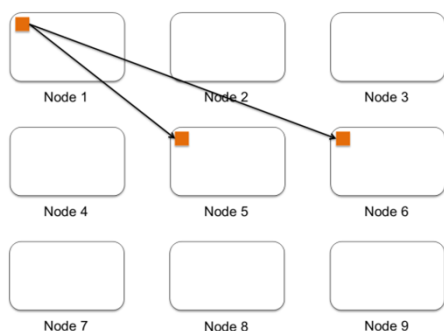
Τα δεδομένα βρίσκονται σε τεμάχια, «chunks», και αντιγράφονται συνολικά τρεις φορές σε διαφορετικούς κόμβους, δηλαδή συνολικά τα δεδομένα βρίσκονται σε τρεις ξεχωριστούς κόμβους, όπως φαίνεται και από τις παρακάτω εικόνες. Εκτελώντας τη μεθοδολογία της τεχνικής τότε παρατηρείτε ότι όλα τα δεδομένα έχουν αντιγραφεί τρεις φορές. Με αυτό το τρόπο διασφαλίζουμε την ελαχιστοποίηση απώλειας αυτών σε περίπτωση σφάλματος.

Ένα επιπλέον χαρακτηριστικό της τεχνικής αυτής αποτελεί ο καθορισμός ενός αριθμού, που ονομάζεται πλάτος σκέδασης-«Scatter Width». Ο σκοπός του αριθμού αυτού είναι να καθορίσει τον αριθμό των κόμβων που έχουν συνολικά αντιγραφεί στο σύνολο των «Coryset». Η χρήση ενός μικρού αριθμού μπορεί να προκαλέσει μικρό χρόνο ανταπόκρισης για την ανάκτηση δεδομένων, ενώ αντίθετα ένας αρκετά μεγάλος αριθμός αυξάνει τη πιθανότητα με αυξημένη συχνότητα απώλειας των δεδομένων. Παράδειγμα αποτελεί ένα σύστημα με εννέα (9) κόμβους, με πλάτος σκέδασης τέσσερα (4). Η τεχνική θα δημιουργήσει έξι (6) «Corysets», καθένα από τα οποία θα αποτελείται από τρεις (3) κόμβους. Δηλαδή το σύνολο των κόμβων αντιγράφεται όπως παρακάτω:

- Κόμβοι: 1, 2, 3, 4, 5, 6, 7, 8, 9
- «Corysets»: {1, 2, 3}, {4, 5, 6}, {7, 8, 9}, {1, 4, 7}, {2, 5, 8}, {3, 6, 9}

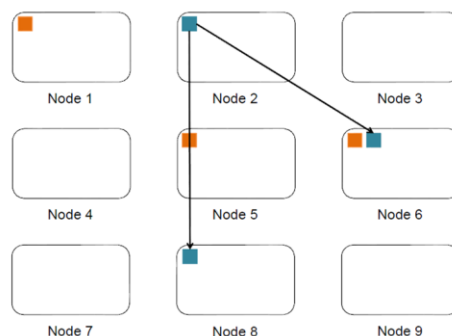


Random Replication



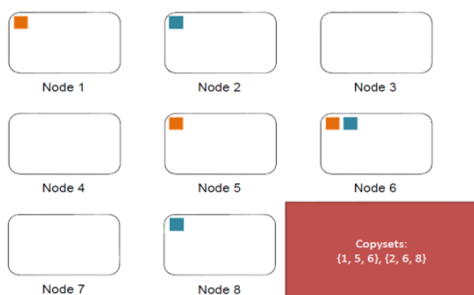
Εικόνα 10. Τυχαία Αντιγραφή (α)

Random Replication



Εικόνα 11. Τυχαία Αντιγραφή (β)

Random Replication



Εικόνα 12. Τυχαία Αντιγραφή (γ)

Random Replication



Εικόνα 13. Τυχαία Αντιγραφή (δ)

Η τεχνική «Copyset Replication» περιγράφεται σε δύο φάσεις. Η πρώτη, αποτελεί τη φάση μετάθεσης, «permutation phase», στην οποία οι κόμβοι των δεδομένων εξασφαλίζουν μία σειρά. Στην εικόνα φαίνεται η φάση μετάθεσης, όπου οι κόμβοι λαμβάνουν τον αριθμό 1 έως και 9 από αριστερά προς τα δεξιά. Η φάση αυτή είναι υπεύθυνη για τη δημιουργία των «Copysets» με τυχαίο τρόπο. Προϋπόθεση για την εφαρμογή αυτή αποτελεί το γεγονός ότι κάθε «Copyset» θα πρέπει να αποτελείται από τρεις (3) κόμβους, και κάθε κόμβος θα πρέπει να έχει αντιγραφεί τρεις(3) φορές. Στην αναφορά [17], αναφέρεται ότι στην τεχνική αυτή πραγματοποιείται η φάση μετάθεσης δύο (2) φορές. Σε κάθε φάση αυξάνεται το πλάτος σκέδασης κατά «R-1», όπου «R» είναι ο αριθμός των κόμβων που υπάρχουν σε κάθε «Copyset». Συνολικά, το πλάτος σκέδασης εκφράζεται με τον τύπο « $S=P(R-1)$ », όπου «P» είναι ο αριθμός μετάθεσης της φάσης αυτής. Συνολικά δημιουργούνται « $P * N/R$ » «Copysets». Η δεύτερη φάση, αποτελεί τη φάση αντιγραφής όπου πλέον τα δεδομένα αντιγράφονται στα «Copyset» που έχουν προκύψει από τη διαδικασία της μετάθεσης. Το ποσοστό απώλειας δεδομένων αναφέρεται ως το πηλίκο του αριθμού των «Copysets» με τον μέγιστο αριθμό των συνόλων.



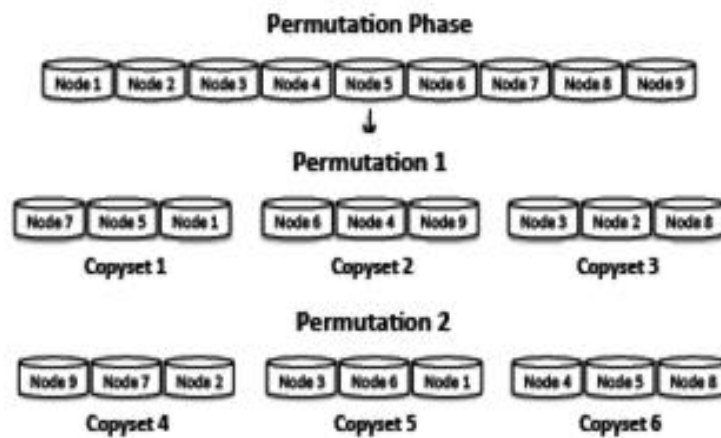
#copysets

$$\begin{pmatrix} N \\ R \end{pmatrix}$$

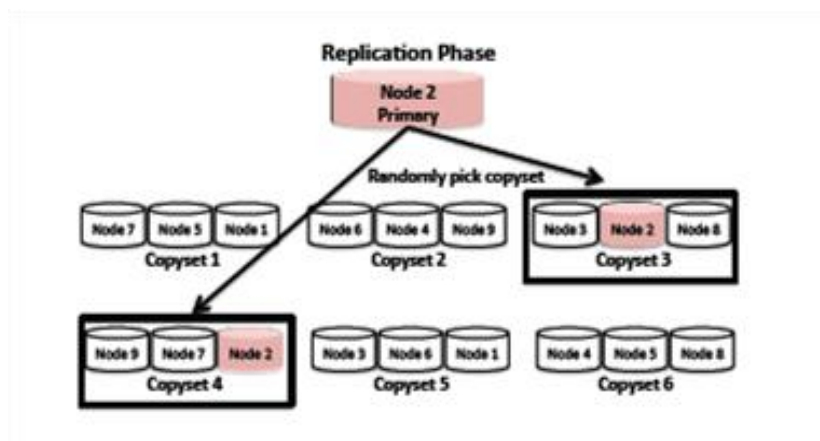
#copysets Αριθμός των «Copysets»

N: Αριθμός Κόμβων

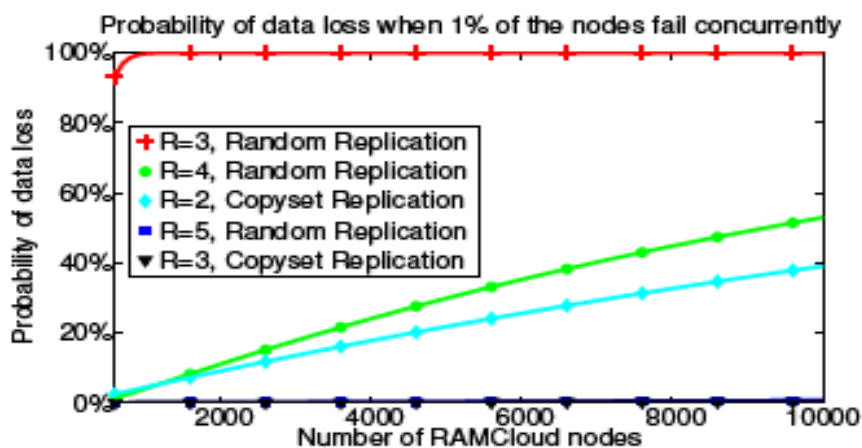
R: αριθμός των κόμβων που υπάρχουν σε κάθε «Copyset»



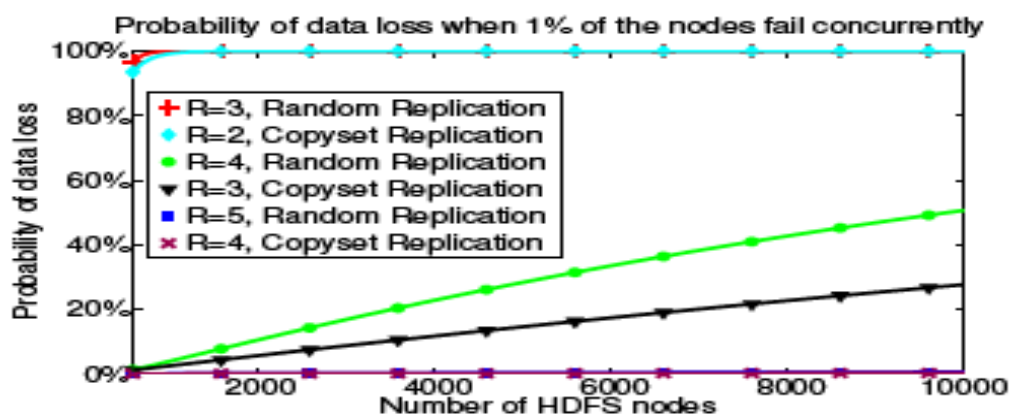
Εικόνα 14. Φάση Μετάθεσης



Εικόνα 15. Φάση Αντιγραφής

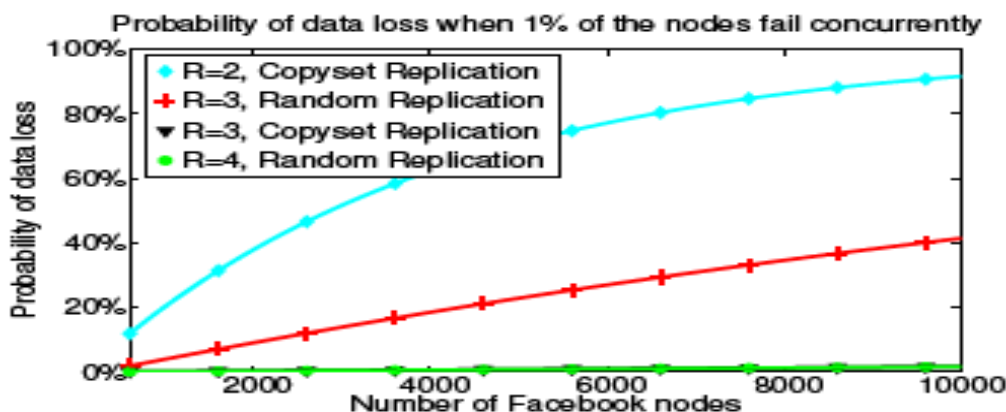


Εικόνα 16.(α) Πιθανότητα Απώλειας Δεδομένων με Σύγκριση Τεχνικών Random και Copyset Replication



Εικόνα 17.(β) Πιθανότητα Απώλειας Δεδομένων με Σύγκριση Τεχνικών Random και Copyset Replication





Εικόνα 18.(γ) Πιθανότητα Απώλειας Δεδομένων με Σύγκριση Τεχνικών Random και Copyset Replication

Οι Εικόνα 16.(α) Πιθανότητα Απώλειας Δεδομένων με Σύγκριση Τεχνικών Random και Copyset Replication, Εικόνα 17.(β) Πιθανότητα Απώλειας Δεδομένων με Σύγκριση Τεχνικών Random και Copyset Replication, Εικόνα 18.(γ) Πιθανότητα Απώλειας Δεδομένων με Σύγκριση Τεχνικών Random και Copyset Replication, παραθέτουν την εκτέλεση της τεχνικής «Copyset Replication» συγκρίνοντας τα αποτελέσματα αυτής με την εκτέλεση της τεχνικής τυχαίας αντιγραφής «Random Replication» σε διαφορετικά συστήματα που λειτουργούν σε περιβάλλον νέφους. Τα συστήματα που επεβλήθησαν στη σύγκριση των δύο (2) τεχνικών αποτελούν όπως «RAMCloud» [18] [19], «HDFS» [20], «Facebook». Παρατηρείτε ότι η τεχνική «Copyset Replication» μειώνει τη πιθανότητα απώλειας δεδομένων. Η μείωση αυτή εξαρτάται από τον αυξανόμενο αριθμό αντιγράφων των τεμαχίων των δεδομένων. Παρατηρείτε ότι ο αλγόριθμος «Copyset Replication» μειώνει τη πιθανότητα απώλειας δεδομένων σε περιπτώσεις σφαλμάτων (πχ. πτώση τάσης), για τα συστήματα «RAMCloud» και «Facebook», κοντά στο μηδέν. Από την άλλη πλευρά, δε βελτιώνεται το ποσοστό σε συστήματα «HDFS». Σε μια συστάδα κόμβων με αριθμό 5000, ο «Copyset Replication» μειώνει το ποσοστό απώλειας δεδομένων από 99.99% σε 0.15%. Για το σύστημα «Facebook» από 22.80% σε 0.78%. Επιπρόσθετα, από τα παραπάνω διαγράμματα παρατηρείτε ότι το ποσοστό μείωσης απώλειας δεδομένων, κάνοντας χρήση του «Copyset Replication» είναι ανάλογο του αυξανόμενου αριθμού αντιγράφων των κόμβων δεδομένων. Για παράδειγμα, στη περίπτωση του συστήματος «RAMCloud», εάν το σύστημα εφαρμόσει τον «Copyset Replication» με τρία (3) αντίγραφα συνολικά, έχει μειωμένες πιθανότητες απώλειας δεδομένων σε σχέση με τη τεχνική τυχαίας αντιγραφής με πέντε (5) αντίγραφα. Όμοια, ο «Copyset Replication» με τρία (3) αντίγραφα έχει την ίδια απόδοση απώλειας δεδομένων σε σχέση με την εφαρμογή της τεχνικής τυχαίας αντιγραφής με τέσσερα (4) αντίγραφα σε μια συστάδα «Facebook».

Η τεχνική «Copyset Replication» δεν αποτελεί δέσμευση ότι σε μια παρουσία σφάλματος δε θα χαθούν δεδομένα, αλλά προσπαθεί με τη χρήση της μετάθεσης και της αντιγραφής να μειώσει το ρυθμό και το ποσοστό απώλειας δεδομένων. Αποτελεί ένα γενικής χρήσης σχήμα που μπορεί να εφαρμοστεί σε οποιοδήποτε κέντρο δεδομένων («Datacenter»), με οποιοδήποτε πλάτος σκέδασης. [17]



### 3.4.3. Αλγόριθμος QADR

Η δεύτερη πρόταση που αφορά την ανοχή σε σφάλματα σε περιβάλλον νέφους αποτελεί ο αλγόριθμος «QADR(QoS-aware data replication)», ο οποίος έχει στόχο τη συνέχιση ικανοποίησης και υποστήριξης των απαιτήσεων των υπηρεσιών σε μια εφαρμογή που εκτελείται σε περιβάλλον νέφους. Πρέπει να αναφερθεί ότι η μέτρηση των σφαλμάτων που μπορεί να συμβούν στο νέφος έχουν στόχο τη παροχή δεδομένων στους πελάτες με αξιόπιστο, αποδοτικό και αποτελεσματικό τρόπο χωρίς απώλειες και καθυστερήσεις. Το ίδιο όμως δε συμβαίνει και σε επίπεδο παροχής ποιότητας των υπηρεσιών αυτών. [21] Τη στιγμή που εμφανίζεται ένα σφάλμα τότε η ποιότητα των υπηρεσιών υποβαθμίζεται και δε παρέχεται δυναμικά και με συνέχεια στους πελάτες. Το πρόβλημα που προσπαθεί να λύσει ο αλγόριθμος «QADR» είναι η απόδοση και αποτελεσματικότητα των απαιτήσεων των υπηρεσιών-εφαρμογών κατά τη εκτέλεση τεχνικών αντιγραφής. Σκοπός του «QADR» είναι δηλαδή, η ελαχιστοποίηση του κόστους αντιγραφής των δεδομένων σε τεμάχια, καθώς επίσης και η ελαχιστοποίηση του αριθμού των αντιγραφών που δεν μπορούν πλέον να εξυπηρετήσουν τις ανάγκες των υπηρεσιών.

Ο «QADR» διαιρείται σε δύο αλγορίθμους. Ο πρώτος αφορά την υιοθέτηση της ιδέας αντιγραφής των δεδομένων ανάλογα με τη σπουδαιότητά τους. Αναφέρεται ως «High-QoS First Replication - HQFR» και ο δεύτερος μετατρέπει τη γραμμική λύση του «QADR» σε αποδεκτή. Ο λόγος ύπαρξης του δεύτερου είναι το γεγονός ότι ο πρώτος αλγόριθμος αποτελεί έναν άπληστο αλγόριθμο, ο οποίος δεν μπορεί να ελαχιστοποιήσει το κόστος ελαχιστοποίησης της τυχαίας αντιγραφής των δεδομένων. Επίσης, για την καλύτερη εκτέλεση των αλγορίθμων παρέχεται η τεχνική του συνδυασμού κόμβων ώστε να μειωθεί ο χρόνος αντιγραφής μεγάλων, σε όγκο, δεδομένων. [21]

Ο «HQFR» αποτελεί άπληστο αλγόριθμο με σκοπό τη κατανομή της προτεραιότητας εκτέλεσης της αντιγραφής των δεδομένων. Αν μια εφαρμογή, για παράδειγμα, έχει υψηλό δείκτη προτεραιότητας τότε σύμφωνα με τον αλγόριθμο θα εκτελέσει τη τεχνική της αντιγραφής νωρίτερα από μια άλλη εφαρμογή με μικρότερο δείκτη. Για την καλύτερη απόδοση του αλγορίθμου πραγματοποιείται μετατροπή αυτού σε πρόβλημα γραμμικού προγραμματισμού («Integer Linear Programming - ILP»), με πολύπλοκες όμως διαδικασίες και υπολογισμούς. Οι διαδικασίες απλουστεύουν με τη μετατροπή αυτού σε πρόβλημα ελαχίστου κόστους μέγιστης ροής «Minimum-cost Maximum Flow - MCMF», το οποίο λύνει σε πολυωνυμικό χρόνο το «QADR» αλγόριθμο.

Η επίλυση του προβλήματος «QADR», επιδιώκεται από τον «HQFR» και η βέλτιστη λύση αυτού από τη μετατροπή του «QADR» σε «MCMF». Ο υπολογισμός της βέλτιστης λύσης αυτού ονομάζεται «Minimum-cost Maximum Flow Replication - MCMFR»

Ο αλγόριθμος έχει ομοιότητες με την αρχιτεκτονική του «Hadoop Distributed File» [22] για τον σχεδιασμό των αλγορίθμων αντιγραφής. Αποτελείται από ένα κεντρικό κόμβο «NameNode» και ένα σύνολο από «DataNodes», τα οποία υπάρχουν μέσα σε «Racks», όπως φαίνονται και στη παρακάτω Εικόνα 19. Αρχιτεκτονική Hadoop.

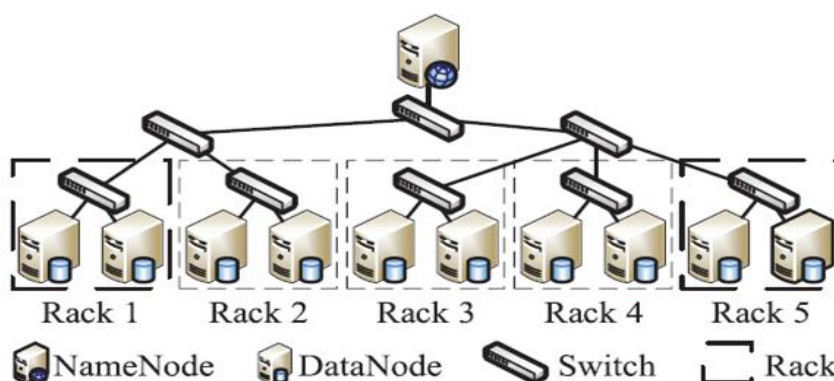
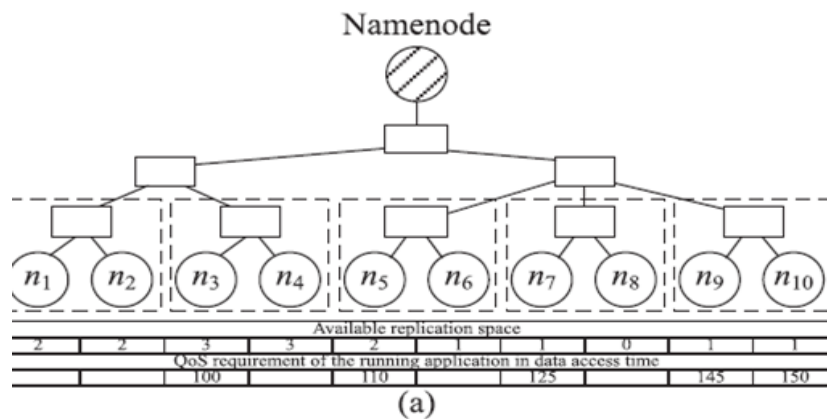


Fig. 1. The architecture of the Hadoop distributed file system.

### Εικόνα 19. Αρχιτεκτονική Hadoop

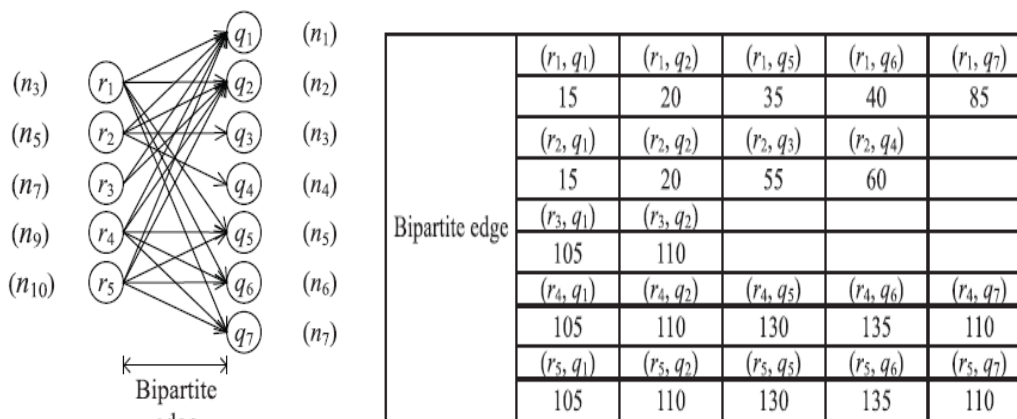
Όπως προαναφέρθηκε, ο «HQFR», έχει σκοπό να καθορίσει τη προτεραιότητα εκτέλεσης της αντιγραφής των εφαρμογών. Ο τρόπος με τον οποίο επιτυγχάνεται είναι με την αποστολή αιτήσεων αντιγραφής, ώστε να μπορεί να γίνεται έλεγχος της προτεραιότητας. Διακρίνονται τρεις (3) βασικές ιδέες-υποθέσεις. Η πρώτη αφορά τη θεώρηση ότι σε ένα κόμβο εκτελείται μία μόνο εφαρμογή σε ένα χρονικό κύκλο. Η δεύτερη αφορά τον αριθμό των αντιγράφων και αυτός δε πρέπει να ξεπερνά τα τρία (3) αντίγραφα και η τρίτη αφορά τις απαιτήσεις των υπηρεσιών. Ο «HQFR» έχει δύο καταστάσεις που θα πρέπει να πληρούνται, ώστε να είναι αποτελεσματικός. Η πρώτη, αναφέρεται στην μη ύπαρξη δεδομένων των κόμβων στο ίδιο τεμάχιο «rack» και η δεύτερη είναι ότι ο χρόνος αντιγραφής θα πρέπει να είναι μικρότερος από τον χρόνο εκτέλεσης των απαιτήσεων των υπηρεσιών. Στην Εικόνα 20. Δομή QADR Αλγορίθμου, παρατηρείται ένα σύστημα με δέκα (10) κόμβους δεδομένων μέσα σε πέντε «Racks». Σε μια χρονική στιγμή, εκτελούνται πέντε QoS αιτήματα αντιγραφής από τους κόμβους 3,5,7,9,10. Με βάση τον αλγόριθμο «HQFR», τα αιτήματα αντιγραφής ακολουθούν τη ροή 3,5,7,9,10 και καταγράφεται η σχέση των προς αντιγραφή κόμβων με τους υπόλοιπους του συστήματος. Επιπρόσθετα, παρατηρούμε, όπως φαίνεται και στις Εικόνα 21. Χρόνος πρόσβασης αντιγραφής μεταξύ κόμβων και Εικόνα 22. Γράφημα Περιγραφής Κόμβων ότι κάθε κόμβος έχει συγκεκριμένο χώρο αποθήκευσης αντιγραφών, οπότε ένα χαρακτηριστικό του αλγορίθμου είναι ότι ερευνά και αξιολογεί τη διαθεσιμότητα των αντιγραφών των δεδομένων. Ο αλγόριθμος έχει στόχο να ελαχιστοποιήσει το σύνολο των αντιγραφών των κόμβων καθώς επίσης και να ελαχιστοποιήσει το σύνολο των αντιγραφών που έχουν χαρακτηριστεί «violated». Ο χαρακτηρισμός αυτός αποδίδεται στα αντιγραμμένα δεδομένα που δεν ανταποκρίνονται στις απαιτήσεις υπηρεσιών των εφαρμογών στα οποία εκτελούνται. Ο αριθμός αυτός θα πρέπει να είναι όσο το δυνατό πιο μικρός



Εικόνα 20. Δομή QADR Αλγορίθμου

Original node	Original Destination node									
	$n_1$	$n_2$	$n_3$	$n_4$	$n_5$	$n_6$	$n_7$	$n_8$	$n_9$	$n_{10}$
$n_1$	5	20	55	60	35	40	85	90	135	140
$n_2$	15	10	55	60	35	40	85	90	135	140
$n_3$	15	20	45	60	35	40	85	90	135	140
$n_4$	15	20	55	50	35	40	85	90	135	140
$n_5$	15	20	55	60	50	65	110	115	135	140
$n_6$	15	20	55	60	60	55	110	115	135	140
$n_7$	105	110	165	170	130	135	90	115	135	140
$n_8$	105	110	165	170	130	135	110	95	135	140
$n_9$	105	110	165	170	130	135	110	115	95	140
$n_{10}$	105	110	165	170	130	135	110	115	135	100

Εικόνα 21. Χρόνος πρόσβασης αντιγραφής μεταξύ κόμβων

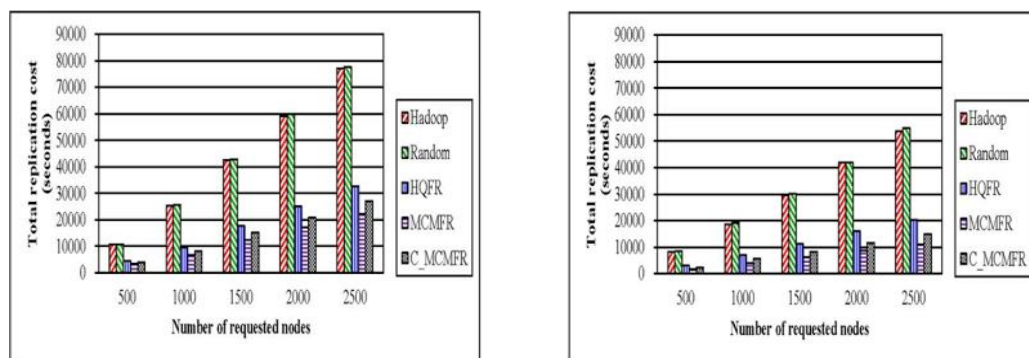


Εικόνα 22. Γράφημα Περιγραφής Κόμβων

Η εκτέλεση του αλγορίθμου πραγματοποιήθηκε με χρήση του «MatLab» [21], ώστε να αξιολογηθεί η αντιγραφή των κόμβων και κατ' επέκταση των δεδομένων σε ένα μεγάλο περιβάλλον νέφους. Ο πειραματισμός εκτελέστηκε σε 3500 κόμβους και τα αποτελέσματα φαίνονται στη παρακάτω εικόνα. Η εικόνα δείχνει το συνολικό κόστος από διαφορετικό αριθμό κόμβων από 500 έως και 2500. Η κλίμακα φανερώνει την εξάρτηση των αιτήσεων σε ένα κόμβο σε σχέση με το κόστος αντιγραφής. Τα συστήματα του «Hadoop» και των

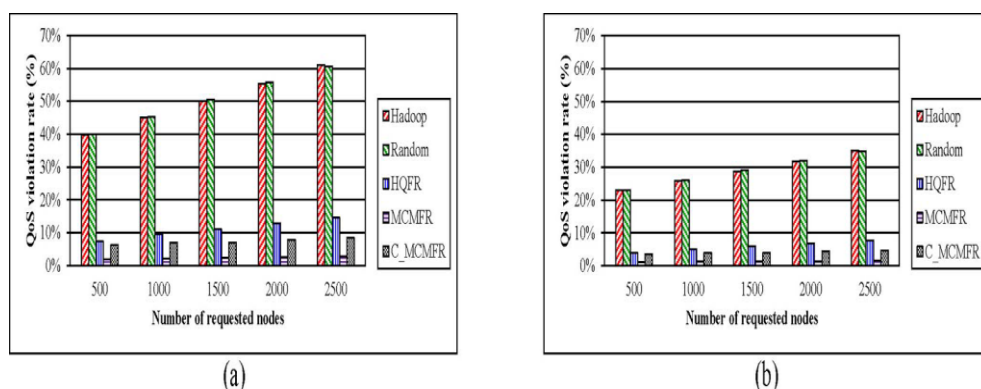


αλγορίθμων συγκρίνονται και παρουσιάζονται στην Εικόνα 23. Συνολικό κόστος αντιγραφής. Όπως παρατηρείτε, το συνολικό κόστος όλων των αλγορίθμων αυξάνεται καθώς αυξάνονται και οι κόμβοι του συστήματος. Το κόστος αντιγραφής του «Hadoop» είναι ίδιο σε σχέση με τη τεχνική της τυχαίας αντιγραφής. Και τα δύο συστήματα δεν λαμβάνουν υπόψη τις απαιτήσεις των εφαρμογών. Παρατηρείτε, ότι διαθέτουν αρκετά μεγάλο κόστος σε σχέση με τους προτεινόμενους αλγορίθμους, οι οποίοι έχουν σχεδόν το μισό από αυτά. Ο «MCMFR» είναι 2,47 φορές μικρότερος σε σχέση με το «Hadoop», ενώ σε εκτέλεση αυτών σε περισσότερους τύπους απόδοσης της συσκευής.



Εικόνα 23. Συνολικό κόστος αντιγραφής

Στη Εικόνα 24. Αναλογία Κόστους παρατηρούμε την αναλογία παραβάσεων σε σύγκριση με τους αλγορίθμους. Η αναλογία καθορίζεται ως το πηλίκο του συνόλου των απαιτήσεων που δεν ανταποκρίνονται στις απαιτήσεις προς το σύνολο των αντιγραφμένων κόμβων. Στο «Hadoop» και στους αλγορίθμους τυχαίας αντιγραφής, δεν λαμβάνονται υπόψη οι απαιτήσεις του συστήματος μιας εφαρμογής. Η πρόταση του «QADR» προσπαθεί να λύσει το πρόβλημα και να μειώσει τον αριθμό που δεν ανταποκρίνονται στις εφαρμογές τους έπειτα από τη διαδικασία της αντιγραφής των δεδομένων των κόμβων.



Εικόνα 24. Αναλογία Κόστους



### 3.5. Προσομοίωση στο Νέφος

Η αλγοριθμική προσπάθεια που αναφέρθηκε στο παρόν κεφάλαιο, αποτέλεσε αρωγός σε μια κίνηση των ερευνητών να μπορέσουν να υλοποιήσουν τις υποδομές του νέφους σε περιβάλλον προσομοίωσης. Η προσομοίωση του νέφους έχει στόχο να παρέχει όλες τις βασικές λειτουργίες αυτού, δίνοντας τη δυνατότητα στους μελετητές να ερευνήσουν και να αποδώσουν στο καλύτερο βαθμό τα αποτελέσματα των πειραμάτων στο νέφος. Η προσομοίωση του νέφους υποστηρίζει τη μοντελοποίηση των κέντρων δεδομένων του νέφους, καθώς επίσης και την εικονική ύπαρξη κέντρων υποδοχής («Hosts»). Επιπρόσθετα, η ύπαρξη και δοκιμής της υποδομής του νέφους σε περιβάλλον προσομοίωσης υποστηρίζει σε μεγάλο βαθμό τον έλεγχο της διακύμανσης του πειράματος και να εφαρμόζει τις πολιτικές που επιθυμεί, καθώς ο μελετητής μπορεί να δοκιμάζει τα πειράματά του χωρίς κόστος. Τα εργαλεία προσομοίωσης αποτελούν επεκτάσιμα προγράμματα ανάλογα με τις ανάγκες του ερευνητή του νέφους. Στα επόμενα κεφάλαια, παρουσιάζονται δύο εργαλεία προσομοίωσης του νέφους, περιγράφοντας τις βασικές δομές λειτουργίας τους, όπως επίσης και παρουσιάζοντας την ανάπτυξη, εγκατάσταση και εκτέλεσή τους.



## 4. Εργαλείο Προσομοίωσης «CloudSim»

### 4.1. Περιγραφή

Πολλές από τις υπηρεσίες του νέφους που εκτελούνται από εφαρμογές εγκατεστημένες σε αυτό, όπως υπηρεσίες κοινωνικής δικτύωσης, «web hosting», παράδοση μηνυμάτων και άμεσης απόκρισης διαδικασιών δεδομένων, απαιτούν και περιέχουν διαφορετική σύνθεση, σύσταση και απαιτήσεις για την εκτέλεσή τους. Για να μπορέσουν όμως οι υπηρεσίες αυτές να αξιολογηθούν κατά την δημιουργία τους απαιτείται η προ-εγκατάστασή τους σε περιβάλλον νέφους και η συνεχής εκτέλεση πειραμάτων. Το γεγονός αυτό όμως, η εκτέλεση συνεχόμενων πειραμάτων δηλαδή, αποτελεί μία διαδικασία που απαιτεί χρόνο και υψηλό κόστος. Η αντιμετώπιση του προβλήματος αυτού αποτελεί η ανάπτυξη και η χρήση εργαλείων προσομοίωσης του νέφους, με προεγκατεστημένες διαδικασίες χωρίς την απαίτηση εγκατάστασης των υπηρεσιών σε νέφος, αρκεί μόνο η εκτέλεση των διαδικασιών και των πειραμάτων σε περιβάλλον προσομοίωσης. Το εργαλείο προσομοίωσης «CloudSim» παρέχει βασικές κλάσεις ώστε να περιγράψει τα κέντρα δεδομένων, τα εικονικά μηχανήματα, τις εφαρμογές, τους χρήστες, τους πόρους, τις πολιτικές για διαχείριση των πόρων του συστήματος. [23] Το «CloudSim» αποτελεί μία βιβλιοθήκη κλάσεων, όπου ο ερευνητής «γράφει» και δοκιμάζει το πρόγραμμά και την εκτέλεσή του. Επίσης, αποτελεί ένα νέο, γενικής χρήσης και επεκτάσιμο πλαίσιο («framework») το οποίο επιτρέπει τη μοντελοποίηση, τη προσομοίωση και την εκτέλεση πειραμάτων σε τοπικό επίπεδο. Εκτελεί, δηλαδή ενέργειες προσομοίωσης του νέφους σε τοπικό επίπεδο, για παράδειγμα στον προσωπικό υπολογιστή. [24] Υποστηρίζει δύο επίπεδα προγραμματισμού («Scheduling»). Το επίπεδο υποδοχής («Host»), στο οποίο καθορίζεται η ισχύς κάθε πυρήνα του εικονικού μηχανήματος και στο δεύτερο επίπεδο, του εικονικού μηχανήματος («VM»), όπου καθορίζεται η επεξεργαστική ισχύς σε ανεξάρτητα γεγονότα («tasks») μέσα στο εικονικό μηχάνημα. [25] Ο σκοπός υλοποίησης ενός τέτοιου είδους εργαλείου, το οποίο προσομοιώνει το νέφος, αυξάνει τη πιθανότητα αποτελεσματικής αξιολόγησης των υπηρεσιών νέφους, εκτελώντας σε πειραματικό στάδιο διαδικασίες και δοκιμές, ώστε να διασφαλιστεί η ορθή και χωρίς λάθη παροχή αυτών των υπηρεσιών. Η παροχή υπηρεσιών σε περιβάλλον νέφους χρεώνονται από τους παρόχους ανάλογα με τη χρήση τους, οπότε η εκτέλεση αυτών σε προσομοίωση προσφέρει στους παρόχους περιβάλλον δοκιμών, με άμεσο σκοπό τη διαφύλαξη των πελατών από τη χρήση ασταθών υπηρεσιών. Στους ερευνητές, από την άλλη πλευρά, παρέχεται η δυνατότητα με τη χρήση του εργαλείου «CloudSim» να αξιολογήσουν αλγοριθμικές δοκιμές και προσεγγίσεις, ώστε η ανάλυσή τους να αποδώσει αποτελέσματα που μπορεί να χρησιμοποιηθούν για τη βελτίωση των ήδη υπαρχόντων υπηρεσιών νέφους.

### 4.2. Αρχιτεκτονική

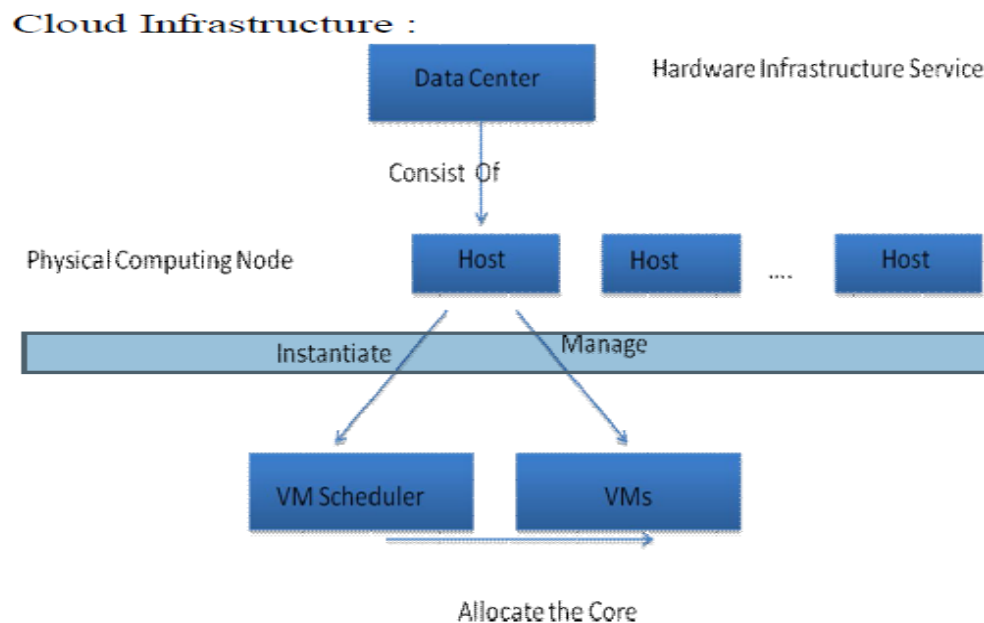
Η χρήση του εργαλείου προσομοίωσης «CloudSim» βοηθά του ερευνητές και μελετητές να αξιολογήσουν τις παρεχόμενες υπηρεσίες που προσπαθούν να αναπτύξουν. Τα βασικά πλεονεκτήματα του γεγονότος αυτού αποτελεί τόσο η χρονική διάρκεια ανάπτυξης και αξιολόγησης σε σχέση με την εξολοκλήρου ανάπτυξη του περιβάλλοντος νέφους, όσο και η ικανότητα ελιγμών και άμεσων διορθώσεων σε τυχόν προβλήματα. Το «CloudSim» υποστηρίζει την ύπαρξη κέντρων δεδομένων σε ένα φυσικό μηχάνημα, καθώς επίσης και τη βασική μοντελοποίηση του νέφους με την ύπαρξη «Service Brokers», παροχή και διαμόρφωση πολιτικών αποφάσεων. Επιπλέον, υποστηρίζει μέσω του συστήματος





Κωνσταντίνος Αθανασίου , 'Ανοχή σε σφάλματα σε υπολογιστικό περιβάλλον νέφους'

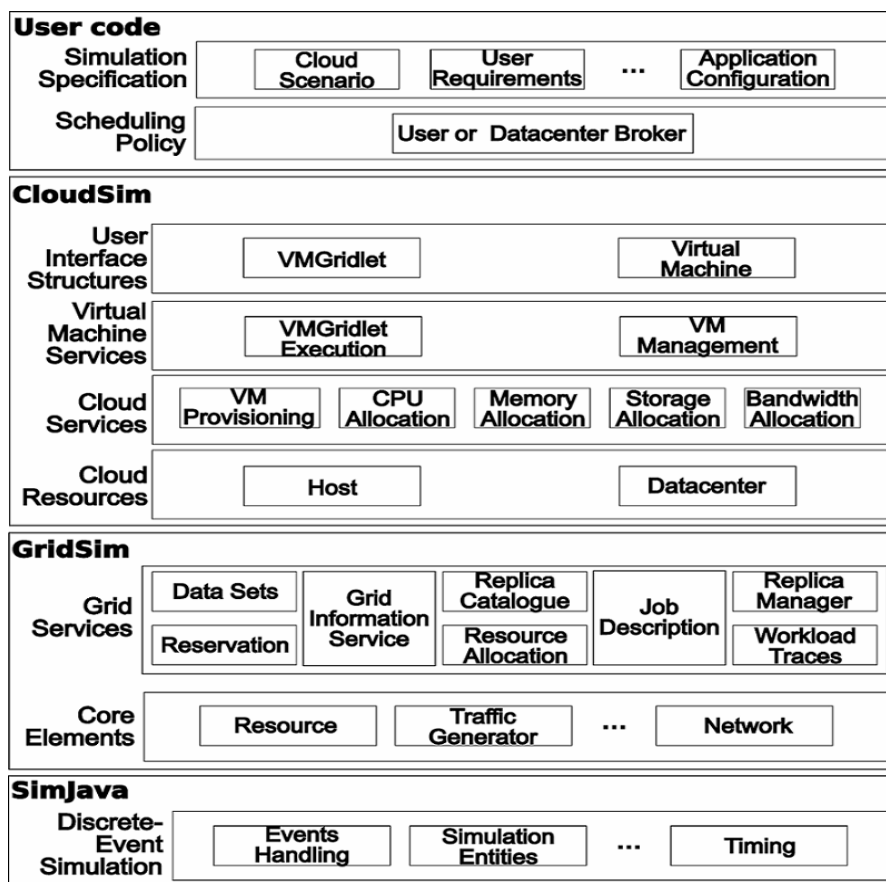
προσομοίωσης και τη προσομοίωση του δικτύου. Στην Εικόνα 25. Δομή Νέφους παρατηρείτε η γενική εικόνα του νέφους, στο οποίο τα κέντρα δεδομένων αποτελούνται από «Hosts», τα οποία διαχειρίζονται τα εικονικά μηχανήματα και το προγραμματισμό αυτών.



Εικόνα 25. Δομή Νέφους

Η αρχιτεκτονική του εργαλείου προσομοίωσης «CloudSim» αποτελείται από επίπεδα. Η αρχική έκδοση του εργαλείου περιελάμβανε το «SimJava», το οποίο αποτελεί μία μηχανή προσομοίωσης γεγονότων. Το «SimJava» υποστηρίζει διάφορες βασικές λειτουργίες όπως γεγονότων ουράς και εκτέλεσης διαδικασιών, δημιουργία οντοτήτων, όπως υπηρεσιών, «Hosts», κέντρα δεδομένων, «Broker», εικονικών μηχανημάτων. Επίσης, υποστηρίζει λειτουργίες επικοινωνίας και διαχείρισης του κύκλου λειτουργίας της προσομοίωσης.





Εικόνα 26. Δομή CloudSim προηγούμενης έκδοσης

Η νεότερη έκδοση του εργαλείου προσομοίωσης δε περιλαμβάνει το «SimJava», αλλά μια νεότερη εκδοχή αυτού. [26] Περιλαμβάνονται διαδικασίες εκτέλεσης ουράς, γεγονότων, δημιουργία οντοτήτων και επικοινωνίας μεταξύ των συστατικών του εργαλείου προσομοίωσης. Διακρίνονται τρία (3) επίπεδα όπως φαίνεται και στη παρακάτω εικόνα. Στο πιο χαμηλό επίπεδο υπάρχει ο πυρήνας της μηχανής προσομοίωσης στο οποίο καθορίζονται οι βασικές λειτουργίες όπως του «SimJava» που προαναφέρθηκε. Στο δεύτερο (2) επίπεδο διακρίνεται το μοντέλο προσομοίωσης για εικονικό περιβάλλον νέφους, όπου υπάρχει δυναμική διαχείριση των εικονικών μηχανημάτων. Στο επίπεδο αυτό, καθορίζονται η μνήμη, ο εικονικός χώρος στο δίσκο και η διαμόρφωση δικτύου. Στο πιο ψηλό επίπεδο του εργαλείου προσομοίωσης διακρίνεται ο κώδικας του χρήστη. Σε αυτό το επίπεδο, καθορίζονται οι οντότητες για τους «Hosts», ο αριθμός των εικονικών μηχανημάτων που θα δημιουργηθούν, ο αριθμός των χρηστών, καθώς επίσης και ο αριθμός των διαδικασιών που θα εκτελεστούν σε κάθε εικονικό μηχάνημα. Με τη χρήση του «CloudSim» οι ερευνητές έχουν τη δυνατότητα εκτέλεσης δοκιμών βασισμένων σε συγκεκριμένα σενάρια με αντίστοιχες διαμορφώσεις των απαιτούμενων, όπως εικονικής μνήμης, ισχύς, χώρου, πολιτικής εκτέλεσης των διαδικασιών. [23] [24] [25] [27]

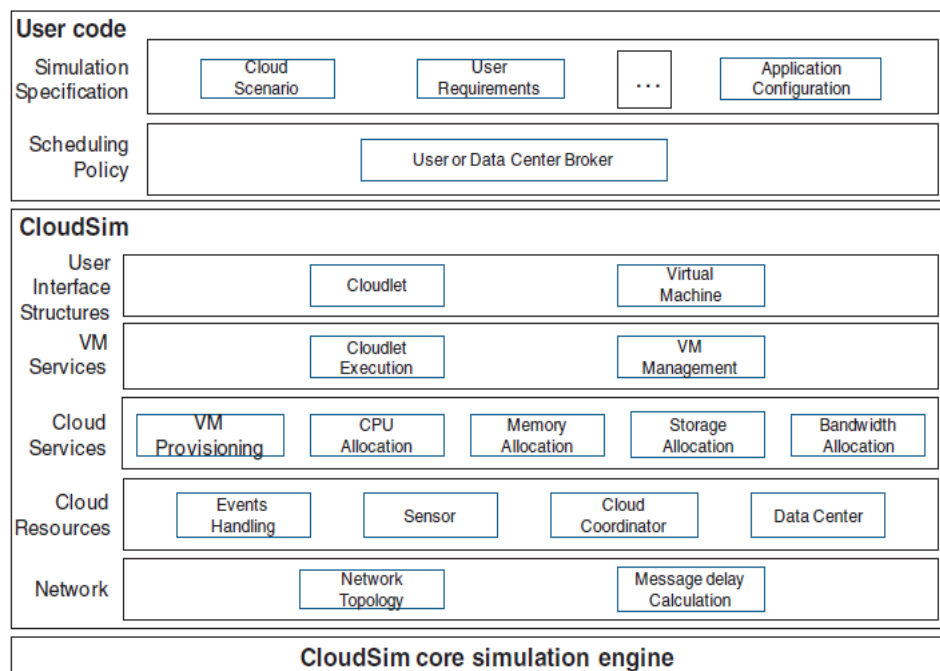


Figure 3. Layered CloudSim architecture.

### Εικόνα 27. Αρχιτεκτονική CloudSim

Πιο αναλυτικά, η παραπάνω εικόνα δείχνει την αρχιτεκτονική και τα στρώματα που χρησιμοποιεί ο προσομοιωτής νέφους «CloudSim». Στο χαμηλότερο επίπεδο διακρίνεται η μηχανή προσομοίωσης, η οποία είναι υπεύθυνη για τη διαχείριση των οντοτήτων του εργαλείου, δηλαδή είναι υπεύθυνη για τη δημιουργία, διαχείριση και διαγραφή αυτών. Το επόμενο επίπεδο, αναφέρεται στις κύριες λειτουργίες χρήσης του προσομοιωτή ως πλαίσιο. Σε αυτό το σημείο διακρίνονται κύριες οντότητες όπως το κέντρο δεδομένων, τα εικονικά μηχανήματα, οι διαδικασίες, το εύρος ζώνης κ.α. Στο δεύτερο επίπεδο καθορίζονται οι βασικές λειτουργίες που έχει ο προσομοιωτής, όπως μνήμη, ισχύς, χώρος, πως αυτά διαμοιράζονται στα εικονικά μηχανήματα καθώς επίσης και με ποια πολιτική.

Το ανώτερο επίπεδο αποτελεί τη δομή της διεπαφής με τους χρήστες της προσομοίωσης, στην οποία πραγματοποιείται επικοινωνία μεταξύ των οντοτήτων. Η επικοινωνία πραγματοποιείται με τη χρήση της διεπαφής, των εικονικών μηχανημάτων και των «Cloudlets». Το ανώτερο επίπεδο περιγράφει τον κώδικα της προσομοίωσης όπου ο χρήστης έχει τη δυνατότητα να «γράψει» και να διαχειριστεί τις ήδη υπάρχουσες κλάσεις, τροποποιώντας με αυτό τον τρόπο το πλαίσιο λειτουργίας της προσομοίωσης, με βάση τις δικές του ανάγκες μελέτης και έρευνας. Επίσης, σε αυτό το σημείο αποφασίζεται η πολιτική και ο προγραμματισμός των διαδικασιών προσομοίωσης, καθώς και οι πολιτικές απόφασης με τη χρήση του «broker» του «CloudSim».

Το εργαλείο προσομοίωσης είναι «γραμμένο» σε γλώσσα προγραμματισμού «Java», οπότε ο ερευνητής/μελετητής για να μπορέσει να ανταπεξέλθει εξολοκλήρου στο εργαλείο προσομοίωσης θα πρέπει να γνωρίσει τη γλώσσα αυτή. Η γνώση αυτή διευκολύνει στην ανάπτυξη και επέκταση του εργαλείου γράφοντας οι ίδιοι οι μελετητές δικό τους κώδικα με βάση τις δικές τους ανάγκες, χρησιμοποιώντας τις ήδη υπάρχουσες βιβλιοθήκες του



πλαίσιου. Η καλή κατανόηση του «CloudSim» είναι απαραίτητη ώστε να μπορέσει ο ερευνητής να ανταπεξέλθει στη μελέτη του καθώς και να εξάγει σωστά και τεκμηριωμένα αποτελέσματα. Το πλαίσιο έχει τη δυνατότητα επέκτασης και προσαρμογής σε οποιοδήποτε σενάριο του μελετητή, το οποίο αφορά μικρής ή μεγάλης έκτασης κέντρων δεδομένων χωρίς κόστος, μόνο με τη χρήση προσωπικού ή φορητού υπολογιστή.

### 4.3. Σχεδιασμός-Βασικές Δομές

Στη παρακάτω Εικόνα 28. Διάγραμμα Κλάσης CloudSim, αναγράφονται οι βασικές δομές του εργαλείου προσομοίωσης. Η εικόνα αποτελεί και το διάγραμμα κλάσεων του πλαισίου. [26]

Το εργαλείο προσομοίωσης αποτελείται από τις παρακάτω βασικές δομές:

- «BwProvisioner». Αποτελεί μία «Abstract» κλάση, η οποία μοντελοποιεί τη πολιτική τροφοδότησης της διαμόρφωσης του εύρους ζώνης. Κατανέμει το εύρος ζώνης σε ένα σύνολο ανταγωνιστικών εικονικών μηχανημάτων, τα οποία έχουν αναπτυχθεί στο κέντρο δεδομένων. Οι ερευνητές-προγραμματιστές των συστημάτων νέφους έχουν τη δυνατότητα επέκτασης της κλάσης αυτής με σκοπό να υλοποιήσουν δικές τους πολιτικές, ώστε να καλύψουν τις ανάγκες τους.

- «CloudCordinator». Η κλάση αυτή είναι υπεύθυνη για τη παρακολούθηση των πόρων του κέντρου δεδομένων και με βάση αυτή λαμβάνονται αποφάσεις σχετικά με τη κατανομή των πόρων. Η παρακολούθηση των πόρων του κέντρου δεδομένων πραγματοποιούνται από τη μέθοδο «updateDatacenter()». Η διάθεση των πόρων αναφέρονται με τη χρήση της μεθόδου «setDatacenter».

- «Cloudlet». Η κλάση αυτή μοντελοποιεί τις υπηρεσίες που λειτουργούν σε περιβάλλον νέφους. Οι υπηρεσίες αυτές μπορεί να είναι περιεχόμενο, κοινωνική δικτύωση ή ροή εργασίας. Το εργαλείο προσομοίωσης «CloudSim» προσομοιώνει τη πολυπλοκότητα των εργασιών σε ένα κέντρο δεδομένων δίνοντας έμφαση στις απαιτήσεις των διεργασιών/υπηρεσιών. Κάθε υπηρεσία μιας εφαρμογής περιέχει συγκεκριμένες οδηγίες εφαρμογής κατά τη διάρκεια του κύκλου ζωής της, όπως χωρητικότητα, διάρκεια και μέγεθος μεταφοράς στο δίκτυο. Οι ερευνητές έχουν τη δυνατότητα επέκτασης αυτής, ώστε να μελετήσουν μοντελοποιήσεις απόδοσης ή σύνθεσης μετρικών όπως ανταλλαγής μηνυμάτων σε βάσεις δεδομένων με γνώμονα τις εφαρμογές («Database-oriented applications»).

- «CloudletScheduler». Αφορά τη πολιτική που αποφασίζεται σχετικά με την εκτέλεση των «Cloudlet» σε ένα εικονικό μηχάνημα. Το εργαλείο προσομοίωσης διαθέτει δύο τύπους πολιτικής απόφασης, «space-shared» και «time-shared». Η πρώτη πολιτική χρησιμοποιεί τη μέθοδο «CloudletSchedulerSpaceShared», σκοπός της οποίας είναι ο διαμοιρασμός των εργασιών με βάση τη διαθεσιμότητα του χώρου. Η δεύτερη χρησιμοποιεί τη μέθοδο «CloudletSchedulerTimeShared», η οποία λειτουργεί με βάση το χρόνο, δηλαδή διαμοιράζει το διαθέσιμο χρόνο στις προς εκτέλεση υπηρεσίες.

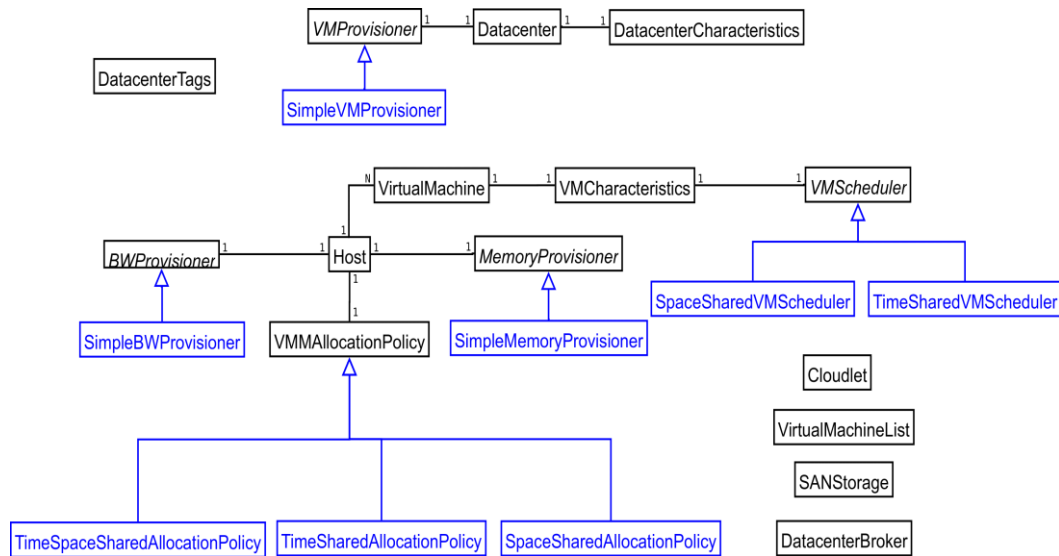
- «Datacenter». Αποτελεί το κέντρο δεδομένων. Περιλαμβάνει ένα ή περισσότερους «host», καθορίζεται η μνήμη, ο χώρος στο δίσκο, η ισχύς και τέλος,



*Κωνσταντίνος Αθανασίου , 'Ανοχή σε σφάλματα σε υπολογιστικό περιβάλλον νέφους'*

καθορίζεται η πολιτική διάθεσης των πόρων στους «hosts». Στο κέντρο δεδομένων, μπορούν να ενσωματωθούν επιπλέον «Hosts» και μπορεί να λειτουργεί ομογενοποιημένα ή με ετερογενή χαρακτήρα, ανάλογα της διαμόρφωσης του υλικού.

- «DatacenterBroker» ή «Cloud Broker». Η κλάση αυτή μοντελοποιεί έναν «Broker», ο οποίος είναι υπεύθυνος για τη συνεργασία μεταξύ των υπηρεσιών και των πόρων. Είναι υπεύθυνος για την σύνδεση και επικοινωνία μεταξύ των «SaaS» και παρόχων νέφους. Η επικοινωνία και η σχέση μεταξύ των δύο προαναφερθέντων οδηγείται κάθε φορά από τις απαιτήσεις του συστήματος. Ο «Broker» εργάζεται ως πάροχος «SaaS» υπηρεσιών. Οι μελετητές μπορούν να επεκτείνουν τη κλάση αυτή, ώστε να εφαρμόσουν διαφορετικές πολιτικές, ανάλογα με τις ανάγκες που πρέπει να καλύψουν. Η διαφορά με τον «CloudCoordinator» αποτελεί το γεγονός ότι ο «Broker» έχει σκοπό τη λήψη αποφάσεων ώστε να αυξήσει την απόδοση των μετρικών που έχει ορίσει ο χρήστης (χρήση εφαρμογών), ενώ ο «CloudCoordinator» λειτουργεί με σκοπό να βελτιώσει το κέντρο δεδομένων και την απόδοσή του
- «DatacenterCharacteristics». Διαθέτει τα χαρακτηριστικά του κέντρου δεδομένων
- «Host». Η κλάση αυτή μοντελοποιεί ένα φυσικό μηχάνημα, όπως είναι ένας «Server» για υπολογισμούς ή αποθήκευσης. Ενσωματώνει σημαντικές πληροφορίες σχετικά με το σύνολο της μνήμης και του χώρου αποθήκευσης, τους διαθέσιμους πυρήνες, δηλώνοντας με τον τρόπο αυτό την υπολογιστική ισχύ αυτού και τέλος τη πολιτική διαμοιρασμού των πόρων με τα εικονικά μηχανήματα.
- «NetworkTopology». Η κλάση περιέχει τις απαραίτητες πληροφορίες συμπεριφοράς του δικτύου της προσομοίωσης.
- «Vm». Η κλάση μοντελοποιεί ένα εικονικό μηχάνημα, το οποίο είναι βρίσκεται υπό διαχείριση από ένα «host». Οι πληροφορίες που είναι διαθέσιμες σε αυτή τη κλάση αποτελούν η μνήμη, η ισχύς, ο χώρος και η πολιτική διανομής των μηχανημάτων στους «hosts».
- «VmmAllocationPolicy». Η κλάση αυτή αναπαριστά τη πολιτική παρακολούθησης και κατανομή των «VMs» σε «Hosts». Επιλέγει τον διαθέσιμο «Host» στο κέντρο δεδομένων, το οποίο ικανοποιεί τους αντίστοιχους πόρους, όπως μνήμη και διαθεσιμότητα για την ανάπτυξη ενός εικονικού μηχανήματος.
- «VmScheduler». Αποτελεί μία κλάση που υλοποιείται σε ένα «Host» και μοντελοποιεί τις απαιτούμενες πολιτικές απόφασης («space-shared», «time-shared»), οι οποίες είναι απαραίτητες ώστε να διανεμηθεί σωστά και ικανοποιητικά η υπολογιστική ισχύς στα εικονικά μηχανήματα..



Εικόνα 28. Διάγραμμα Κλάσης CloudSim

Το εργαλείο προσομοίωσης «CloudSim» διαθέτει βασικές κλάσεις για να περιγράψει τα κέντρα δεδομένων, τα εικονικά μηχανήματα, τις εφαρμογές, τους χρήστες, τους υπολογιστικούς πόρους και τις πολιτικές που απαιτούνται. [25] Υποστηρίζει τον προγραμματισμό των εικονικών μηχανημάτων σε δύο επίπεδα. Στο πρώτο, το επίπεδο «Host» καθορίζεται πόση υπολογιστική ισχύς κάθε πυρήνα θα μοιραστεί σε κάθε εικονικό μηχάνημα. Στο δεύτερο επίπεδο, στο επίπεδο εικονικού μηχανήματος, καθορίζεται η ισχύς για κάθε διαδικασία αυτού, το οποίο βρίσκεται σε ένα «Host». Στο «CloudSim» διακρίνεται ένας βασικός διαχωρισμός προγραμματισμού των «Cloudlets». Ο καθορισμός αυτός επιτυγχάνεται με την αντίστοιχη κλάση «CloudletScheduler», η οποία καθορίζει πως οι διαθέσιμοι πόροι μοιράζονται στα εικονικά μηχανήματα, διαμέσου των «Cloudlets». [28] Η υλοποίηση του εργαλείου προτείνει δύο είδη πολιτικών. Η πρώτη αναφέρεται στην ύπαρξη συγκεκριμένου αριθμού πυρήνων σε συγκεκριμένα εικονικά μηχανήματα «Space-Shared – Cloudlet Scheduler Space Shared». Η δεύτερη πολιτική έχει τίτλο «Time-Shared – Cloudlet Scheduler Time Shared», η οποία δυναμικά διαμοιράζει την ικανότητα ενός πυρήνα σε όλα τα εικονικά μηχανήματα. Από την άλλη πλευρά, το εργαλείο προσομοίωσης «CloudSim» δίνει τη δυνατότητα προγραμματισμού των εικονικών μηχανημάτων. Η κλάση «VmScheduler» καθορίζει τον αριθμό των πυρήνων ενός «host» που αντιστοιχούν και λειτουργούν σε ένα εικονικό μηχάνημα, καθώς επίσης και πόσοι πυρήνες θα διαμοιραστούν σε κάθε μηχάνημα.

Το «CloudSim» διαθέτει ισχυρές ικανότητες που μπορεί να βοηθήσουν το μελετητή, ώστε να κατανοήσει τη λειτουργία του νέφους, τη κατανομή των εικονικών μηχανημάτων και των πόρων ενός κέντρου δεδομένων. Διαθέτει δυνατότητες παύσης και συνέχισης της προσομοίωσης, καθώς επίσης και δημιουργία νέων οντοτήτων κατά τη διάρκεια της προσομοίωσης.

Η εκτέλεση της προσομοίωσης του νέφους περιέχει την εκτέλεση ενδιάμεσων βημάτων ώστε όλες οι εικονικές λειτουργίες των μηχανημάτων να λειτουργούν και να μπορέσουν να εξάγουν αποτελέσματα πάντα ανάλογα με τις ανάγκες των ερευνητών. Αρχικά, θα πρέπει να



πραγματοποιηθεί λήψη του πακέτου από την αρμόδια ιστοσελίδα και αποθήκευση αυτού σε τοπικό ή φορητό υπολογιστή. Η αρμόδια ιστοσελίδα είναι η [29], και από την οποία πραγματοποιείται λήψη του πακέτου με όνομα «CloudSim 3.0.3». Σε συνέχεια, με τη χρήση «Integrated Development Environment -IDE» «Netbeans» [30], βοηθητικών προγραμμάτων ανάπτυξης λογισμικού και προγραμματισμού πραγματοποιείται η εγκατάσταση.

Ο χρήστης θα πρέπει να δημιουργήσει ένα καινούργιο έργο και σε συνέχεια να εισάγει τις βιβλιοθήκες του «CloudSim». Μόλις ολοκληρωθούν οι παραπάνω ενέργειες το εργαλείο προσομοίωσης είναι έτοιμο για χρήση και πειραματισμό.

Η εκτέλεση της προσομοίωσης ακολουθεί τη δομή του κύκλου ζωής, εκτελώντας διαδοχικά βήματα μέχρι να ολοκληρωθεί η προσομοίωση και εκτύπωση των αποτελεσμάτων, όπως φαίνεται και στη παρακάτω Εικόνα 29. Κύκλος Ζωής CloudSim . Αρχικά, πραγματοποιείται αρχικοποίηση του πακέτου στο εργαλείο ανάπτυξης λογισμικού. Σε συνέχεια, δημιουργούνται τα κέντρα δεδομένων («Datacenter»), καθώς επίσης και ο «Broker». Στο επόμενο βήμα είναι η δημιουργία των εικονικών μηχανημάτων. Σε αυτό το σημείο καθορίζονται οι πόροι των μηχανημάτων και έπειτα αυτά τοποθετούνται σε μια λίστα, η οποία στέλνεται στον «Broker». Η δημιουργία των «Cloudlet» με παρόμοιο τρόπο με των εικονικών μηχανημάτων, δηλαδή καθορίζονται οι πόροι, τοποθετούνται σε μία λίστα και , η οποία στέλνεται στον «Broker». Το επόμενο βήμα αποτελεί την έναρξη και λήξης της προσομοίωσης και τέλος η εκτύπωση των αποτελεσμάτων.

Συγκεντρωτικά ο κύκλος ζωής της προσομοίωσης έχει τη δομή που φαίνεται στη παρακάτω λίστα:

- Βήμα 1: Αρχικοποίηση του πακέτου
- Βήμα 2: Δημιουργία κέντρων δεδομένων
  - ο Αποτελούν του παρόχους πόρων στο εργαλείο «CloudSim». Είναι απαραίτητο τουλάχιστον ένα κέντρο δεδομένων
- Βήμα 3: Δημιουργία «Broker»
- Βήμα 4: Δημιουργία εικονικών μηχανημάτων «VM»
  - ο Περιγραφή των «VM»
  - ο Δημιουργία των «VM»
  - ο Προσθήκη των «VM» σε λίστα
  - ο Επιβεβαίωση λίστας των «VM» στον «Broker»
- Βήμα 5: Δημιουργία των «Cloudlet»
  - ο Καθορισμός ιδιοτήτων των «Cloudlet»
  - ο Προσθήκη των «Cloudlet» σε λίστα
  - ο Επιβεβαίωση λίστας των «Cloudlet» στον «Broker»
- Βήμα 6: Εκκίνηση της προσομοίωσης
- Βήμα 7: Εκτύπωση των αποτελεσμάτων όταν ολοκληρωθεί η προσομοίωση



#### CloudSim Life Cycle :



Figure : 2

Εικόνα 29. Κύκλος Ζωής CloudSim [28]

#### 4.4. Εγκατάσταση - Εκτέλεση του «CloudSim»

Η διαδικασία εγκατάστασης και εκτέλεσης της προσομοίωσης, από την εγκατάσταση έως και την εκτύπωση των αποτελεσμάτων, φαίνεται και στις ακόλουθες εικόνες.

Μετάβαση του ερευνητή στην ιστοσελίδα του «CloudSim» [29], όπου και πραγματοποιείται επιλογή του εργαλείου προσομοίωσης. Η επιλογή γίνεται με «κλικ» πάνω στον αντίστοιχο σύνδεσμο.



cloudsim  
CloudSim: A Framework For Modeling And Simulation Of Cloud Computing Infrastructures And Services

Project Home Downloads Wiki Issues Source

Search Current downloads for

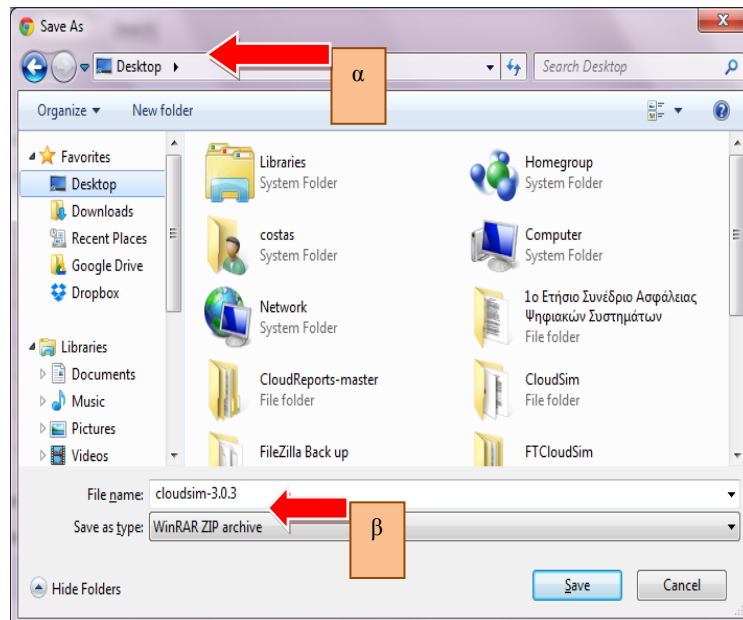
Filename	Summary + Labels	Uploaded	ReleaseDate	Size	DownloadCount
cloudsim-3.0.3.tar.gz	CloudSim 3.0.3: bug fix release	May 2013	May 2013	9.9 MB	13140
cloudsim-3.0.3.zip	CloudSim 3.0.3: bug fix release	May 2013	May 2013	13.1 MB	22614
cloudsim-3.0.2.tar.gz	CloudSim 3.0.2: bug fix release	Nov 2012	Nov 2012	9.9 MB	4513
cloudsim-3.0.2.zip	CloudSim 3.0.2: bug fix release	Nov 2012	Nov 2012	13.0 MB	4640
cloudsim-3.0.1.tar.gz	CloudSim 3.0.1: bug fix release	Oct 2012	Oct 2012	9.9 MB	759
cloudsim-3.0.1.zip	CloudSim 3.0.1: bug fix release	Oct 2012	Oct 2012	13.0 MB	1153
cloudsim-3.0.tar.gz	CloudSim 3.0	Jan 2012	Jan 2012	9.9 MB	4368
cloudsim-3.0.zip	CloudSim 3.0	Jan 2012	Jan 2012	13.0 MB	13225
cloudsim-2.1.1.tar.gz	CloudSim 2.1.1: bug fix release	Feb 2011	Feb 2011	816 KB	1102
cloudsim-2.1.1.zip	CloudSim 2.1.1: bug fix release	Feb 2011	Feb 2011	1.3 MB	3707
cloudsim-2.1.tar.gz	CloudSim 2.1: new directory structure, code refactoring and clean-up, bug fixes	Jul 2010	Jul 2010	819 KB	741
cloudsim-2.1.zip	CloudSim 2.1: new directory structure, code refactoring and clean-up, bug fixes	Jul 2010	Jul 2010	1.3 MB	2285

1 - 12 of 12

[Terms](#) - [Privacy](#) - [Project Hosting Help](#)  
Powered by [Google Project Hosting](#)

Εικόνα 30. Λήψη CloudSim

Αποθήκευση του πακέτου σε τοπικό ή φορητό υπολογιστή, όπου το (α) υποδηλώνει τη τοποθεσία και (β) την ονομασία του εργαλείου. Ο χρήστης έχει τη δυνατότητα επιλογής τοποθεσίας αποθήκευσης καθώς και μετονομασίας.



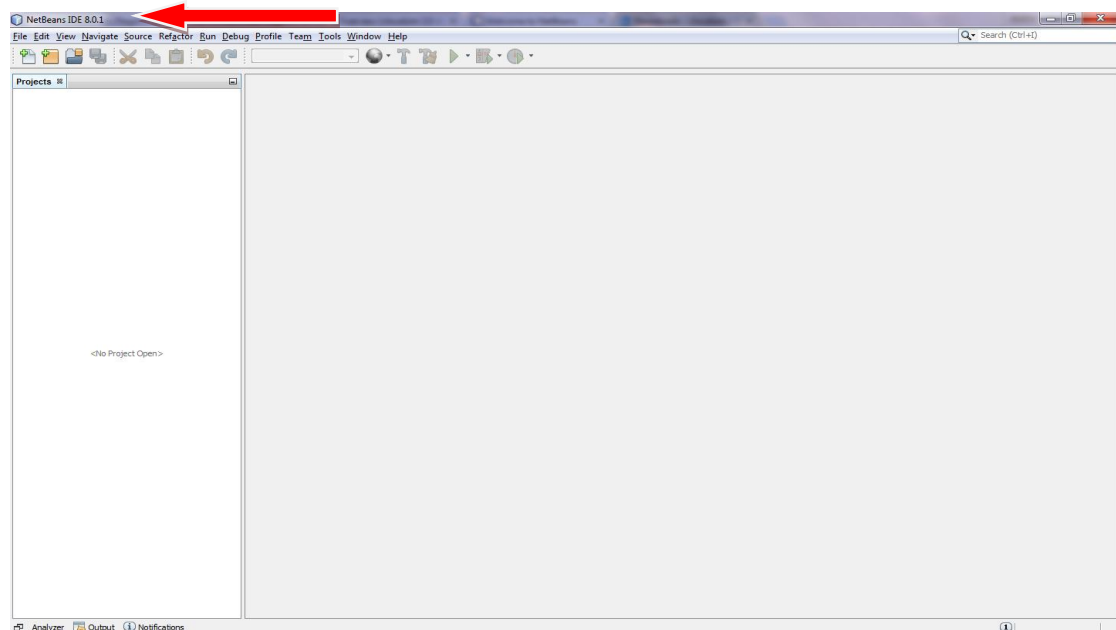
Εικόνα 31. Αποθήκευση σε τοπικό ή φορητό μηχάνημα





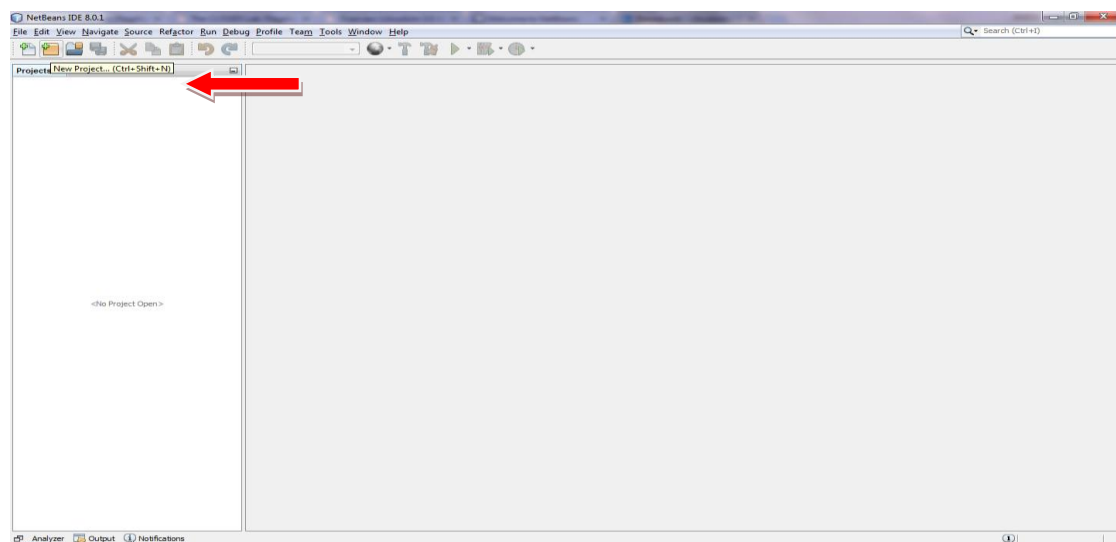
Κωνσταντίνος Αθανασίου , 'Ανοχή σε σφάλματα σε υπολογιστικό περιβάλλον νέφους'

Εκκίνηση του «IDE Netbeans», εφόσον αυτό έχει εγκατασταθεί στον υπολογιστή, με βάση τις οδηγίες που περιέχονται στον ακόλουθο σύνδεσμο [30].



Εικόνα 32. Λειτουργία IDE

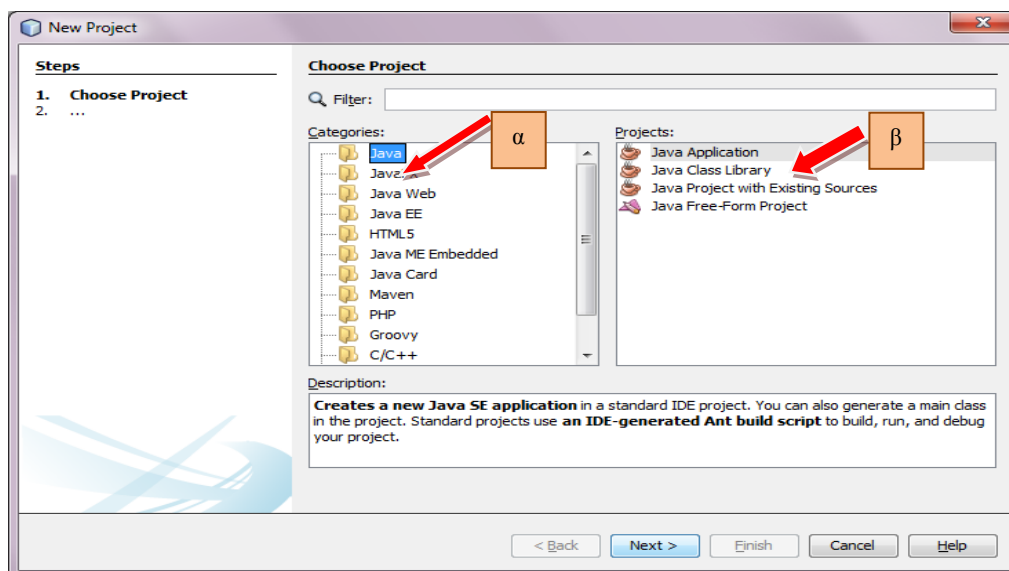
Δημιουργία νέου έργου («New Project»). Ο χρήστης του «IDE» πατώντας το «+» δημιουργεί ένα καινούργιο έργο («New Project»).



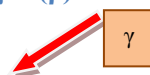
Εικόνα 33. Δημιουργία Νέου Έργου(α)



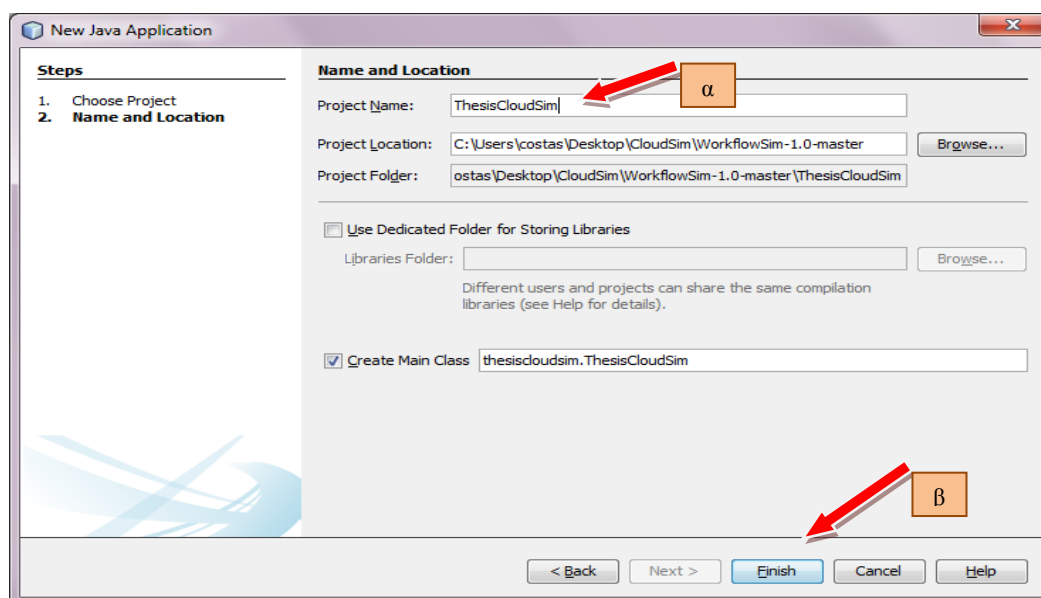
Δημιουργία Νέας Java Εφαρμογής («Java Application»). Πατώντας «New Project», ο χρήστης μεταφέρεται στην επιλογή του είδους του έργου. Σε αυτό το σημείο επιλέγονται (α) «java» και (β) «Java Application». Στη συνέχεια πατώντας το (γ) «Next >» μεταφέρεται σε επόμενη διαδικασία δημιουργίας του έργου.



Εικόνα 34. Δημιουργία Νέου Έργου(β)



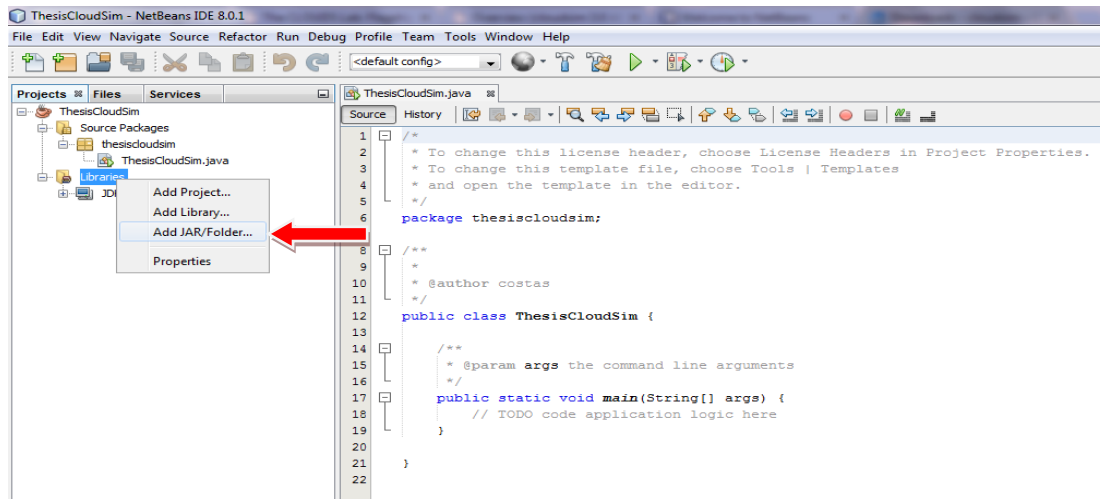
Δίνεται (α) ένα όνομα στην νέα εφαρμογή («Project Name») και (β) πατώντας «Finish» ολοκληρώνεται η δημιουργία του έργου.



Εικόνα 35. Δημιουργία Νέου Έργου(γ)

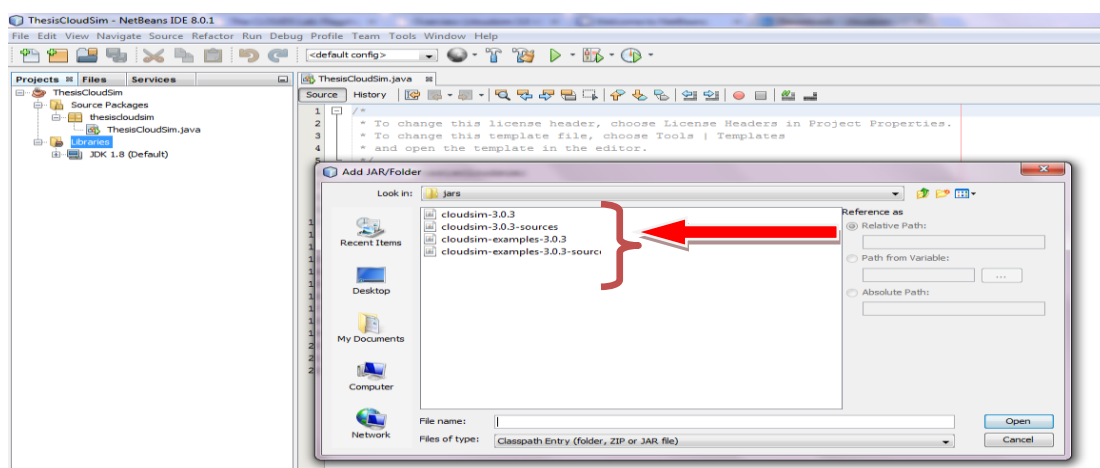


Εισαγωγή των βιβλιοθηκών του εργαλείου προσομοίωσης («Add JAR folder»). Η διαδικασία είναι ως εξής: δεξί κλικ στο πεδίο «Libraries» και πατώντας «Add JAR/Folder» ο χρήστης μεταφέρεται στην επόμενη διαδικασία.



Εικόνα 36. Εισαγωγή Βιβλιοθηκών (α)

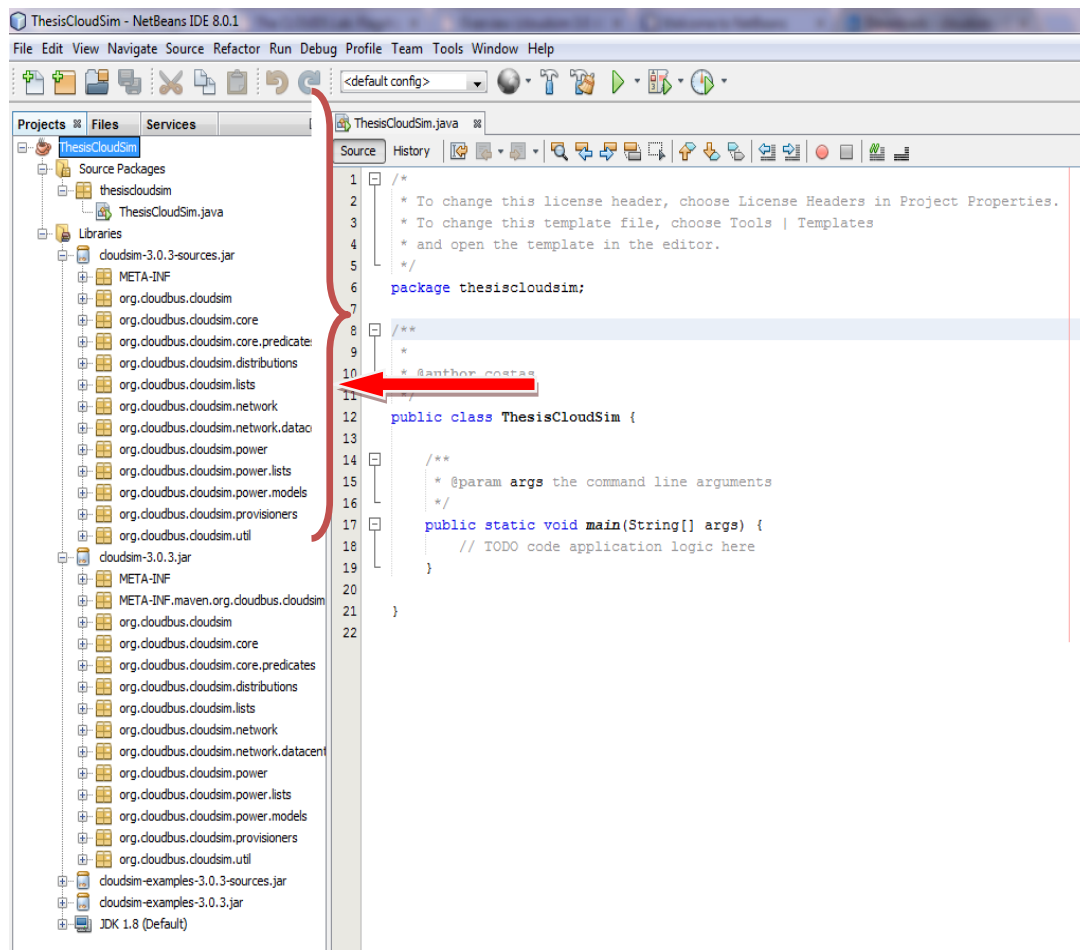
Επιλέγονται οι αντίστοιχες βιβλιοθήκες. («cloudsim-3.0.3», «cloudsim-3.0.3-sources», «cloudsim-examples-3.0.3», «cloudsim-examples-3.0.3-sources») και σε συνέχεια την επιλογή «Open».



Εικόνα 37. Εισαγωγή Βιβλιοθηκών (β)



## Προβολή οντοτήτων και κλάσεων του εργαλείου «CloudSim»

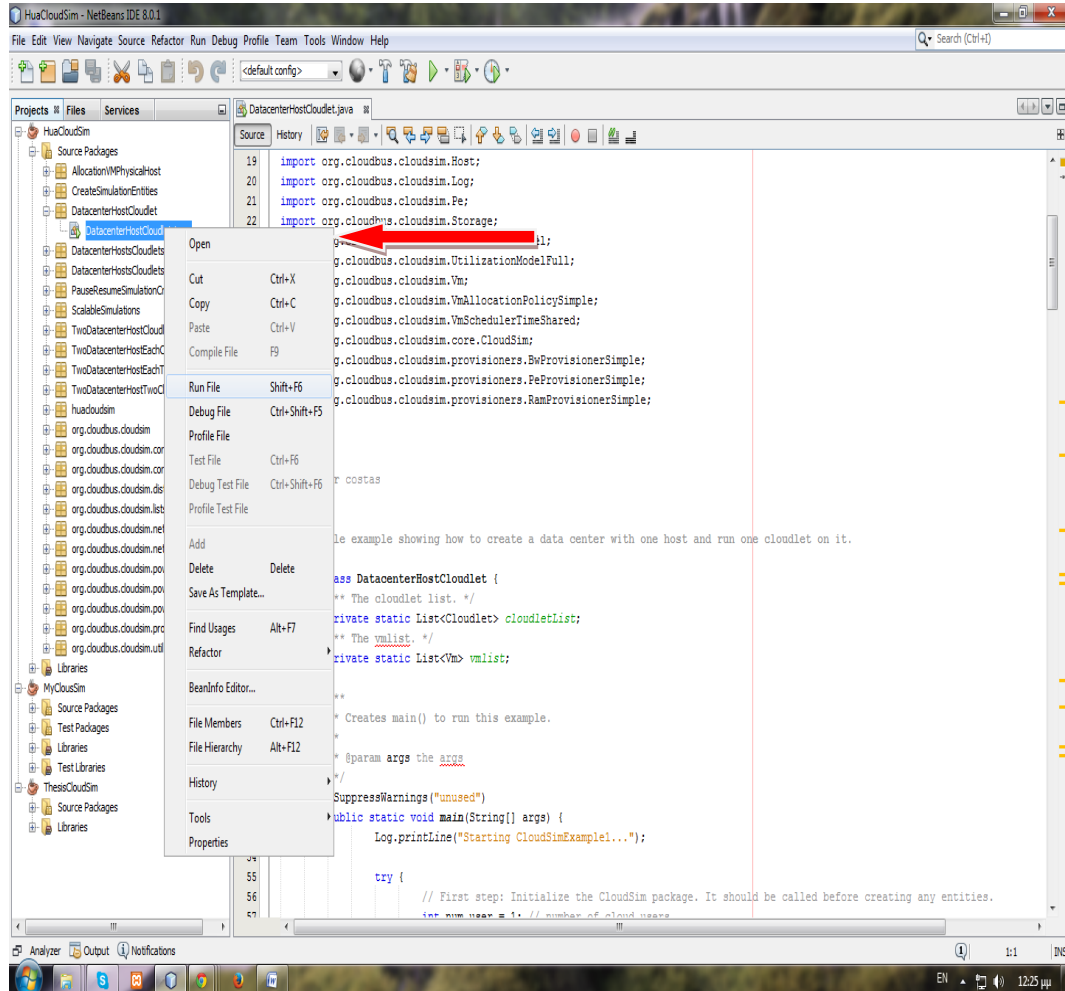


Εικόνα 38. Προβολή Οντοτήτων - Κλάσεων



Κωνσταντίνος Αθανασίου, 'Ανοχή σε σφάλματα σε υπολογιστικό περιβάλλον νέφους'

Πραγματοποιήθηκε δημιουργία προσομοίωσης, με τη δημιουργία ενός (1) κέντρου δεδομένων, ενός (1) «Host» και ενός (1) «Cloudlet». Επιλέγοντας «Run File» εκτελείτε η προσομοίωση.



Εικόνα 39. Εκτέλεση Προσομοίωσης



Προβολή αποτελεσμάτων. Στα αποτελέσματα διακρίνονται η δημιουργία όλων των βημάτων εκτέλεσης του κύκλου ζωής του «CloudSim»

```
run:
Starting CloudSimExample1...
Initialising...
Starting CloudSim version 3.0
Datacenter_0 is starting...
Broker is starting...
Entities started.
0.0: Broker: Cloud Resource List received with 1 resource(s)
0.0: Broker: Trying to Create VM #0 in Datacenter_0
0.1: Broker: VM #0 has been created in Datacenter #2, Host #0
0.1: Broker: Sending cloudlet 0 to VM #0
400.1: Broker: Cloudlet 0 received
400.1: Broker: All Cloudlets executed. Finishing...
400.1: Broker: Destroying VM #0
Broker is shutting down...
Simulation: No more future events
CloudInformationService: Notify all CloudSim entities for shutting down.
Datacenter_0 is shutting down...
Broker is shutting down...
Simulation completed.
Simulation completed.

===== OUTPUT =====
Cloudlet ID   STATUS   Data center ID   VM ID   Time   Start Time   Finish Time
0            SUCCESS    2                0       400    0,1          400,1
CloudSimExample1 finished!
BUILD SUCCESSFUL (total time: 4 seconds)
```

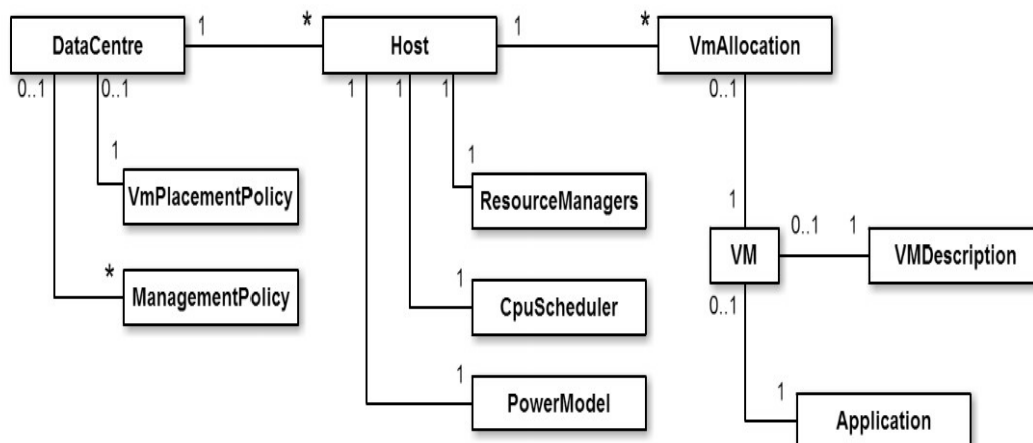
Εικόνα 40. Προβολή Αποτελεσμάτων



## 5. Εργαλείο Προσομοίωσης «DCSim»

### 5.1. Περιγραφή

Η αυξανόμενη ανάπτυξη της τεχνολογίας και η μετάβαση στο υπολογιστικό νέφος αυξάνει και τη ανάπτυξη εργαλείων προσομοίωσης ώστε να υπάρχει το περιβάλλον ανάπτυξης δοκιμών και πειραμάτων από τους μελετητές. Η διαδικασία προσομοίωσης αποτελεί μία τεχνική διαχείρισης των ενδιαφερομένων, η οποία βοηθά τους ερευνητές να αξιολογούν γρήγορα, μεθοδικά και αποτελεσματικά πραγματικές υλοποιήσεις. Το «DCSim» αποτελεί ένα εργαλείο προσομοίωσης, το οποίο προσομοιώνει ένα κέντρο δεδομένων, όπως ένα «IaaS». Το «IaaS» είναι ένας τύπος του υπολογιστικού νέφους που παρέχει στους χρήστες πρόσβαση σε χαμηλό επίπεδο πόρων και επιτρέπει σε πολλαπλές εικονικές μηχανές, τη συσχέτιση μεταξύ τους, κάτω από τη συστέγαση μια φυσικής μηχανής. Μοντελοποιεί ένα κέντρο δεδομένων με έμφαση στο διαδραστικό φόρτο εργασίας. Έχει τη δυνατότητα να διαμοιράσει το φόρτο εργασίας στα εικονικά μηχανήματα, καθώς επίσης και να υπολογίσει τη κατανάλωση ισχύος, παραβιάσεις «SLA», με σκοπό να μετρήσει την απόδοση του κέντρου δεδομένων. Το «DCSim» παρέχει ένα πλαίσιο για την ανάπτυξη και τις δοκιμές στο κέντρο δεδομένων, αλγορίθμους διαχείρισης συστημάτων «IaaS», όπου υπάρχουν πολλοί χρήστες την ίδια στιγμή. Είναι γραμμένο σε γλώσσα προγραμματισμού «Java» και είναι εύκολα επεκτάσιμο ώστε να περιλαμβάνει νέες δυνατότητες και λειτουργικότητα. [31] [32]



Εικόνα 41. DCSim Μοντέλο

### 5.2. Αρχιτεκτονική

Το «DataCentre» αποτελεί τη κεντρική κλάση της προσομοίωσης. Αποτελείται από ένα σύνολο φυσικών μηχανημάτων, τα οποία με τη σειρά τους περιέχουν ένα σύνολο από εικονικά μηχανήματα («Virtual Machines-VMs»). Ένα «VM» αποτελεί μία οντότητα, η οποία είναι υπεύθυνη για τον εγκλεισμό των εφαρμογών που πρέπει να εκτελεστούν και ποικίλει ανάλογα με τις ανάγκες των μηχανημάτων. Κάθε «VM» κατανέμεται σε ένα «Host»,



με σκοπό την εκτέλεση της εφαρμογής. Η διαμοίραση των «VMs» σε «Hosts», καθώς και πως αυτοί οργανώνονται στο κέντρο δεδομένων καθορίζεται από ένα σύνολο πολιτικών και διαχείρισης. Η Εικόνα 41. DCSim Μοντέλο , παρουσιάζει το μοντέλο «DCSim» και τα χαρακτηριστικά του. Ο σκοπός του «DataCentre» είναι να φιλοξενεί το σύνολο των «VMs», κάθε «VM» με τους αντίστοιχους πόρους να προσομοιώνουν εκτελέσεις εφαρμογών με εξωτερικό φόρτο εργασίας. Οι αρχικές απαιτήσεις πόρων καθορίζονται στη κλάση περιγραφής των «VMs», «VMDescription». Όταν ένα «VM» εκχωρηθεί σε ένα «Host», αυτό αποθηκεύει τη κατανομή με αποτέλεσμα να κατέχει το σύνολο της κατανομής των «VMs» σε «Host». Η κατανάλωση πόρων καθορίζεται από το αντίστοιχο μοντέλο. Στη περίπτωση του «DCSim» αποτελεί το «PowerModel», το οποίο διαθέτει τρία επίπεδα: «On», «Off» και «Suspended». Στη πρώτη περίπτωση «On», ο «Host» είναι σε κανονική λειτουργία, στο «Off» δε καταναλώνει ενέργεια, και στη τρίτη περίπτωση, «Suspended», καταναλώνει μικρή ποσότητα ενέργειας.

Εάν οι πόροι του «Host» χρησιμοποιούνται ευρύτατα τότε αυτός θεωρείται «Stressed». Ο χαρακτηρισμός αυτός δηλώνει ότι μία αύξηση των απαιτήσεων των πόρων δε θα μπορέσει να ικανοποιηθεί από τον «Host», με αποτέλεσμα η απαιτούμενη εργασία να μη μπορέσει να ικανοποιηθεί, προκαλώντας με τη σειρά της παραβίαση επιπέδου υπηρεσιών («Service Level Agreement»).

### 5.3. Σχεδιασμός-Βασικές Δομές

Το κέντρο δεδομένων της προσομοίωσης στηρίζεται σε πολιτικές απόφασης «Management Policies». Οι πιο σημαντικές από αυτές είναι:

- «VMPlacement Policy» : Η πολιτική αυτή είναι υπεύθυνη για το καθορισμό διανομής των «VMs» στους «Hosts». Ενεργοποιείται τη στιγμή ανάγκης διανομής ενός νέου «VM».
- «Relocation Policy» : Η πολιτική αυτή είναι υπεύθυνη για τη διανομή των «Stressed Hosts» σε άλλους «Hosts». Με τη δημιουργία της πολιτικής αυτής, καθορίζεται και η συχνότητα (συνήθως μερικές ώρες) επίκλησης των «VMs».
- «Consolidation Policy» : Η πολιτική αυτή είναι υπεύθυνη για το καθορισμό των «VMs» που μετακινήθηκαν σε άλλους «Hosts». Με τη δημιουργία της πολιτικής αυτής, καθορίζεται και η αντίστοιχη συχνότητα (συνήθως μερικές ώρες) επίκλησης των «VMs»
- «HostMonitoring Policy» : Η πολιτική αυτή είναι υπεύθυνη για την ενημέρωση της τρέχουσας κατάστασης των «Hosts».

Κάθε «VM» έχει ανάγκη από πόρους, οι οποίοι αλλάζουν δυναμικά. Οι ανάγκες αυτές καθορίζονται από μία εφαρμογή, η οποία ποικίλει ανάλογα με τις απαιτήσεις του φόρτου εργασίας. Η καθορισμένη εφαρμογή προσομοιώνει το φόρτο εργασίας ενός «Web Server». Ο φόρτος εργασίας καθορίζει το χρόνο εισερχόμενων εργασιών, οι οποίες χαρακτηρίζονται και ως αιτήματα.

Το «DCSim» αποτελεί ένα προσομοιωτή γεγονότων. Όλες οι ενέργειες, οι διεργασίες και οι αλλαγές κατάστασης ενεργοποιούνται από ένα γεγονός. Το εργαλείο διαθέτει μία ουρά προτεραιότητας, η οποία στηρίζεται στο χρόνο διενέργειας της προσομοίωσης και στο χρόνο





*Κωνσταντίνος Αθανασίου , 'Ανοχή σε σφάλματα σε υπολογιστικό περιβάλλον νέφους'*

αποστολής ενός γεγονότος, είναι υπεύθυνη για την εκτέλεση όλων των γεγονότων. Εάν ένα γεγονός αποσταλεί για εκτέλεση από τη προσομοίωση, κατά την εκτέλεση το γεγονός θα προστεθεί στην ουρά και θα εκτελεστεί όταν γίνει το πρώτο σε αυτή. Τα γεγονότα χρησιμοποιούνται επίσης, ώστε να υπάρχει επικοινωνία στο σύστημα προσομοίωσης. Τα κύρια γεγονότα είναι όπως στη παρακάτω λίστα:

- «VM Placement Event» : Όταν ένα νέο «VM» πρέπει να κατανεμηθεί, ώστε να αποφασιστεί από τη προσομοίωση πότε θα εκτελεστεί, ενσωματώνεται στο παρών γεγονός και έπειτα στέλνεται στην ουρά με την αντίστοιχη πολιτική απόφασης, η οποία στη προκειμένη περίπτωση είναι η «VM Placement Policy»
- «Instantiate VM Event» : Όταν ένα καινούργιο «VM» πρέπει να κατανεμηθεί κατά τη διαδικασία της προσομοίωσης και το εργαλείο «DCSim» έχει ήδη αποφασίσει πού θα κατανεμηθεί, το «VM» και ο στόχος του, ο «Host», ενσωματώνονται σε ένα γεγονός ενσωμάτωσης («Instantiate»)
- «Migrate VM Event» : Το γεγονός αυτό ενεργοποιείται όταν ένα ενεργό «VM», οι πληροφορίες του καθώς και οι πληροφορίες του «Host» ενσωματώνονται και πρέπει να μετεγκατασταθούν
- «Host Status Event» : Όταν η κατάσταση ενός «Host» πρέπει να διαφοροποιηθεί, η ενσωμάτωσή του καθώς και όλες οι πληροφορίες που το συνοδεύουν μεταφέρονται στο γεγονός αυτό.

Το «DCSim» παρέχει διαφορετικές μετρικές αξιολόγησης της προσομοίωσης. Αυτές είναι όπως στη λίστα που ακολουθεί:

- «SLA Violation» : Η μετρική αυτή αναφέρει το ποσοστό εισερχόμενων αιτημάτων εκτέλεσης εργασίας, τα οποία δε μπόρεσαν να ολοκληρωθούν
- «Active Hosts» : Η μετρική αυτή καταγράφει το ελάχιστο, μέγιστο και μέσο αριθμό των «Hosts» που είναι ενεργά, σε οποιαδήποτε χρονική στιγμή της προσομοίωσης
- «Host-Hours» : Η προσομοίωση καταγράφει το συνολικό χρόνο στον οποίο οι «Hosts» είναι σε ενεργή κατάσταση. Ενεργή, είναι όταν αυτοί βρίσκονται σε «ON» κατάσταση
- «Active Host Utilization» : Η μετρική αυτή καταγράφει τη μέση χρήση των ενεργών «Hosts» της προσομοίωσης
- «Number of Migration» : Η μετρική καταγράφει τον αριθμό των «VMs», τα οποία έχουν μετακινηθεί ή ανακατανεμηθούν



*Κωνσταντίνος Αθανασίου , 'Ανοχή σε σφάλματα σε υπολογιστικό περιβάλλον νέφους'*

- «Power Consumption» : Αναφέρεται στη καταγραφή της ισχύος που καταναλώθηκε κατά τη διάρκεια της προσομοίωσης. Η μετρική αυτή καταγράφει «Kilowatt-hours»
- «Simulation and Algorithm Running Time» : Η προσομοίωση καταγράφει το χρόνο προσομοίωσης, ώστε να έχει τη δυνατότητα σύγκρισης αυτού με διαφορετικούς αλγορίθμους.

Το κέντρο δεδομένων του «Cloud Computing» αποτελείται από πολλαπλούς «Servers», οι οποίοι θα πρέπει να διαχειρίζονται. Οι περισσότεροι των «Servers» αποτελούν εκτελεστικά μηχανήματα, ενώ υπάρχουν και αυτοί, οι οποίοι έχουν σκοπό τη διαχείριση όλων των «Servers» του κέντρου δεδομένων. Ένας «Server» διαχείρισης, σύμφωνα με τη θεωρία [33] [31] αποτελεί «Server» ελέγχου εκτελέσεων ενεργειών ή «Server» διαχείρισης. Ο πρώτος λειτουργεί σαν τα φύλλα ενός δέντρου και ο «Server» διαχείρισης αποτελεί τους εσωτερικούς κόμβους.

Στο «DCSim» πλαίσιο, αναπτύσσεται μία δεντρική ιεραρχική δομή, η οποία αποτελείται από τέσσερα στρώματα. Η κύρια λειτουργία της δομής αυτής είναι η δημιουργία και διαχείριση μιας ιεραρχίας στο περιβάλλον του πλαισίου της προσομοίωσης.

Το πρώτο επίπεδο αποτελεί το «Tree Management Layer», αποτελεί το σύνδεσμο μεταξύ του πλαισίου και των αλγορίθμων διαχείρισης που είναι αναπτυγμένοι σε αυτό. Είναι υπεύθυνο για τη διαχείριση της ιεραρχίας, η οποία περιέχει κόμβους για κάθε «Host» στο κέντρο δεδομένων, με λογικές αναφορές στους άλλους κόμβους που αποτελούν γονείς και παιδιά του δέντρου. Η απόφαση σχέσης μεταξύ των κόμβων αποφασίζεται με τη διεπαφή («Interface») «TreeMgnAlgorithms», η οποία διαχειρίζεται τη δομή του δέντρου. Το επίπεδο αυτό, επίσης περιέχει διεπαφές με πολιτικές ανανέωσης και παρακολούθησης της κατάστασης των «Hosts», με ονομασία «TreeHostStatusPolicy» και «TreeMonitoringPolicy».

Το επόμενο επίπεδο αποτελεί το «Aggregation Layer», στο οποίο γίνεται σαφής διαχωρισμός της κατάστασης των κόμβων με τη βοήθεια των κόμβων διαχείρισης. Το επίπεδο αυτό είναι υπεύθυνο για το συνυπολογισμό της κατάστασης του δέντρου, από τα φύλλα έως και τη ρίζα. Επιπλέον, μία σημαντική λειτουργία του επιπέδου είναι η απόφαση ενημέρωσης του δέντρου για μια αλλαγή που συμβαίνει κατά την εκτέλεση της προσομοίωσης, όπως για παράδειγμα το καθορισμό ενός «VM» σε ένα «Server». Η χρησιμοποιούμενη διεπαφή στο επίπεδο αυτό είναι η «Aggregations», η οποία χρησιμοποιείται από τους κόμβους και αποτελεί ένα είδος βιβλιοθήκης.

Συνεχίζοντας τη περιγραφή του «DCSim» πλαισίου διακρίνεται το στρώμα των αλγορίθμων «Algorithm Layer». Αποτελεί το ανώτερο στρώμα της αρχιτεκτονικής και περιέχει υλοποιήσεις διεπαφών που περιέχονται στο επίπεδο «Tree Management Layer».

Η συμπεριφορά του κέντρου δεδομένων καθορίζεται από την επιλογή αλγορίθμου του κόμβου διαχείρισης. Αρχικά, το «DCSim» πλαίσιο, διαθέτει τον κύριο και βασικό αλγόριθμο δημιουργίας της ιεραρχικής δομής, με ονομασία «BasicTreeMgntAlgorithm», ο οποίος αποφασίζει τη σύνθεση των κόμβων, ώστε κάθε ένας κόμβος να έχει ένα γονέα και παιδιά.



Με τον τρόπο λειτουργίας του καθορίζεται ότι οι κόμβοι δε θα αποτελούν ταυτόχρονα εκτελέσιμοι ή κόμβοι διαχείρισης.

Η πολιτική επιλογής ενός κόμβου από την ιεραρχική δομή αποτελεί τη «BasicTreeNodeSelectionPolicy». Η βασική της υλοποίηση, επιλέγει με τυχαίο τρόπο, ένα κόμβο διαχείρισης που βρίσκεται στα χαμηλότερα στρώματα του δέντρου. Η πολιτική αυτή αναφέρεται στη περίπτωση όπου ένα νέο εικονικό μηχανήμα πρέπει να τοποθετηθεί σε ένα «Host». Ο αλγόριθμος στέλνει ένα αίτημα ώστε να ερευνηθεί και να βρεθεί το νέο εικονικό μηχανήμα. Στη περίπτωση μιας δεντρικής δομής, ο αλγόριθμος εξυπηρετεί την εύρεση και προσθήκη ενός νέου κόμβου στο δέντρο.

Η υλοποίηση βασικού αλγορίθμου διαχείρισης αποτελεί τη «BasicMgntAlgorithm» και είναι υπεύθυνη για την απόφαση κατανομής ενός νέου «VM» σε «Host». Επίσης, είναι υπεύθυνος για τη μετεγκατάσταση των «VM» σε «Host», με σκοπό τη καλύτερη απόδοση αυτών, εάν αυτό θεωρηθεί απαραίτητο κατά τη διάρκεια της προσομοίωσης. Η υλοποίηση του αλγορίθμου διαθέτει τις παρακάτω βασικές κλάσεις:

- «MaxCPUAgregation» : Αναφέρεται στη γνώση των κόμβων σχετικά με τη μέγιστη διαθέσιμη υπολογιστική ισχύ σε ένα «Host»
- «MaxMemoryAggregation» : Αναφέρεται στη γνώση των κόμβων σχετικά με τη μέγιστη διαθέσιμη μνήμη σε ένα «Host»
- «MaxCPUInOnStateAggregation» : Αναφέρεται στη μέγιστη υπολογιστική ισχύ όταν ένας «Host» είναι σε κατάσταση «On»
- «StressedHostAggregation» : Αναφέρεται στη «Stressed» κατάσταση που μπορεί να βρεθεί ένας «Host» με ενημέρωση των αντίστοιχων κόμβων του δέντρου
- «UnderutilizedAggregation» : Αναφέρεται στη γνώση που παρέχεται από τη κλάση αυτή σχετικά με τη διάθεση προς χρήση διαθέσιμων «Host»

Κατά την αρχικοποίηση ενός νέου «VM», πρέπει να τοποθετηθεί σε ένα «Host». Η πολιτική που είναι αρμόδια για τη διαδικασία αυτή είναι η «VM Placement Policy», η οποία ορίζει όλες τις διαθέσιμες πληροφορίες σχετικά με τους πόρους του «VM» και τις απαιτήσεις του. Στη συνέχεια, η απόφαση του κόμβου προορισμού είναι στην αρμοδιότητα του «BasicMgntAlgorithm», ο οποίος αναφέρει τη διαθεσιμότητα σε πόρους (υπολογιστική ισχύ, μνήμη) και ερευνά το κατάλληλο «VM». Η ίδια διαδικασία πραγματοποιείται όταν τα διαθέσιμα «VMs» θεωρηθούν «Stressed» και δεν υπάρχουν καινούργια εισερχόμενα «VMs». Κατά την εκτέλεση του αλγορίθμου, εάν διαπιστωθεί μη ικανοποίηση των απαιτήσεων και διαθέσιμων πόρων, και επίσης, καμία διάθεση τοποθέτησης του «VM», τότε η τοποθέτηση των εικονικών μηχανημάτων θεωρείται αποτυχημένη.

Η μετατόπιση («Relocation») ενός «VM» από τη κατάσταση «Stressed» σε «non-Stressed», ώστε να αποφεύγεται η ύπαρξη «Stressed Hosts» στο κέντρο δεδομένων, με άμεση συνέπεια την μείωση ύπαρξης σφαλμάτων κατά τη διάρκεια εκτέλεσης ενεργειών, είναι υπαρκτή στο εργαλείο προσομοίωσης με την εφαρμογή διαχείρισης των «Stressed Hosts» με ονομασία «StressedHostAggregation». Η διαδικασία αυτή χρησιμοποιηθεί τις διαδικασίες εύρεσης μέγιστης υπολογιστικής ισχύς και μνήμης και αποφασίζει την μετατόπιση των «VMs» σύμφωνα με την απόφαση του «BasicMgntAlgorithm».



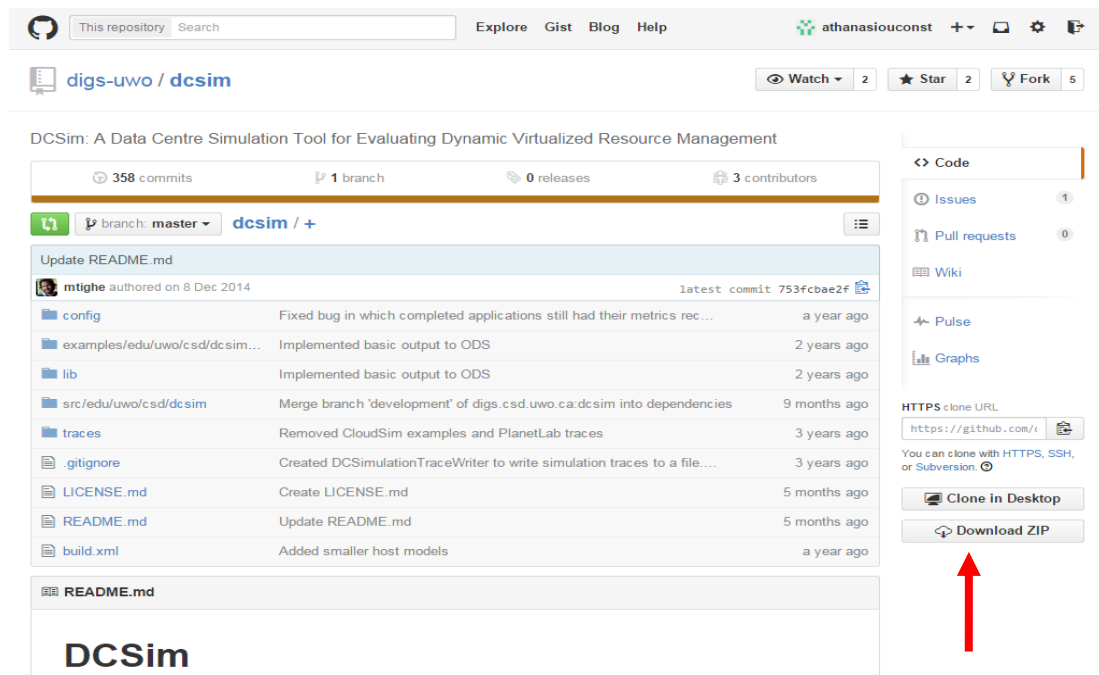
Κωνσταντίνος Αθανασίου, 'Ανοχή σε σφάλματα σε υπολογιστικό περιβάλλον νέφους'

Τέλος, η ενοποίηση («Consolidation») των «VMs» ενεργοποιείται όταν οποιοσδήποτε κόμβος μεταβαίνει σε ανενεργή κατάσταση. Η ενοποίηση έχει στόχο να ενώσει εικονικά μηχανήματα με «Hosts», ώστε να μειώσει τη κατανάλωση ισχύος στο κέντρο δεδομένων. Για κάθε εικονικό μηχάνημα που βρίσκεται σε «Host», ο οποίος αδρανοποιείται, ενεργοποιείται μηχανισμός που ελέγχει το χώρο και τους διαθέσιμους πόρους, ώστε να τοποθετήσει εκ νέου το εικονικό μηχάνημα.

## 5.4. Εγκατάσταση – Εκτέλεση του «DCSim»

Η διαδικασία εγκατάστασης και εκτέλεσης της προσομοίωσης, από την εγκατάσταση έως και την εκτύπωση των αποτελεσμάτων, φαίνεται και στις ακόλουθες εικόνες.

Μετάβαση του ερευνητή στην ιστοσελίδα του «GitHub» και σε συνέχεια στο χρήστη «DCSim» [34], όπου και πραγματοποιείται επιλογή του εργαλείου προσομοίωσης. Η επιλογή γίνεται με «κλικ» πάνω στον αντίστοιχο σύνδεσμο επιλογής λήψης σε μορφή «.zip».



Εικόνα 42. Λήψη του "DCSim"

Ο μελετητής εργάζεται με τον ίδιο τρόπο, όπως και με το «CloudSim», δηλαδή αποθηκεύει το αρχείο σε τοπικό μηχάνημα και εισάγει τα αρχεία «.jar» που βρίσκονται εσωτερικά των φακέλων που ελήφθησαν, μόλις δημιουργήσει ένα καινούργιο «Project» στο αντίστοιχο εργαλείο προγραμματισμού που διαθέτει.

Το αποτέλεσμα των παραπάνω ενεργειών είναι η δημιουργία ενός έργου με ονομασία «DCSim», το οποίο περιέχει τις βιβλιοθήκες που είναι απαραίτητες για την εκτέλεση της προσομοίωσης.



## 6. Συμπέρασμα - Επέκταση

### 6.1. Συμπέρασμα

Η ανοχή σε σφάλματα αποτελεί μια πολύ ευαίσθητη περιοχή, καθώς καλύπτει την ανάγκη των ενδιαφερομένων να μπορούν να ελέγξουν τα δεδομένα τους και τις υπηρεσίες τους. Οι ενδιαφερόμενοι, μπορεί να αποτελούν εταιρίες-πάροχοι ή και απλοί χρήστες του διαδικτύου. Η ανοχή σε σφάλματα, συμπερασματικά, παρέχει την ιδιότητα στις υπηρεσίες να είναι ελαστικά δυναμικές, ενσωματώνοντας τις τεχνικές ανοχής σε βασικές διεργασίες εκτέλεσης αυτών. Η ύπαρξη μηχανισμών και τεχνικών ανοχής απλοποιεί την ανάπτυξη νέων εφαρμογών και υπηρεσιών, καθώς οι χρήστες εργάζονται σε ασφαλές περιβάλλον ανάπτυξης και διαχείρισης αυτών. Επιπρόσθετα, η πολυπλοκότητα του συστήματος του νέφους έχει ανατεθεί στον πάροχο, ο οποίος είναι και ο άμεσα εμπλεκόμενος για να βρεθεί λύση σε προβλήματα, όπως επίσης να υποστηρίξει την ανοχή σε σφάλματα για τις δικές του παρεχόμενες υπηρεσίες και προγράμματα υποδομών νέφους.

Η αυξανόμενη ζήτηση για ευελιξία και επεκτασιμότητα σε δυναμικό περιβάλλον υπολογισμών και εφαρμογών, χωρίς αποδέσμευση των υπολογιστικών πόρων και με μικρό κόστος ανάπτυξης αυτών έχει οδηγήσει την υιοθέτηση του νέφους και της ανοχής σε σφάλματα. Η υποδομή του νέφους έχει επεξεργαστεί την ανοχή σε σφάλματα και προσπαθεί να προσφέρει την ιδιότητα αυτή στις υπηρεσίες και στα προγράμματα που λειτουργούν σε περιβάλλον νέφους. Αποτελεί ζωτικής σημασίας διασφάλιση εκτέλεσης υπηρεσιών και εξ ολοκλήρου του συστήματος. Η επιλογή των λύσεων αποδίδεται στην εφαρμογή των τεχνικών αντιμετώπισης των σφαλμάτων στο νέφος τόσο από παρόχους όσο και από χρήστες.

Η μελέτη και τα εργαλεία προσομοίωσης αποτελούν ενδεικτικό παράδειγμα του ενδιαφέροντος των μελετητών να μπορέσουν να κατανοήσουν το «Cloud Computing», τα πλεονεκτήματα που προσφέρει σε σχέση με τη συμβατική χρήση των υπολογιστικών πόρων, καθώς και να μπορέσουν να εφαρμόσουν πάνω σε αυτά τεχνικές ανοχής σε σφάλματα, προσπαθώντας με αυτό τον τρόπο να προσφέρουν τη λύση απώλειας των δεδομένων σε περιπτώσεις σφαλμάτων.

Στη παρούσα διπλωματική εργασία, πραγματοποιήθηκε προσπάθεια μελέτης της ανοχής σε σφάλματα δίνοντας αναφορά σε τεχνικές και διαδικασίες, οι οποίες έχουν σκοπό να προστατεύσουν την απώλεια δεδομένων από παράγοντες όπως πτώση τάσης ή διακοπή δικτύου ή και καταστροφή υλικού. Δόθηκε, επίσης η ανάλυση δύο προσομοιώσεων του υπολογιστικού νέφους, μέσα από τα οποία μπορούμε να αντλήσουμε πολλές πληροφορίες για τη λειτουργία του νέφους.

Επιπρόσθετα, η ανάλυση των προσομοιώσεων πραγματοποιήθηκε με σκοπό τη χρήση για επέκταση αυτών εγκαθιστώντας την ιδιότητα που αναπτύχθηκε σε αυτή την εργασία, της ανοχής σε σφάλματα. Η υλοποίηση των αλγορίθμων και τεχνικών ανοχής σε σφάλματα είναι αναγκαία, καθώς με τον τρόπο αυτό δίνεται ξεκάθαρα ο τρόπος επίλυσης προβλημάτων στο νέφος. Προβλήματα υλικού, λογισμικού και δεδομένων, με την ύπαρξη μηχανισμών ανοχής σε σφάλματα μπορούν να λύνονται κατά τη ανάπτυξη και εκτέλεση των προσομοιώσεων.

Ένα βασικό χαρακτηριστικό των προσομοιώσεων είναι ότι εκτελούν τις βασικές διεργασίες με τρόπο συνεχή, χωρίς να δύναται η διακοπή τόσο αυτόματα όσο και από το



*Κωνσταντίνος Αθανασίου , 'Ανοχή σε σφάλματα σε υπολογιστικό περιβάλλον νέφους'*

χρήστη. Η δομή των προσομοιώσεων είναι κατασκευασμένη με τέτοιο τρόπο, ώστε με την εγκατάσταση ενός κέντρου δεδομένων να παρέχονται οι βασικές πολιτικές διανομής των πόρων των υπό ανάπτυξη εικονικών μηχανημάτων και σε συνέχεια, αφού προωθηθούν στη βασική διεργασία διαμοίρασης των πόρων, να εκτελούνται άνευ διακοπής.

Η συνεχή εκτέλεση των διαδικασιών των προσομοιώσεων αποτέλεσε ένα πρωταρχικό πρόβλημα για την υλοποίηση των αλγορίθμων ανοχής. Με την υπάρχουσα δομή αυτών, υπάρχει η δυνατότητα να εφαρμοστεί η διενέργεια αντιγραφής πακέτων δεδομένων, αλλά δε υπάρχει πρόβλεψη διακοπής της αντιγραφής αυτών σε τυχαίο χρόνο. Με την υλοποίηση αυτή ο μελετητής μπορεί να καθορίσει το ποσοστό αποτυχίας και απώλειας των δεδομένων.

## **6.2. Επέκταση**

Η αναβάθμιση των εργαλείων είναι βασική προϋπόθεση, ώστε ο μελετητής να μπορέσει μέσω αυτών, να εξάγει συμπεράσματα κάνοντας χρήση των τεχνικών ανοχής σε σφάλματα.



## Βιβλιογραφία

- [1] Πληροφορική. [Online]. <http://el.wikipedia.org/wiki/%CE%A0%CE%BB%CE%B7%CF%81%CE%BF%CF%86%CE%BF%CF%81%CE%B9%CE%BA%CE%AE>
- [2] Armando Fox, Rean Griffith, Anthony D. Joseph, Randy Katz, Andy Konwinski, Gunho Lee, David Patterson, Ariel Rabkin, Ion Stoica, Matei Zaharia, Michael Armbrust, "A View of Cloud Computing," *communications of the ac m*, vol. vol. 53, April 2010.
- [3] Google Company. Google Company. [Online]. <http://www.google.com/about/>
- [4] Yahoo Company. Yahoo Company. [Online]. <https://info.yahoo.com/company>
- [5] Amazon Company. Amazon Company. [Online]. <http://en.wikipedia.org/wiki/Amazon.com>
- [6] "Cloud Computing: An Overview, A summary of important cloud-computing issues distilled from ACM CTO Roundtables," *DISTRIBUTED COMPUTING*.
- [7] Dimitrios Katsaros, Pankaj Mehra, Athena Vakali, Marios D. Dikaiakos, "Cloud Computing, Distributed Internet Computing for IT and Scientific Research," *IEEE*, 2009.
- [8] Inderveer Chana, Anju Bala, "Fault Tolerance- Challenges, Techniques and Implementation in Cloud Computing," *Computer Science and Engineering Department, Thapar University*, January 2012.
- [9] Armando Fox, Rean Griffith, Anthony D. Joseph, Randy H. Katz, Andrew Konwinski, Gunho Lee, David A. Patterson, Ariel Rabkin, Ion Stoica, Matei Zaharia, Michael Armbrust, "Above the Clouds: A Berkeley View of Cloud Computing," *Electrical Engineering and Computer Sciences, University of California at Berkeley*, 2009.
- [10] Dr. Ajay Shanker Singh, Sweta Patel, "Fault Tolerance Mechanisms and its Implementation in Cloud Computing – A Review," *International Journal of Advanced Research in Computer Science and Software Engineering*, 2013.
- [11] Zhiguo Luo, Yujian Du, and Leitao Guo Ling Qian, "Cloud Computing: An Overview".
- [12] Syed (Shawon) M. Rahman, Anthony Bisong, "AN OVERVIEW OF THE SECURITY CONCERNS IN ENTERPRISE CLOUD COMPUTING," *International Journal of Network Security & Its Applications (IJNSA)*, no. 3, January 2011.
- [13] Richard Mosher, "Cloud Computing Risks," *ISSA*, July 2011.
- [14] Yumnam Langlen Devi, Dr. Lakshmi Prasad Saikia, "FAULT TOLERANCE TECHNIQUES AND ALGORITHMS IN CLOUD," *International Journal of Computer Science & Communication Networks COMPUTING*.



- [15] Mohammad Jahangir Alam Golam Moktader Nayeem, "Analysis of Different Software Fault Tolerance Techniques," 2006.
- [16] Alexandru Costan, Valentin Cristea Elvin Sindrilaru, "Fault Tolerance and Recovery in Grid Workflow Management Systems," *International Conference on Complex, Intelligent and Software Intensive Systems*, 2010.
- [17] Stephen M. Rumble, Ryan Stutsman Sachin Katti, John Ousterhout and Mendel Rosenblum Asaf Cidon, "Copssets: Reducing the Frequency of Data Loss in Cloud Storage ," *Stanford University*.
- [18] Stephen M. Rumble, Ryan Stutsman, John Ousterhout, and Mendel Rosenblum Diego Ongaro, "Fast Crash Recovery in RAMCloud," *Stanford University*.
- [19] Parag Agrawal, David Erickson, Christos Kozyrakis, Jacob Leverich, David Mazières, Subhasish Mitra, Aravind Narayanan, Mendel Rosenblum, Stephen M. Rumble, Eric Stratmann, and Ryan Stutsman John K. Ousterhout, "The Case for RAMClouds: Scalable High-Performance Storage Entirely in DRAM," *Department of Computer Science, Stanford University*.
- [20] Pooja S. Honnutagi, "The Hadoop distributed file system," *International Journal of Computer Science and Information Technologies*, 2014.
- [21] Chien-Hung Chen, and J. Morris Chang, Senior Member, IEEE Jenn-Wei Lin, "QoS-Aware Data Replication for Data-Intensive Applications in Cloud Computing Systems," *IEEE TRANSACTIONS ON CLOUD COMPUTING*, 2013.
- [22] Thomas Kiencke, "Hadoop Distributed File System (HDFS)," *Institute of Telematics, University of L'ubeck, Germany*.
- [23] CloudSim FAQ. CloudSim: A Framework For Modeling And Simulation Of Cloud Computing Infrastructures And Services. [Online]. <https://code.google.com/p/cloudsim/wiki/FAQ>
- [24] Rajiv Ranjan, César A. F. De Rose, and Rajkumar Buyya Rodrigo N. Calheiros, "CloudSim: A Novel Framework for Modeling and Simulation of Cloud Computing Infrastructures and Services," *Grid Computing and Distributed Systems (GRIDS) Laboratory, Department of Computer Science and Software Engineering, The University of Melbourne, Australia*.
- [25] G. Sahoo Ranjan Kumar, "Cloud Computing Simulation Using CloudSim," *International Journal of Engineering Trends and Technology (IJETT)*, no. 8, Feb 2014.
- [26] Rajiv Ranjan, Anton Beloglazov, César A. F. De Rose and Rajkumar Buyya Rodrigo N. Calheiros, "CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms," *Wiley Online Library*, 2010.





- [27] Steve Versteeg and Rajkumar Buyya Saurabh Kumar Garg, "SMICloud: A Framework for Comparing and Ranking Cloud Services," *Fourth IEEE International Conference on Utility and Cloud Computing*, 2011.
- [28] G.Sahoo Ranjan Kumar, "Cloud Computing Simulation Using CloudSim," *International Journal of Engineering*, 2014.
- [29] CloudSim: A Framework For Modeling And Simulation Of Cloud Computing Infrastructures And Services. CloudSim 3.0.3. [Online]. <https://code.google.com/p/cloudsim/downloads/list>
- [30] Oracle Corporation and/or its affiliate. (2015) NetBeans IDE. [Online]. <https://netbeans.org/>
- [31] Michael Tighe, Hanan Lutfiyya and Michael Bauer Gaston Keller, "DCSim: A Data Centre Simulation Tool".
- [32] Joan Soriano Soteras, "Simulation of Hierarchical Data Centre Resource Allocation Strategies using DCSim".
- [33] Supervised by H. Moens and F. De Turck Joan Soriano Soteras, "Simulation of Hierarchical Data Centre Resource Allocation Strategies using DCSim".
- [34] DCSim. DCSim-GitHub. [Online]. <https://github.com/digs-uwo/dcsim>
- [35] Pangfeng Liu , Jan-Jan Wu Hsiangkai Wang, "A QoS-Aware Heuristic Algorithm for Replica Placement".
- [36] Ruchuan Wang,Yang Wang,Song Deng Xiong Fu, "A Replica Placement Algorithm in Mobile Grid Environments," *International Conferences on Embedded Software and Systems*, 2009.
- [37] Armando Fox, Rean Griffith,Anthony D. Joseph, Randy Katz, Andy Konwinski,Gunho Lee, David Patterson, Ariel Rabkin, Ion Stoica,Matei Zaharia Michael Armbrust, "Clearing the clouds away from the true potential and obstacles posed by this computing capability".
- [38] Xiaofeng Meng, Jing Zhao, Xiangmei Hu, Bingbing Liu and Haiping Wang Yingjie Shi, "Benchmarking Cloud-based Data Management Systems".
- [39] "Cloud Service Assurance for Virtualized Multiservice Data Center," *Cisco*.
- [40] Francois Labelle, Florentina I. Popovici, Murray Stokely, Van-Anh Truong,Luiz Barroso, Carrie Grimes, and Sean Quinlan Daniel Ford, "Availability in Globally Distributed Storage Systems".
- [41] Adam Silberstein, Erwin Tam, Raghu Ramakrishnan, Russell Sears Brian F. Cooper,



"Benchmarking Cloud Serving Systems with YCSB," *Yahoo*.

- [42] George Pal 1 i s, "Cloud Computing, The New Frontier of Internet Computing," *University of Cyprus*.
- [43] Doug Jewett Timothy K. Tsai and Ravishankar K. Iyer, "An Approach towards Benchmarking of Fault-Tolerant Commercial Systems," 1996.
- [44] Yuchen Jia , Renan Jia Fanrong Xie, "Algorithm for Minimum Cost Maximum Flow in Transportation Network," April 2012.
- [45] Anirban Mandal, Charles Koelbel and Keith Cooper Yang Zhang, "Combined Fault Tolerance and Scheduling Techniques for Workflow Applications on Computational Grids".
- [46] ROB E R T J. CHANS L E R, "Data Availability and Durability with the Hadoop Distributed File System".
- [47] Scott Brock, Ronald Skeoch, Shrisha Rao Sumit Kumar Bose, "CloudSpider: Combining Replication with Scheduling for Optimizing Live Migration of Virtual Machines Across Wide Area Networks".
- [48] "Fault tolerance in Hadoop MapReduce implementation," 2013.
- [49] P. M. Melliar-Smith and L. E. Moser Wenbing Zhao, "Fault Tolerance Middleware for Cloud Computing," *IEEE 3rd International Conference on Cloud Computing*, 2010.
- [50] Thanh Do, Joseph M. Hellerstein, Ion Stoica, Dhruba Borthakur, Jesse Robbins Haryadi S. Gunawi, "Failure as a Service (FaaS): A Cloud Service for Large-," *Electrical Engineering and Computer Sciences, University of California at Berkeley*, July 28, 2011.
- [51] Laurent Broto and Daniel Hagimont Alain Tchana, "Review of Approaches to Cloud Computing Fault Tolerance".
- [52] Howard Gobioff, and Shun-Tak Leung Sanjay Ghemawat, "The Google File System, Distributed and Cloud Computing".
- [53] Howard Gobioff, and Shun-Tak Leung Sanjay Ghemawat, "The Google File System," *Google*.
- [54] Brad Calder and et, "Windows Azure Storage: A Highly Available Cloud Storage Service with Strong Consistency".
- [55] Sugandha Banga Princy Gupta, "Topic - Review of Cloud Computing in Fault Tolerant Environment With Efficient Energy Consumption," *International Journal of scientific research and management (IJSRM)*, 2013.



- [56] Prof. R. K. Kapoor Bhupendra Panchal, "Dynamic VM Allocation Algorithm using Clustering in Cloud Computing," *International Journal of Advanced Research in Computer Science and Software Engineering*, 2013.
- [57] Jie Qiu, Jie Yang Feng Wang, "Hadoop High Availability through Metadata Replication".
- [58] Ryan Stutsman, Stephen Rumble, Sachin Katti, John Ousterhout and Mendel Rosenblum Asaf Cidon, "MinCopssets: Derandomizing Replication in Cloud Storage".
- [59] Ryan Stutsman, Stephen Rumble, Sachin Katti, John Ousterhout and Mendel Rosenblum Asaf Cidon, "MinCopssets: Derandomizing Replication In Cloud Storage".
- [60] Mohamed-H MOUSA Mohamed-K HUSSEIN, "A Light-weight Data Replication for Cloud Data Centers Environment," *International Journal of Innovative Research in Computer and Communication Engineering*, 2014.
- [61] Joydeep Sen Sarma, Jonathan Gray Dhruba Borthakur, "Apache Hadoop Goes Realtime at Facebook".
- [62] Djamal Zeghlache Makhoul Hadji, "Minimum Cost Maximum Flow Algorithm for Dynamic Resource Allocation in Clouds".
- [63] Rajiv Ranjan and Rodrigo N. Calheiros Rajkumar Buyya, "Modeling and Simulation of Scalable Cloud Computing Environments and the CloudSim Toolkit: Challenges and Opportunities," *Grid Computing and Distributed Systems (GRIDS) Laboratory*.
- [64] Saurabh Kumar Garg, and Rodrigo N. Calheiros Rajkumar Buyya, "SLA-Oriented Resource Provisioning for Cloud Computing: Challenges, Architecture, and Solutions," *International Conference on Cloud and Service Computing*, 2011.
- [65] Department of Computing Jia Ru, "An Investigation on Scheduling Policies for Cloud-based Software Systems," *The Hong Kong Polytechnic University*.
- [66] "Message from the UCC 2011 General Co-Chairs," *IEEE/ACM International Conference*, 2011.
- [67] Feiyan Shi , Wei Du and Hongfeng Li Wei Liu, "A Cost-Aware Resource Selection for Dataintensive Applications in Cloud-oriented DataCenters," *I.J. Information Technology and Computer Science*, 2011.
- [68] Shangguang Wang, Qibo Sun, Hua Zou, Fangchun Yang Ao Zhou, "FTCloudSim: A Simulation Tool for Cloud Service Reliability Enhancement Mechanisms," *State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing, China*.
- [69] Australia The University of Melbourne, "Cloud Computing and Distributed Systems Laboratory and the Cloudbus Project," *Department of Computing and Information*



Κωνσταντίνος Αθανασίου , 'Ανοχή σε σφάλματα σε υπολογιστικό περιβάλλον νέφους'

*Systems, Melbourne School of Engineering*, 2011.

- [70] Ajit Singh, Aakanksha Agrawal Tarun Goyal, "Clousim: simulator for cloud computing infrastructure and modeling," *SciVerse ScienceDirect*, 2012.
- [71] Vikash, "Dynamic Creation and Placement of Virtual Machine Using CloudSim," *International Journal of Emerging Technology and Advanced Engineering*, vol. 8, no. 4, August 2014.