



ΠΑΝΕΠΙΣΤΗΜΙΟ
ΘΕΣΣΑΛΙΑΣ

ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

Ευφυής Διαχείριση Κατανεμημένων Δεδομένων

GIORGI JANELIDZE

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

ΥΠΕΥΘΥΝΟΣ

.....ΚΩΝΣΤΑΝΤΙΝΟΣ ΚΟΛΟΜΒΑΤΣΟΣ.....
.....Βαθμίδα.....

ΣΥΝΕΠΙΒΛΕΠΩΝ
(εφόσον υπάρχει)

.....Όνοματεπώνυμο Υπεύθυνου.....
.....Βαθμίδα.....

Λαμβάνει έτος



ΠΑΝΕΠΙΣΤΗΜΙΟ
ΘΕΣΣΑΛΙΑΣ

ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

Ευφυής Διαχείριση Κατανεμημένων Δεδομένων

GIORGI JANELIDZE



UNIVERSITY OF
THESSALY

SCHOOL OF SCIENCE

DEPARTMENT OF COMPUTER SCIENCE & TELECOMMUNICATIONS

..... THESIS TITLE.....

..... STUDENT'S FULL NAME.....

ΥΠΕΥΘΥΝΟΣ

.....Ονοματεπώνυμο Υπεύθυνου.....
.....Βαθμίδα.....

ΣΥΝΕΠΙΒΛΕΠΩΝ
(εφόσον υπάρχει)

FINAL THESIS

ADVISOR

.....Full Name.....
.....Position.....

CO ADVISOR

.....Full Name.....
.....Position.....

Lamia year

«Με ατομική μου ευθύνη και γνωρίζοντας τις κυρώσεις ⁽¹⁾, που προβλέπονται από της διατάξεις της παρ. 6 του άρθρου 22 του Ν. 1599/1986, δηλώνω ότι:

1. Δεν παραθέτω κομμάτια βιβλίων ή άρθρων ή εργασιών άλλων αυτολεξεί **χωρίς να τα περικλείω σε εισαγωγικά** και χωρίς να αναφέρω το συγγραφέα, τη χρονολογία, τη σελίδα. Η αυτολεξεί παράθεση χωρίς εισαγωγικά χωρίς αναφορά στην πηγή, είναι λογοκλοπή. Πέραν της αυτολεξεί παράθεσης, λογοκλοπή θεωρείται και η παράφραση εδαφίων από έργα άλλων, συμπεριλαμβανομένων και έργων συμφοιτητών μου, καθώς και η παράθεση στοιχείων που άλλοι συνέλεξαν ή επεξεργάστηκαν, χωρίς αναφορά στην πηγή. Αναφέρω πάντοτε με πληρότητα την πηγή κάτω από τον πίνακα ή σχέδιο, όπως στα παραθέματα.
2. Δέχομαι ότι η αυτολεξεί **παράθεση χωρίς εισαγωγικά**, ακόμα κι αν συνοδεύεται από αναφορά στην πηγή σε κάποιο άλλο σημείο του κειμένου ή στο τέλος του, είναι αντιγραφή. Η αναφορά στην πηγή στο τέλος π.χ. μιας παραγράφου ή μιας σελίδας, δεν δικαιολογεί συρραφή εδαφίων έργου άλλου συγγραφέα, έστω και παραφρασμένων, και παρουσίασή τους ως δική μου εργασία.
3. Δέχομαι ότι υπάρχει επίσης περιορισμός στο μέγεθος και στη συχνότητα των παραθεμάτων που μπορώ να εντάξω στην εργασία μου εντός εισαγωγικών. Κάθε μεγάλο παράθεμα (π.χ. σε πίνακα ή πλαίσιο, κλπ), προϋποθέτει ειδικές ρυθμίσεις, και όταν δημοσιεύεται προϋποθέτει την άδεια του συγγραφέα ή του εκδότη. Το ίδιο και οι πίνακες και τα σχέδια
4. Δέχομαι όλες τις συνέπειες σε περίπτωση λογοκλοπής ή αντιγραφής.

Ημερομηνία:/...../20.....

Ο – Η Δηλ.

(1) «Όποιος εν γνώσει του δηλώνει ψευδή γεγονότα ή αρνείται ή αποκρύπτει τα αληθινά με έγγραφη υπεύθυνη δήλωση του άρθρου 8 παρ. 4 Ν. 1599/1986 τιμωρείται με φυλάκιση τουλάχιστον τριών μηνών. Εάν ο υπαίτιος αυτών των πράξεων σκόπευε να προσπορίσει στον εαυτόν του ή σε άλλον περιουσιακό όφελος βλάπτοντας τρίτον ή σκόπευε να βλάψει άλλον, τιμωρείται με κάθειρξη μέχρι 10 ετών.»

ΠΕΡΙΛΗΨΗ

Στη σύγχρονη εποχή, η ανάγκη για διαχείριση και αποθήκευση μεγάλου όγκου δεδομένων και πληροφοριών αυξάνεται συνεχώς. Η συνεχής αύξηση του όγκου των πληροφοριών δημιουργεί την ανάγκη για νέες λύσεις που να είναι ταυτόχρονα αποδοτικές, επεκτάσιμες και αξιόπιστες. Οι παραδοσιακές μέθοδοι αποθήκευσης, που βασίζονται σε μεμονωμένους κεντρικούς κόμβους ή υπολογιστές, αντιμετωπίζουν σημαντικούς περιορισμούς όσον αφορά την απόδοση τους, τη διαθεσιμότητα των αποθηκευμένων δεδομένων και τη δυνατότητα κλιμάκωσης, καθώς δεν μπορούν να ανταποκριθούν στις αυξανόμενες ανάγκες αποθήκευσης πληροφοριών.

Η καταναμεμημένη αποθήκευση δεδομένων έχει αναδειχθεί ως μια σύγχρονη και αποτελεσματική λύση στο ζήτημα της αποθήκευσης δεδομένων. Ένα σύστημα καταναμεμημένης αποθήκευσης αποτελείται από πολλαπλούς διασυνδεδεμένους υπολογιστές, δηλαδή πρόκειται για ένα δίκτυο υπολογιστών, όπου τα δεδομένα και οι πληροφορίες αποθηκεύονται και διαχειρίζονται από πολλαπλούς αυτόνομους κόμβους. Αυτή η προσέγγιση επιτρέπει την καλύτερη αξιοποίηση των διαθέσιμων υπολογιστικών πόρων, την εξισορρόπηση του φορτίου και την αύξηση της διαθεσιμότητας των αποθηκευμένων δεδομένων.

Η παρούσα διπλωματική εργασία εξετάζει τις βασικές μεθόδους και τεχνικές καταναμεμημένης αποθήκευσης δεδομένων σε δίκτυα υπολογιστών, αναλύοντας τα πλεονεκτήματα και τους περιορισμούς κάθε προσέγγισης. Αρχικά, παρουσιάζονται οι κύριες και πιο βασικές τεχνικές καταναμεμημένης αποθήκευσης, εστιάζοντας στα χαρακτηριστικά και τις λειτουργίες τους. Στη συνέχεια, υλοποιείται μια πρακτική εφαρμογή που περιλαμβάνει το σχεδιασμό και την ανάπτυξη ενός συστήματος για την καταναμεμημένη αποθήκευση δεδομένων. Στόχος είναι η παρουσίαση μιας ολοκληρωμένης προσέγγισης για τη δημιουργία αποδοτικών και αξιόπιστων συστημάτων αποθήκευσης, ικανών να ανταποκριθούν σε αυξανόμενες απαιτήσεις.

ABSTRACT

The increasing need for efficient data management and storage has become more evident in recent years. As data volumes continue to grow, new solutions are required that are both efficient, scalable, and reliable. Traditional storage methods, which rely on centralized nodes or individual computers, face significant limitations in terms of performance, data availability, and scalability, making it difficult to meet the rising storage demands.

Distributed data storage has emerged as a modern and effective solution to these challenges. A distributed storage system consists of multiple interconnected computers, forming a network where data and information are stored and managed across multiple autonomous nodes. This approach enables better utilization of available computational resources, load balancing, and increased data availability.

This thesis explores the fundamental methods and techniques of distributed data storage in computer networks, analyzing the advantages and limitations of each approach. Initially, the core techniques of distributed storage are presented, focusing on their characteristics and functionalities. Subsequently, a practical implementation is developed, involving the design and deployment of a distributed storage system. The aim is to provide a comprehensive approach to the creation of efficient and reliable storage systems capable of meeting growing demands.

ΠΕΡΙΕΧΟΜΕΝΑ

ΠΕΡΙΛΗΨΗ	1
ABSTRACT	3
ΠΕΡΙΕΧΟΜΕΝΑ	4
ΚΕΦΑΛΑΙΟ 1 Εισαγωγή	6
1.1 Σκοπός και στόχοι της εργασίας	6
1.2 Περιορισμοί και προβλήματα στην αποθήκευση δεδομένων σε ένα κεντρικό κόμβο	6
1.3 Εισαγωγή στην κατανομημένη αποθήκευση δεδομένων	8
1.4 Προκλήσεις και δυσκολίες στην κατανομημένη αποθήκευση δεδομένων	10
ΚΕΦΑΛΑΙΟ 2 Αποθήκευση δεδομένων βάση κλειδιού-τιμής	10
2.1 Πλεονεκτήματα στην αποθήκευση δεδομένων βάση κλειδιού-τιμής	11
ΚΕΦΑΛΑΙΟ 3 Βασικές τεχνικές κατανομημένης αποθήκευσης δεδομένων	12
3.1 Κυκλική κατανομή (Round Robin)	13
3.1.α Πλεονεκτήματα της κυκλικής κατανομής	17
3.1.β Μειονεκτήματα της κυκλικής κατανομής	17
3.2 Στατική Κατανομή (Static Partitioning)	17
3.2.α Πλεονεκτήματα της στατικής κατανομής	22
3.2.β Μειονεκτήματα της στατικής κατανομής	22
3.3 Αντιγραφή δεδομένων (Replication)	23
3.3.α Πλήρης αντιγραφή δεδομένων	24
3.3.β Μερική αντιγραφή δεδομένων	24
3.3.γ Πλεονεκτήματα της αντιγραφής δεδομένων	26
3.3.δ Μειονεκτήματα της αντιγραφής δεδομένων	27
3.4 Κατανομή με υπόλοιπο της τιμής κατακερματισμού (Hash Modulo)	28
3.4.α Πλεονεκτήματα της μεθόδου υπόλοιπου κατακερματισμού	32
3.4.β Μειονεκτήματα της μεθόδου υπόλοιπου κατακερματισμού	32
3.5 Δυναμικός κατακερματισμός (Dynamic Hashing)	33
3.5.α Πλεονεκτήματα του δυναμικού κατακερματισμού	34
3.5.β Μειονεκτήματα του δυναμικού κατακερματισμού	34
ΚΕΦΑΛΑΙΟ 4 Συναρτήσεις κατακερματισμού	34
4.1 Εισαγωγή	35
4.2 Ορισμός και βασικές αρχές του κατακερματισμού	35
4.3 Χρήσεις της συνάρτησης κατακερματισμού	36
4.3.α Κρυπτογραφία και ασφάλεια δεδομένων	37
4.3.β Αποθήκευση και αναζήτηση δεδομένων	38
4.4 Ιδιότητες της συνάρτησης κατακερματισμού	39
4.4.α Ντετερμινιστικότητα (Determinism)	39
4.4.β Αντίσταση σε συγκρούσεις (Collision Resistance)	39
4.4.γ Αντίσταση σε προεικόνα (Preimage Resistance)	40
4.4.δ Ομοιομορφία στην κατανομή (Uniform Distribution)	40
4.4.ε Απόδοση (Efficiency)	40

ΚΕΦΑΛΑΙΟ 5 Σχεδιασμός του συστήματος αποθήκευσης	42
5.1 Προδιαγραφές και ζητούμενα του συστήματος	42
5.2 Επιλογή της κατάλληλης μεθόδου για την κατανεμημένη αποθήκευση.....	44
5.3 Η μέθοδος του συνεπή κατακερματισμού (Consistent Hashing).....	45
5.4 Επέκταση του συνεπή κατακερματισμού για τη διατήρηση αντιγράφων.....	59
ΚΕΦΑΛΑΙΟ 6 Υλοποίηση του συστήματος αποθήκευσης	62
6.1 Επιλογή της γλώσσας προγραμματισμού	63
6.2 Βιβλιοθήκες και εργαλεία που χρησιμοποιούνται	63
6.3 Δομή και λειτουργία του συστήματος	65
6.4 Δεδομένα και μεταβλητές που διατηρεί κάθε κόμβος.....	66
6.5 Λειτουργίες κάθε κόμβου	67
ΚΕΦΑΛΑΙΟ 7 Βελτιώσεις και μελλοντικές προσθήκες	76
7.1 Συνεπής κατακερματισμός με τη χρήση εικονικών κόμβων	76
7.2 Πώς λειτουργούν οι εικονικοί κόμβοι.....	77
7.3 Πλεονεκτήματα της χρήσης εικονικών κόμβων	80
ΣΥΜΠΕΡΑΣΜΑΤΑ	81
ΒΙΒΛΙΟΓΡΑΦΙΑ	84

ΚΕΦΑΛΑΙΟ 1 Εισαγωγή

1.1 Σκοπός και στόχοι της εργασίας

Η παρούσα διπλωματική εργασία εστιάζει στη διερεύνηση μεθόδων και τεχνικών που χρησιμοποιούνται στην συνεργατική αποθήκευση δεδομένων στις παρυφές ενός δικτύου, αξιοποιώντας πολλαπλούς αυτόνομους κόμβους. Ο σκοπός της εργασίας είναι η ανάπτυξη ενός συστήματος που θα επιτρέπει την αποθήκευση δεδομένων στους κόμβους του, κατανέμοντας τα αντικείμενα που χρειάζεται να αποθηκευτούν στους επιμέρους κόμβους του συστήματος με τέτοιο τρόπο, ώστε να επιτευχθεί η αποδοτικότερη αξιοποίηση και η αποτελεσματικότερη διαχείριση του αποθηκευτικού χώρου των κόμβων του δικτύου.

Στο πλαίσιο αυτό, ένα από τα βασικότερα ζητούμενα στην ανάπτυξη του συστήματος είναι η διασφάλιση της αποτελεσματικής διαχείρισης στην αποθήκευση των δεδομένων, ώστε να αξιοποιούνται με τον καλύτερο δυνατό τρόπο οι διαθέσιμοι πόροι στους κόμβους του δικτύου. Παράλληλα, το σύστημα θα πρέπει να εξασφαλίζει την ορθή και αποδοτική κατανομή των αντικειμένων στους ξεχωριστούς αυτόνομους κόμβους του, με τέτοιο τρόπο, ώστε να αποφεύγεται η άνιση κατανομή των αντικειμένων και να εξασφαλίζεται η βέλτιστη χρήση του αποθηκευτικού χώρου, καθιστώντας το σύστημα πιο αποτελεσματικό και αξιόπιστο στη λειτουργία του.

Εξίσου σημαντικός στόχος της παρούσας υλοποίησης είναι η αποδοτική λειτουργία του συστήματος σε ό,τι αφορά τις διαδικασίες εγγραφής και ανάγνωσης δεδομένων στο δίκτυο. Επιπλέον, είναι απαραίτητο να επιτευχθεί η εξισορρόπηση του φορτίου μεταξύ των ξεχωριστών κόμβων, προκειμένου να αποφεύγεται η υπερφόρτωση μεμονωμένων κόμβων και να εξασφαλίζεται η ομαλή λειτουργία του συστήματος. Ένας ακόμα σημαντικός στόχος είναι η αύξηση της διαθεσιμότητας των δεδομένων που αποθηκεύονται, εξασφαλίζοντας ότι τα δεδομένα θα είναι πάντοτε διαθέσιμα και προσβάσιμα, ανεξάρτητα από ενδεχόμενα σφάλματα ή δυσλειτουργίες σε μεμονωμένους κόμβους του δικτύου. Τέλος, το σύστημα θα πρέπει να είναι ικανό να κλιμακώνεται και να μεταβάλλεται, επιτρέποντας με τον τρόπο αυτό την αύξηση του αποθηκευτικού χώρου και του πόρους του συστήματος με στόχο τη δυνατότητα υποστήριξης μεγαλύτερου όγκων δεδομένων.

Στην εργασία δίνεται έμφαση στην ικανότητα του συστήματος να προσαρμόζεται σε δυναμικά περιβάλλοντα, όπου μπορεί να υπάρξουν αλλαγές στη σύνθεση και τη τοπολογία του δικτύου, όπως για παράδειγμα η προσθήκη ή η αφαίρεση κόμβων στο δίκτυο. Σε αυτές τις περιπτώσεις, το σύστημα θα πρέπει να έχει μηχανισμούς που να του επιτρέπουν να διαχειρίζεται αυτές τις αλλαγές χωρίς να επηρεάζεται η ομαλή λειτουργία του, η απόδοση όπως επίσης και η διαθεσιμότητα των αποθηκευμένων αντικειμένων. Με τον τρόπο αυτό διασφαλίζεται η λειτουργία του, ανεξάρτητα από αλλαγές και μεταβολές που μπορεί να προκύψουν στη σύνθεση του.

1.2 Περιορισμοί και προβλήματα στην αποθήκευση δεδομένων σε ένα κεντρικό κόμβο

Το μοντέλο της αποθήκευσης δεδομένων σε ένα μεμονωμένο κεντρικό κόμβο αποτελεί μια σχετικά απλή λύση και είναι η προσέγγιση που συχνά χρησιμοποιείται για να καλύψει βασικές ανάγκες στο ζήτημα της αποθήκευσης δεδομένων. Στην περίπτωση αυτή, ένας μοναδικός κόμβος είναι υπεύθυνος για την αποθήκευση των δεδομένων στο σύστημα, γεγονός που καθιστά τον σχεδιασμό του συστήματος και τη διαχείριση των δεδομένων πιο απλή και εύκολη διαδικασία. Αυτή η προσέγγιση είναι κατάλληλη και χρησιμοποιείται συχνά για την ανάπτυξη συστημάτων με πιο περιορισμένες προδιαγραφές και ανάγκες. Ωστόσο, καθώς οι απαιτήσεις που σχετίζονται με την αποθήκευση δεδομένων αυξάνονται, παρουσιάζονται σημαντικά προβλήματα καθώς και περιορισμοί με τη χρήση του συγκεκριμένου μοντέλου αποθήκευσης

Προκειμένου να αντιμετωπιστούν οι περιορισμοί και τα προβλήματα που αναφέρονται παραπάνω, έχει προκύψει η ανάγκη για την ανάπτυξη πιο πολύπλοκων και σύνθετων συστημάτων αποθήκευσης. Παρακάτω παρουσιάζονται μερικοί από τους πιο βασικούς περιορισμούς που προκύπτουν με τη χρήση του μοντέλου αποθήκευσης δεδομένων σε ένα μεμονωμένο κεντρικό κόμβο.

Ένας από τους πιο βασικούς περιορισμούς που υπάρχει είναι ο κίνδυνος διακοπής λειτουργίας του συστήματος. Σε μία υλοποίηση όπου όλα τα δεδομένα συγκεντρώνονται και αποθηκεύονται σε έναν μοναδικό κόμβο, οποιαδήποτε βλάβη ή αποτυχία του συγκεκριμένου κόμβου οδηγεί στη διακοπή λειτουργίας όλου του συστήματος. Η αποτυχία αυτή ενδέχεται να προκύψει από διάφορες αιτίες, όπως για παράδειγμα κάποια τεχνική βλάβη ή πιθανή υπερφόρτωση του κόμβου από αυξημένο όγκο αιτημάτων για προσπέλαση των δεδομένων του ή ακόμα και κάποια βλάβη στο δίκτυο το οποίο συνδέει τους χρήστες και τις εφαρμογές που χρησιμοποιούν το σύστημα αποθήκευσης με τον κόμβο.

Εκτός από τη διακοπή λειτουργίας του συστήματος, ακόμα ένα βασικό πρόβλημα που προκύπτει, είναι ο κίνδυνος της απώλειας των δεδομένων σε περίπτωση κάποιας βλάβης ή αποτυχίας του κόμβου. Εάν δεν υπάρχουν επαρκείς μηχανισμοί ασφαλείας, όπως για παράδειγμα η διατήρηση αντιγράφων ασφαλείας των δεδομένων, η βλάβη αυτή μπορεί να οδηγήσει στην οριστική απώλεια των αποθηκευμένων δεδομένων στο σύστημα. Αυτός ο κίνδυνος είναι ιδιαίτερα σοβαρός όταν η αποτυχία του κόμβου είναι μόνιμη και μη αναστρέψιμη. Το ενδεχόμενο αυτό καθιστά αδύνατη την ανάκτηση των αποθηκευμένων δεδομένων και έτσι είναι πιθανή η οριστική απώλεια των δεδομένων που είναι αποθηκευμένα στο σύστημα.

Η απόδοση και η αποδοτικότητα ενός συστήματος που λειτουργεί με ένα μοναδικό κεντρικό κόμβο αποτελεί ακόμα ένα σημαντικό περιορισμό στη χρήση του συγκεκριμένου μοντέλου. Καθώς αυξάνεται ο όγκος και η ποσότητα των δεδομένων που χρειάζεται να αποθηκευτούν, όπως επίσης και ο αριθμός των αιτημάτων προσπέλασης σε αυτά, για παράδειγμα αιτήματα εγγραφής και ανάγνωσης δεδομένων, ο κεντρικός κόμβος καλείται να εξυπηρετήσει και να διαχειριστεί έναν συνεχώς αυξανόμενο φόρτο εργασίας. Το αυξημένο φορτίο σε ένα κόμβο μπορεί να οδηγήσει σε υπερφόρτωση του κόμβου, μειώνοντας με τον τρόπο αυτό την συνολική απόδοση του συστήματος.

Σε περίπτωση υπερφόρτωσης του κόμβου, είναι πιθανό να παρουσιαστούν καθυστερήσεις στις λειτουργίες προσπέλασης δεδομένων. Σε περιβάλλοντα με υψηλό αριθμό ταυτόχρονων αιτημάτων ή όταν απαιτείται η διαχείριση μεγάλου όγκου δεδομένων, η απόδοση

του συστήματος μπορεί να μειωθεί σημαντικά. Τέτοιες συνθήκες επηρεάζουν αρνητικά τόσο τη συνολική λειτουργικότητα όσο και την αποδοτικότητα του συστήματος.

Σε συστήματα αποθήκευσης δεδομένων, συχνά προκύπτει η ανάγκη για αύξηση του συνολικού αποθηκευτικού χώρου, καθώς και για αύξηση του αριθμού των αιτημάτων που μπορεί να διαχειριστεί το σύστημα. Η διαδικασία αυτή, η οποία αναφέρεται ως κλιμάκωση του συστήματος, προκύπτει από την ανάγκη αποθήκευσης μεγαλύτερου όγκου δεδομένων και εξυπηρέτησης περισσότερων χρηστών και είναι ιδιαίτερα απαιτητική και χρονοβόρα, ειδικά σε συστήματα που βασίζονται σε έναν μοναδικό κεντρικό κόμβο για τη λειτουργία τους. Ο λόγος που η διαδικασία αυτή είναι απαιτητική, είναι ότι η αύξηση του αποθηκευτικού χώρου προϋποθέτει την αντικατάσταση ή την αναβάθμιση του φυσικού εξοπλισμού ενός κόμβου. Επιπλέον, κατά τη διαδικασία της αντικατάστασης του εξοπλισμού ενός κόμβου στις περισσότερες περιπτώσεις απαιτείται η διακοπή της λειτουργίας του συστήματος αποθήκευσης. Παράλληλα, στο συγκεκριμένο μοντέλο δεν υπάρχει η δυνατότητα της απεριόριστης κλιμάκωσης της υπολογιστικής ισχύος καθώς και του διαθέσιμου αποθηκευτικού χώρου που είναι διαθέσιμος στο σύστημα. Αυτό οφείλεται στο γεγονός ότι η συνολική χωρητικότητα ενός κόμβου περιορίζεται τον εξοπλισμό που μπορεί να υποστηρίξει ένας κόμβος, καθιστώντας την συγκεκριμένη υλοποίηση ακατάλληλη για περιπτώσεις που υπάρχουν αυξανόμενες ανάγκες αποθήκευσης.

Σύμφωνα με τα παραπάνω, ένα σύστημα αποθήκευσης δεδομένων που λειτουργεί με ένα μοναδικό κεντρικό κόμβο, ενώ είναι απλό στην ανάπτυξη και τη λειτουργία του λόγω των βασικών απαιτήσεων και της χαμηλής πολυπλοκότητας, παρουσιάζει αρκετούς σημαντικούς περιορισμούς και προβλήματα στη λειτουργία του. Οι περιορισμοί αυτοί επηρεάζουν την αξιοπιστία, την απόδοση καθώς και τη δυνατότητα κλιμάκωσης του συστήματος. Οι κίνδυνοι διακοπής λειτουργίας, απώλειας δεδομένων και περιορισμένης αποδοτικότητας καθιστούν αυτό το μοντέλο λιγότερο κατάλληλο για συστήματα που απαιτούν αυξημένη ευελιξία, ανθεκτικότητα και απόδοση λειτουργίας.

1.3 Εισαγωγή στην κατανεμημένη αποθήκευση δεδομένων

Η κατανεμημένη αποθήκευση δεδομένων αναφέρεται σε μια πιά σύνηθε προσέγγιση στον κλάδο της αποθήκευσης δεδομένων, βασισμένη στη συνδυαστική χρήση πολλαπλών κόμβων οι οποίοι χρησιμοποιούνται για την αποθήκευση και τη διαχείριση των δεδομένων. Στο μοντέλο αυτό, οι ξεχωριστοί κόμβοι είναι συνήθως διασυνδεδεμένοι μέσω δικτύου και κάθε κόμβος έχει τη δυνατότητα να επικοινωνεί και να συνεργάζεται με τους υπόλοιπους κόμβους στο δίκτυο. Σε αντίθεση με το παραδοσιακό μοντέλο όπου τα δεδομένα συγκεντρώνονται και αποθηκεύονται σε ένα μοναδικό κεντρικό κόμβο, τα συστήματα κατανεμημένης αποθήκευσης αξιοποιούν την υπολογιστική ισχύ όπως επίσης και τον αποθηκευτικό χώρο πολλαπλών διαφορετικών κόμβων.

Η διαχείριση και αποθήκευση δεδομένων σε πολλαπλούς κόμβους υλοποιείται μέσω διαφόρων τεχνικών και μεθόδων, καθεμία από τις οποίες έχει σχεδιαστεί για να καλύπτει συγκεκριμένες απαιτήσεις και ανάγκες. Αυτές οι τεχνικές διαφέρουν ως προς την αρχιτεκτονική, τον σχεδιασμό και τους μηχανισμούς που χρησιμοποιούν για την κατανομή και τη διαχείριση των δεδομένων. Κάθε μέθοδος προσφέρει διαφορετικά χαρακτηριστικά και

δυνατότητες, καθιστώντας την καταλληλότερη για συγκεκριμένα σενάρια χρήσης. Η επιλογή της κατάλληλης μεθόδου εξαρτάται από παράγοντες όπως ο όγκος των δεδομένων που πρέπει να αποθηκευτούν, ο αριθμός των αιτήσεων που πρέπει να εξυπηρετηθούν, καθώς και η επιθυμητή διαθεσιμότητα και αξιοπιστία του συστήματος. Κάθε προσέγγιση συνοδεύεται από τα δικά της πλεονεκτήματα και περιορισμούς. Παρακάτω, παρουσιάζονται μερικές από τις πιο βασικές μεθόδους για την αποθήκευση δεδομένων σε πολλαπλούς κόμβους.

Μία από τις πιο συνηθισμένες τεχνικές που χρησιμοποιούνται σε συστήματα είναι η πλήρης αντιγραφή των δεδομένων σε όλους τους κόμβους. Σε αυτή την προσέγγιση, κάθε κόμβος διατηρεί ένα πλήρες αντίγραφο όλων των δεδομένων. Αυτό εξασφαλίζει την υψηλή διαθεσιμότητα των δεδομένων, καθώς ακόμη και αν ένας ή περισσότεροι κόμβοι αποτύχουν, τα δεδομένα παραμένουν προσβάσιμα από τους υπόλοιπους κόμβους. Επιπλέον, αυτή η μέθοδος αυξάνει την ανθεκτικότητα του συστήματος, καθώς δεν υπάρχει κρίσιμο σημείο αποτυχίας. Ωστόσο, η αντιγραφή δεδομένων σε όλους τους κόμβους απαιτεί μεγάλο αποθηκευτικό χώρο, καθώς κάθε κόμβος πρέπει να αποθηκεύει όλα τα δεδομένα. Αυτό μπορεί να γίνει δαπανηρό και μη αποδοτικό στο θέμα του αποθηκευτικού χώρου που απαιτείται, ειδικά σε συστήματα που χρειάζεται να αποθηκεύσουν μεγάλους όγκους δεδομένων.

Σε αντίθεση με τη μέθοδο της αντιγραφής δεδομένων που περιγράφεται παραπάνω, υπάρχει επίσης η μέθοδος της κατανομής δεδομένων, η οποία βασίζεται στην ιδέα ότι κάθε κόμβος διαχειρίζεται ένα διαφορετικό υποσύνολο των δεδομένων. Αυτή η προσέγγιση μειώνει σημαντικά την ανάγκη για αποθηκευτικό χώρο, καθώς κάθε κόμβος αποθηκεύει μόνο ένα μέρος των δεδομένων. Επιπλέον, η κατανομή δεδομένων μπορεί να αυξήσει την απόδοση του συστήματος, καθώς τα ερωτήματα μπορούν να κατανεμηθούν σε διαφορετικούς κόμβους, μειώνοντας τον χρόνο απόκρισης και αυξάνοντας το συνολικό αριθμό των αιτήσεων που μπορεί να εξυπηρετήσει το σύστημα. Ωστόσο, αυτή η μέθοδος εισάγει τον κίνδυνο της απώλειας δεδομένων εάν ένας κόμβος αποτύχει, καθώς τα δεδομένα που διαχειρίζεται δεν είναι διαθέσιμα σε άλλους κόμβους.

Μια πιο προηγμένη και σύνθετη προσέγγιση στο ζήτημα της αποθήκευσης δεδομένων σε πολλαπλούς κόμβους είναι η ανάπτυξη μεθόδων που χρησιμοποιούν συνδυασμό της αντιγραφής και της κατανομής δεδομένων στους κόμβους του συστήματος. Σε αυτή τη κατηγορία μεθόδων, τα δεδομένα κατανέμονται μεταξύ διαφορετικών κόμβων, αλλά ταυτόχρονα αντιγράφονται σε πολλαπλούς κόμβους. Με τον τρόπο αυτό διασφαλίζεται τόσο η διατήρηση της απόδοσης του συστήματος και του μειωμένου αποθηκευτικού χώρου που απαιτείται, όσο και η διαθεσιμότητα και η ανθεκτικότητα των δεδομένων. Αυτό σημαίνει ότι ακόμη και αν ένας κόμβος αποτύχει ή παρουσιάσει βλάβη, τα δεδομένα παραμένουν διαθέσιμα και προσβάσιμα μέσω των αντιγράφων που υπάρχουν σε άλλους κόμβους.

Κάθε μία από τις παραπάνω μεθόδους προσφέρει διαφορετικά πλεονεκτήματα και έχει διαφορετικούς περιορισμούς, καθιστώντας την κατάλληλη για συγκεκριμένα σενάρια χρήσης. Η αντιγραφή δεδομένων είναι ιδανική για συστήματα που απαιτούν υψηλή διαθεσιμότητα και ανθεκτικότητα, αλλά μπορεί να γίνει δαπανηρή σε αποθηκευτικό χώρο. Η κατανομή δεδομένων προσφέρει υψηλή απόδοση και αποδοτικότητα, αλλά εισάγει τον κίνδυνο απώλειας δεδομένων σε περίπτωση αποτυχίας κόμβων. Ο συνδυασμός αντιγραφής και κατανομής προσφέρει μια ισορροπημένη λύση, εξασφαλίζοντας τόσο την απόδοση όσο και την αξιοπιστία, αλλά απαιτεί περισσότερο αποθηκευτικό χώρο και πιο πολύπλοκους μηχανισμούς διαχείρισης. Η επιλογή της κατάλληλης μεθόδου εξαρτάται από τις συγκεκριμένες ανάγκες και προδιαγραφές του συστήματος αποθήκευσης καθώς και τους διαθέσιμους πόρους.

1.4 Προκλήσεις και δυσκολίες στην κατανεμημένη αποθήκευση δεδομένων

Μία από τις βασικότερες προκλήσεις στην κατανεμημένη αποθήκευση δεδομένων είναι η επιλογή του κατάλληλου κόμβου για την αποθήκευση και την προσπέλαση των δεδομένων. Σε ένα κατανεμημένο σύστημα, τα δεδομένα δεν αποθηκεύονται σε ένα κεντρικό σημείο ή σε έναν μεμονωμένο κόμβο, αλλά κατανέμονται σε πολλαπλούς κόμβους, γεγονός που επηρεάζει άμεσα την απόδοση και τη λειτουργία του συστήματος αποθήκευσης. Η μέθοδος ή αλλιώς ο αλγόριθμος κατανομής πρέπει να διασφαλίζει την αποδοτική και ισότιμη διανομή των δεδομένων στους διαθέσιμους κόμβους. Ζητούμενο της μεθόδου είναι να αποφεύγεται η συγκέντρωση των δεδομένων σε λίγους κόμβους ή αλλιώς η άνιση κατανομή τους, καθώς κάτι τέτοιο μπορεί να οδηγήσει σε συμφόρηση, υπερφόρτωση ορισμένων από τους κόμβους και συνεπώς μειωμένη απόδοση του συστήματος. Παράλληλα, μια μη ισορροπημένη κατανομή των δεδομένων δεν επιτρέπει την αποδοτική αξιοποίηση των διαθέσιμων πόρων του συστήματος αποθήκευσης.

Επιπλέον, σε περιπτώσεις όπου διατηρούνται πολλαπλά αντίγραφα των αποθηκευμένων αντικειμένων στο σύστημα για λόγους ανθεκτικότητας και διαθεσιμότητας των δεδομένων, γίνεται επιλογή όχι ενός αλλά πολλαπλών κόμβων, καθώς απαιτείται η αποθήκευση ξεχωριστών αντιγράφων του ίδιου αντικειμένου σε περισσότερους από έναν κόμβους του συστήματος.

Μια ακόμα σημαντική πρόκληση στην κατανεμημένη αποθήκευση δεδομένων είναι ο συγχρονισμός των ξεχωριστών κόμβων και η διατήρηση μιας κοινής και συνεπούς κατάστασης σε όλους τους κόμβους του συστήματος. Σε ένα κατανεμημένο περιβάλλον, όπου τα δεδομένα αποθηκεύονται σε πολλαπλούς κόμβους, είναι κρίσιμο να διασφαλίζεται ότι κάθε κόμβος διαθέτει την πιο πρόσφατη και έγκυρη εκδοχή των δεδομένων. Επιπλέον, όλοι οι κόμβοι πρέπει να έχουν γνώση της ύπαρξης των υπόλοιπων κόμβων του συστήματος και να ενημερώνονται άμεσα για οποιαδήποτε αλλαγή στη σύνθεση του συστήματος, όπως για παράδειγμα η προσθήκη ή η αφαίρεση ενός κόμβου. Αν η ενημέρωση για μεταβολές στους κόμβους και τα δεδομένα δεν πραγματοποιείται με σωστό τρόπο και έγκαιρα, μπορεί να προκύψουν καταστάσεις ασυνέπειας στο σύστημα, κίνδυνος απώλειας δεδομένων ή μη αποτελεσματικής χρήσης των πόρων του συστήματος.

Η επίτευξη της συνέπειας στο σύστημα, συχνά απαιτεί σύνθετους μηχανισμούς συγχρονισμού και συντονισμού μεταξύ των κόμβων, ιδιαίτερα όταν πρόκειται για δυναμικά συστήματα με μεταβολές στην τοπολογία των κόμβων, συνεχείς εγγραφές, ενημερώσεις ή διαγραφές δεδομένων και αντικειμένων. Η ανάγκη για διατήρηση της κοινής κατάστασης των κόμβων και των δεδομένων σε πραγματικό χρόνο μπορεί να οδηγήσει σε αυξημένη πολυπλοκότητα του συστήματος και καθυστερήσεις λόγω της συνεχούς επικοινωνίας και ανταλλαγής πληροφορίας μεταξύ των κόμβων.

ΚΕΦΑΛΑΙΟ 2 Αποθήκευση δεδομένων βάση κλειδιού-τιμής

Η αποθήκευση δεδομένων ως αντικείμενα με βάση του συνδυασμού ενός κλειδιού και μίας τιμής ή αλλιώς μέθοδο κλειδιού-τιμής (key-value) είναι μια από τις πιο απλές μεθόδους για την αποθήκευση και τη διαχείριση δεδομένων. Αυτή η προσέγγιση χρησιμοποιείται πολύ συχνά σε εφαρμογές όπου απαιτείται ταχύτητα, απλότητα και ευελιξία. Βασίζεται στην ιδέα της χρήσης ενός μοναδικού κλειδιού για την πρόσβαση σε μια συγκεκριμένη πληροφορία ή αλλιώς τιμή, προσφέροντας έναν αποδοτικό τρόπο για την οργάνωση, την αναζήτηση και την ανάκτηση δεδομένων.

Πιο συγκεκριμένα αυτή η προσέγγιση χρησιμοποιείται σε συστήματα βάσεων δεδομένων, συστήματα αποθήκευσης αρχείων και πολυμέσων όπως και σε άλλες και δομές δεδομένων, όπως πίνακες και λεξικά. Η βασική της ιδέα της μεθόδου είναι η χρήση ενός μοναδικού κλειδιού για την πρόσβαση σε μια συγκεκριμένη τιμή. Σε αυτό το μοντέλο, τα δεδομένα είναι ένα σύνολο αντικειμένων που περιγράφονται από ένα ζεύγος ενός κλειδιού και μίας τιμής

Το κλειδί είναι ένα μοναδικό αναγνωριστικό που χρησιμοποιείται για την πρόσβαση ή την προσπέλαση σε μια συγκεκριμένη τιμή. Μπορεί να είναι μια συμβολοσειρά (string) ή ένας αριθμός (integer), ανάλογα με την υλοποίηση. Η μοναδικότητα του κλειδιού είναι απαραίτητη, ώστε να αποφευχθεί η αποθήκευση πολλών διαφορετικών τιμών με το ίδιο κλειδί που θα οδηγήσει στη δημιουργία διπλοτύπων.

Η τιμή είναι τα δεδομένα που αποθηκεύονται και συνδέονται με το κλειδί. Μπορεί να είναι οποιοδήποτε τύπο δεδομένων, όπως αριθμοί, συμβολοσειρές, λίστες, αντικείμενα, ή ακόμη και πολύπλοκες δομές δεδομένων. Η ευελιξία στην αποθήκευση τιμών κάνει αυτή τη μέθοδο ιδανική για εφαρμογές που απαιτούν την αποθήκευση ετερογενών δεδομένων.

Στο πλαίσιο της παρούσας εργασίας, όταν συναντάται ο όρος αντικείμενο, αναφέρεται σε ένα δεδομένο που αποτελείται από έναν συνδυασμό ενός κλειδιού και μίας τιμής. Κάθε αντικείμενο περιγράφεται από ένα ζεύγος κλειδιού-τιμής, όπου το κλειδί είναι το μοναδικό αναγνωριστικό για την προσπέλαση του δεδομένου και η τιμή είναι το συγκεκριμένο δεδομένο που αποθηκεύεται.

2.1 Πλεονεκτήματα στην αποθήκευση δεδομένων βάση κλειδιού-τιμής

Η αποθήκευση αντικειμένων με τη χρήση κλειδιού-τιμής προσφέρει σημαντικά πλεονεκτήματα, τα οποία την καθιστούν ιδανική για πολλές σύγχρονες εφαρμογές. Αυτά τα πλεονεκτήματα περιλαμβάνουν:

1. Ταχύτητα: Η αναζήτηση δεδομένων πραγματοποιείται άμεσα μέσω του κλειδιού, χωρίς την ανάγκη εκτέλεσης περίπλοκων ερωτημάτων ή σύνθετων διαδικασιών ανάκτησης. Η προσπέλαση των δεδομένων γίνεται σε σταθερό χρόνο, ανεξάρτητα από το πλήθος των αποθηκευμένων αντικειμένων, διασφαλίζοντας υψηλή απόδοση στην προσπέλαση των δεδομένων.
2. Απλότητα: Η αποθήκευση δεδομένων με τη μέθοδο κλειδιού-τιμής είναι ιδιαίτερα απλή και εύκολη στην κατανόηση. Τα δεδομένα αποθηκεύονται ανεξάρτητα, χωρίς περίπλοκες σχέσεις, σύνθετα σχήματα ή με τη χρήση πολύπλοκων δομών δεδομένων. Αυτή η απλότητα καθιστά τόσο τον σχεδιασμό όσο και την υλοποίηση του συστήματος πιο εύκολη, μειώνοντας την πολυπλοκότητα στην αποθήκευση και τη διαχείριση των δεδομένων.
3. Ευελιξία: Οι τιμές των αντικειμένων μπορούν να περιέχουν διαφορετικούς τύπους δεδομένων, όπως αριθμούς, συμβολοσειρές, λίστες, αρχεία ή ακόμη και πιο σύνθετες δομές δεδομένων. Αυτή η ευελιξία επιτρέπει την αποθήκευση ετερογενών δεδομένων χωρίς περιορισμούς, καθιστώντας τη μέθοδο ιδιαίτερα χρήσιμη για εφαρμογές με διαφορετικές και απαιτητικές ανάγκες αποθήκευσης δεδομένων.
4. Κλιμάκωση: Η αποθήκευση δεδομένων με τη μέθοδο κλειδιού και τιμής είναι ιδανική για συστήματα που απαιτούν υψηλή κλιμακωσιμότητα και αποθήκευση μεγάλου όγκου δεδομένων. Το σύστημα μπορεί εύκολα να επεκταθεί επιτρέποντας την αποθήκευση περισσότερων αντικειμένων χωρίς να δημιουργούνται περιορισμοί που επηρεάζουν την απόδοση ή τη λειτουργικότητά του.

ΚΕΦΑΛΑΙΟ 3 Βασικές τεχνικές κατανεμημένης αποθήκευσης δεδομένων

Στο παρόν κεφάλαιο θα παρουσιαστούν και θα αναλυθούν οι βασικότερες μέθοδοι κατανεμημένης αποθήκευσης δεδομένων, οι οποίες έχουν σχεδιαστεί για να διαμοιράζουν τα δεδομένα μεταξύ των κόμβων ενός κατανεμημένου συστήματος με διαφορετικούς τρόπους. Κάθε μία από αυτές τις τεχνικές ακολουθεί διαφορετική προσέγγιση ως προς την τοποθέτηση

και την ανάκτηση των δεδομένων, γεγονός που επηρεάζει τη συνολική λειτουργία του συστήματος. Οι τεχνικές που θα εξεταστούν περιλαμβάνουν:

- Κυκλική Κατανομή (Round Robin Allocation)
- Στατική κατανομή (Static Partitioning)
- Κατανομή με Υπόλοιπο της τιμής κατακερματισμού (Hash Modulo)
- Αντιγραφή Δεδομένων (Replication)
- Δυναμικός Κατακερματισμός (Dynamic Hashing)

Κάθε μία από αυτές τις μεθόδους έχει συγκεκριμένα πλεονεκτήματα και περιορισμούς, οι οποίοι καθορίζουν την καταλληλότητά της για συγκεκριμένες εφαρμογές και για διαφορετικά περιβάλλοντα. Για παράδειγμα, ορισμένες τεχνικές δίνουν προτεραιότητα στην ισορροπημένη κατανομή του φορτίου, ενώ άλλες επικεντρώνονται στην ανθεκτικότητα σε αποτυχίες ή στη βελτιστοποίηση της ταχύτητας εγγραφής και ανάκτησης δεδομένων.

Στην ανάλυση που ακολουθεί, θα παρουσιαστούν οι βασικές αρχές λειτουργίας κάθε τεχνικής, καθώς και οι προκλήσεις που προκύπτουν κατά την εφαρμογή τους σε πραγματικά συστήματα. Ιδιαίτερη έμφαση θα δοθεί στον τρόπο με τον οποίο επηρεάζουν την κατανομή των αντικειμένων στους διαθέσιμους κόμβους ενός συστήματος, τη διαθεσιμότητα των δεδομένων, την απόδοση των λειτουργιών ανάγνωσης και εγγραφής, καθώς και τη συνολική ανθεκτικότητα του συστήματος έναντι αποτυχιών.

Η σύγκριση και η αξιολόγηση των μεθόδων αυτών είναι απαραίτητη για την κατανόηση των χαρακτηριστικών και των παραγόντων που καθορίζουν την επιλογή της καταλληλότερης προσέγγισης, ανάλογα με τις απαιτήσεις του εκάστοτε συστήματος. Μέσα από αυτή τη διαδικασία, θα εξαχθούν συμπεράσματα που θα συμβάλουν στην ανάπτυξη του συστήματος αποθήκευσης δεδομένων της παρούσας εργασίας. Συγκεκριμένα, θα γίνει επιλογή κατάλληλων μεθόδων ή χαρακτηριστικών των μεθόδων, τα οποία θα ληφθούν υπόψη κατά τον σχεδιασμό και την υλοποίηση του συστήματος.

3.1 Κυκλική κατανομή (Round Robin)

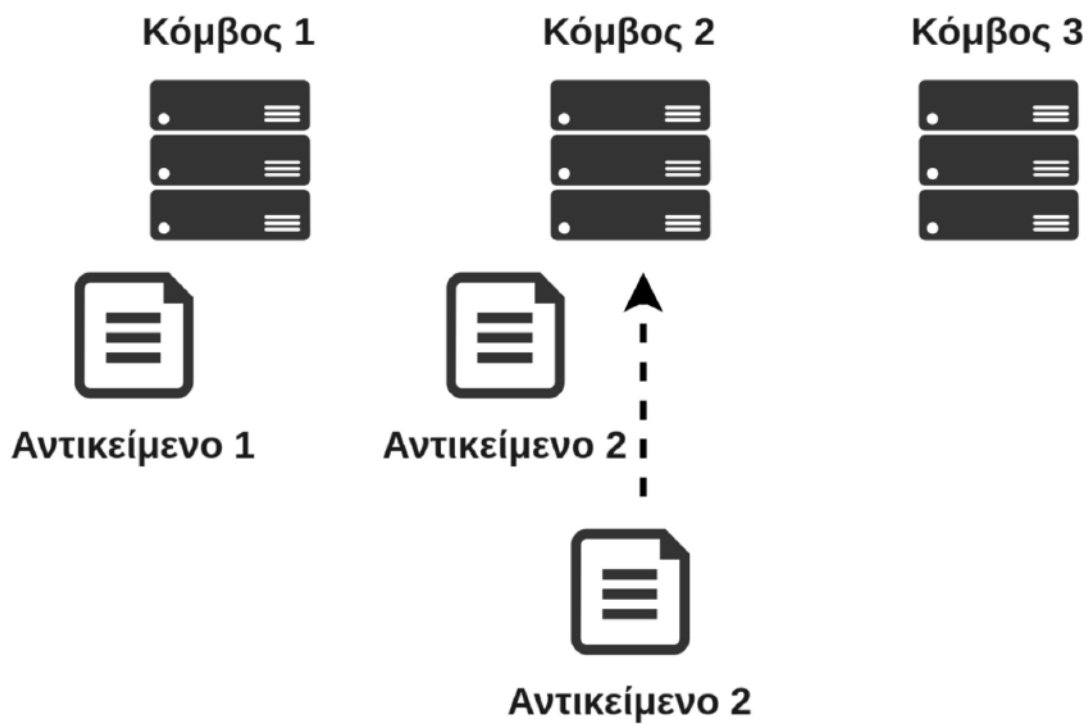
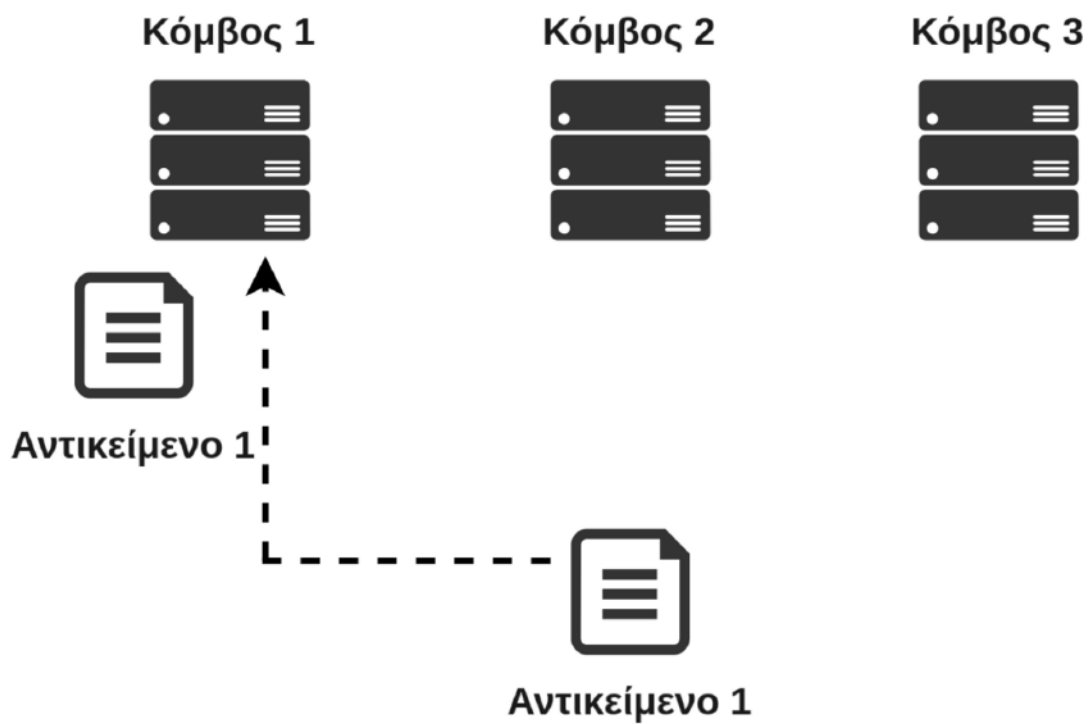
Η τεχνική της κυκλικής κατανομής ή αλλιώς Round Robin είναι μια από τις απλούστερες και πιο άμεσα εφαρμόσιμες μεθόδους κατανομής δεδομένων σε κατανεμημένα συστήματα αποθήκευσης, όπου τα αντικείμενα διανέμονται κυκλικά στους διαθέσιμους κόμβους του συστήματος. Πιο συγκεκριμένα, στην αποθήκευση δεδομένων, το πρώτο αντικείμενο αποθηκεύεται στον πρώτο κόμβο, το δεύτερο αντικείμενο στον δεύτερο κόμβο, και η διαδικασία συνεχίζεται έως ότου φτάσει στον τελευταίο κόμβο του συστήματος. Μόλις ολοκληρωθεί ένας κύκλος κατανομής, η διαδικασία ξεκινά ξανά από τον πρώτο κόμβο,

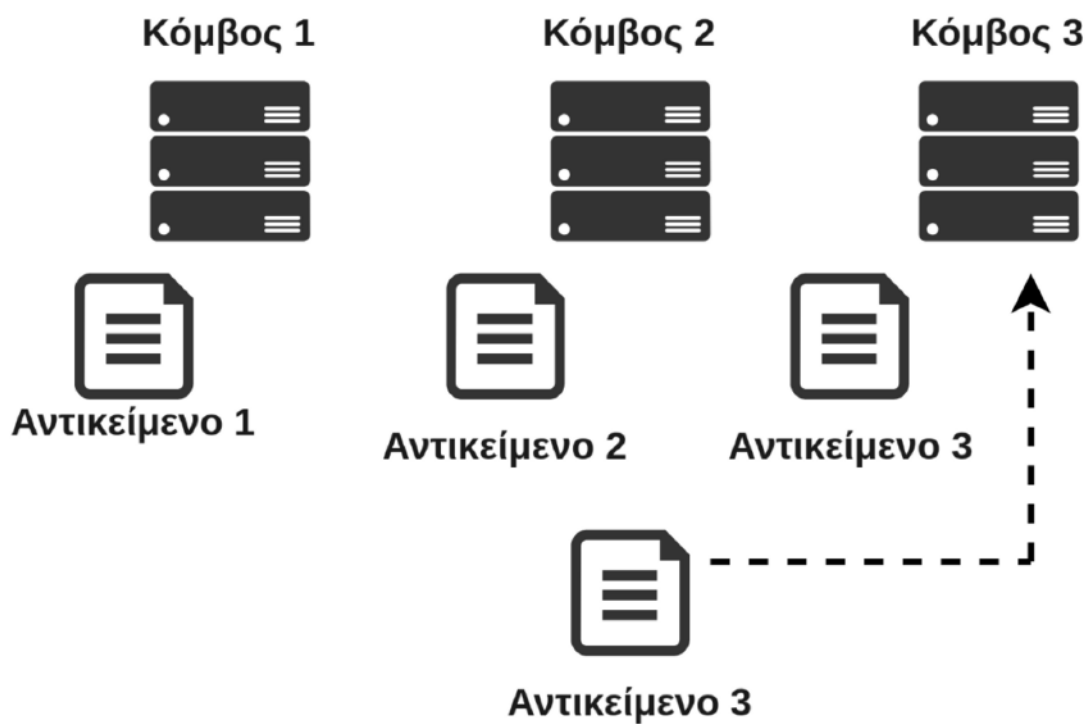
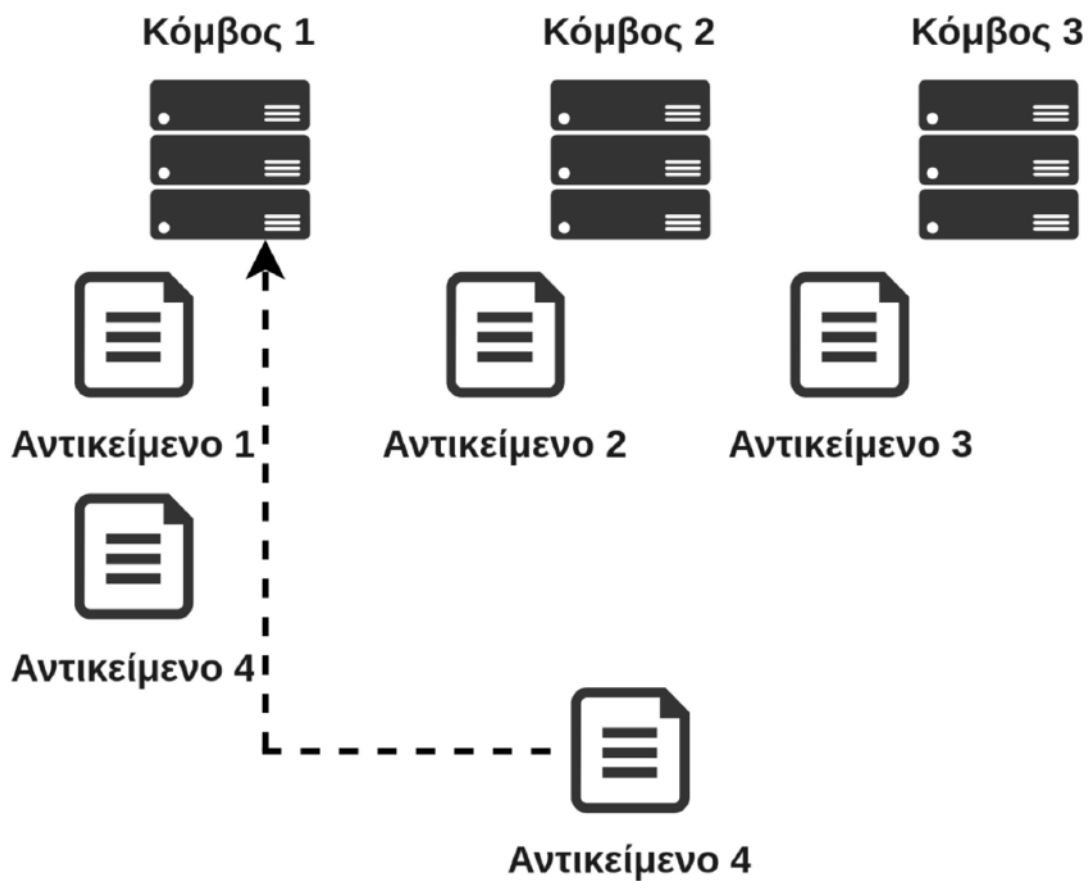
διατηρώντας με τον τρόπο αυτό μια ισορροπημένη κατανομή των αντικειμένων τους κόμβους του συστήματος.

Αυτή η προσέγγιση διασφαλίζει μια πολύ αποδοτική και ισορροπημένη κατανομή των δεδομένων, εφόσον ο αριθμός των κόμβων παραμένει σταθερός, αποτρέποντας έτσι τη συγκέντρωση αντικειμένων σε συγκεκριμένους κόμβους και τη δημιουργία σημείων συμφόρησης στο σύστημα εφόσον τα δεδομένα κατανέμονται κυκλικά επιτυχάνοντας μία ιδανική κατανομή. Ωστόσο, ένα σημαντικό μειονέκτημα της συγκεκριμένης μεθόδου είναι η αδυναμία της να προσαρμοστεί δυναμικά σε αλλαγές στη σύνθεση του συστήματος. Σε περίπτωση προσθήκης ή αφαίρεσης κόμβων, η υπάρχουσα κατανομή των δεδομένων δεν αναδιαμορφώνεται αυτόματα, καθιστώντας απαραίτητη μια πλήρη ανακατανομή των αντικειμένων στο σύστημα για να διατηρηθεί η ισορροπία.

Επιπλέον, η συγκεκριμένη μέθοδος δεν προσφέρει έναν άμεσο τρόπο για τον εντοπισμό του κόμβου στον οποίο έχει αποθηκευτεί ένα συγκεκριμένο αντικείμενο. Αυτό οφείλεται στο γεγονός ότι η τοποθέτηση των αντικειμένων βασίζεται αποκλειστικά στη σειρά εισαγωγής τους στο σύστημα, με αποτέλεσμα κάθε αντικείμενο να μπορεί να αποθηκευτεί σε οποιονδήποτε κόμβο, ανάλογα με τη σειρά που εισάγεται. Χωρίς πρόσθετες πληροφορίες, δεν υπάρχει δυνατότητα άμεσης αναζήτησης της θέσης αποθήκευσης ενός συγκεκριμένου αντικειμένου γεγονός που σημαίνει ότι η ανάκτηση των δεδομένων μπορεί να απαιτήσει σάρωση ή αναζήτηση σε πολλούς κόμβους μέχρι να εντοπιστεί το ζητούμενο αντικείμενο. Αυτή η αδυναμία καθιστά τις λειτουργίες αναζήτησης και την προσπέλασης δεδομένων λιγότερο αποδοτικές, ιδιαίτερα σε συστήματα μεγάλης κλίμακας.

Για να αντιμετωπιστεί αυτό το πρόβλημα, απαιτείται η χρήση επιπλέον δομών ευρετηρίασης, όπως λίστες ή πίνακες, στις οποίες καταγράφεται σε ποιον κόμβο έχει αποθηκευτεί κάθε αντικείμενο. Αυτές οι δομές λειτουργούν ως ένας κατάλογος αναφορών και επιτρέπουν την ταχεία αναζήτηση του κόμβου που φιλοξενεί ένα αντικείμενο κατά τη διαδικασία προσπέλασης. Ωστόσο, η χρήση τέτοιων δομών εισάγει επιπλέον πολυπλοκότητα στο σύστημα. Η διατήρηση και η συνεχής ενημέρωση αυτών των δομών, ιδιαίτερα σε περιβάλλοντα με συχνές εγγραφές, επιφέρει αυξημένο υπολογιστικό κόστος και απαιτήσεις σε πόρους του συστήματος. Αυτό μπορεί να αποτελέσει σημαντικό μειονέκτημα, ειδικά σε συστήματα μεγάλης κλίμακας.





Εικόνες 3.1.1, 3.1.2, 3.1.3, 3.1.4: Αναπαράσταση εισαγωγής αντικειμένων σε ένα σύστημα αποθήκευσης 3 κόμβων που πραγματοποιεί κατανομή των δεδομένων με τη μέθοδο της κυκλικής κατανομής ή αλλιώς Round Robin.

3.1.α Πλεονεκτήματα της κυκλικής κατανομής

- **Ισορροπημένη κατανομή:** Η μέθοδος διασφαλίζει ότι τα δεδομένα κατανέμονται ομοιόμορφα στους επιμέρους κόμβους, χωρίς να συγκεντρώνονται δεδομένα σε μεμονωμένους κόμβους δημιουργώντας σημεία συμφόρησης στο σύστημα
- **Απλότητα εφαρμογής:** Η απλότητα της συγκεκριμένης μεθόδου καθώς και η ευκολία υλοποίησης της, την καθιστά ιδανική για συστήματα που απαιτούν γρήγορη και απλή διαχείριση δεδομένων.
- **Αποδοτικότητα σε σταθερά συστήματα:** Όταν ο αριθμός των κόμβων παραμένει σταθερός, η μέθοδος είναι πολύ αποδοτική, καθώς εξασφαλίζει γρήγορη και πολύ ισορροπημένη κατανομή των δεδομένων.

3.1.β Μειονεκτήματα της κυκλικής κατανομής

- **Αδυναμία δυναμικής προσαρμογής:** Σε περίπτωση προσθήκης ή αφαίρεσης κόμβων, η υπάρχουσα κατανομή των δεδομένων δεν αναδιαμορφώνεται αυτόματα. Αυτό απαιτεί μια πλήρη ανακατανομή των δεδομένων για να διατηρηθεί η ισορροπία στην κατανομή.
- **Αποδοτικότητα αναζήτησης δεδομένων:** Η μέθοδος δεν παρέχει άμεση πληροφορία σχετικά με τη θέση αποθήκευσης ενός συγκεκριμένου αντικειμένου. Αυτό καθιστά την αναζήτηση δεδομένων λιγότερο αποδοτική, ιδιαίτερα σε συστήματα μεγάλης κλίμακας.
- **Ανάγκη για πρόσθετες δομές ευρετηρίασης:** Λόγω της έλλειψης πληροφορίας σχετικά με τη θέση των δεδομένων, απαιτείται η χρήση εξωτερικών δομών ευρετηρίασης. Αυτές οι δομές αυξάνουν την πολυπλοκότητα του συστήματος και επιβαρύνουν τη διαχείριση, καθώς πρέπει να διατηρούνται συνεχώς ενημερωμένες.

3.2 Στατική Κατανομή (Static Partitioning)

Η τεχνική της στατικής κατανομής (Static Partitioning) είναι μια από τις πιο βασικές μεθόδους κατανομής δεδομένων σε συστήματα αποθήκευσης με πολλαπλούς κόμβους. Με βάση αυτή τη προσέγγιση, κάθε αντικείμενο αντιστοιχίζεται άμεσα σε έναν κόμβο του συστήματος με βάση έναν προκαθορισμένο κανόνα ή κάποιο κριτήριο. Αυτός ο κανόνας μπορεί να βασίζεται σε διάφορα χαρακτηριστικά ή κριτήρια του αντικειμένου, όπως το κλειδί του αντικειμένου, μια μαθηματική συνάρτηση (π.χ. συνάρτηση κατακερματισμού ή αλλιώς hash function), καθώς και άλλα χαρακτηριστικά των δεδομένων.

Στην στατική κατανομή, κάθε αντικείμενο αντιστοιχίζεται άμεσα σε έναν κόμβο με βάση έναν κανόνα. Για παράδειγμα, σε ένα σύστημα με τρεις κόμβους, θα μπορούσε να χρησιμοποιηθεί ο πρώτος χαρακτήρας του κλειδιού του αντικειμένου ως κριτήριο κατανομής. Πιο συγκεκριμένα:

- Τα αντικείμενα με κλειδιά που ξεκινούν από το "Α" έως το "Η" αποθηκεύονται στον πρώτο κόμβο του συστήματος.
- Τα αντικείμενα με κλειδιά που ξεκινούν από το "Θ" έως το "Λ" αποθηκεύονται στον δεύτερο κόμβο.
- Τα αντικείμενα με κλειδιά που ξεκινούν από το "Μ" έως το "Ω" αποθηκεύονται στον τρίτο και τελευταίο κόμβο.

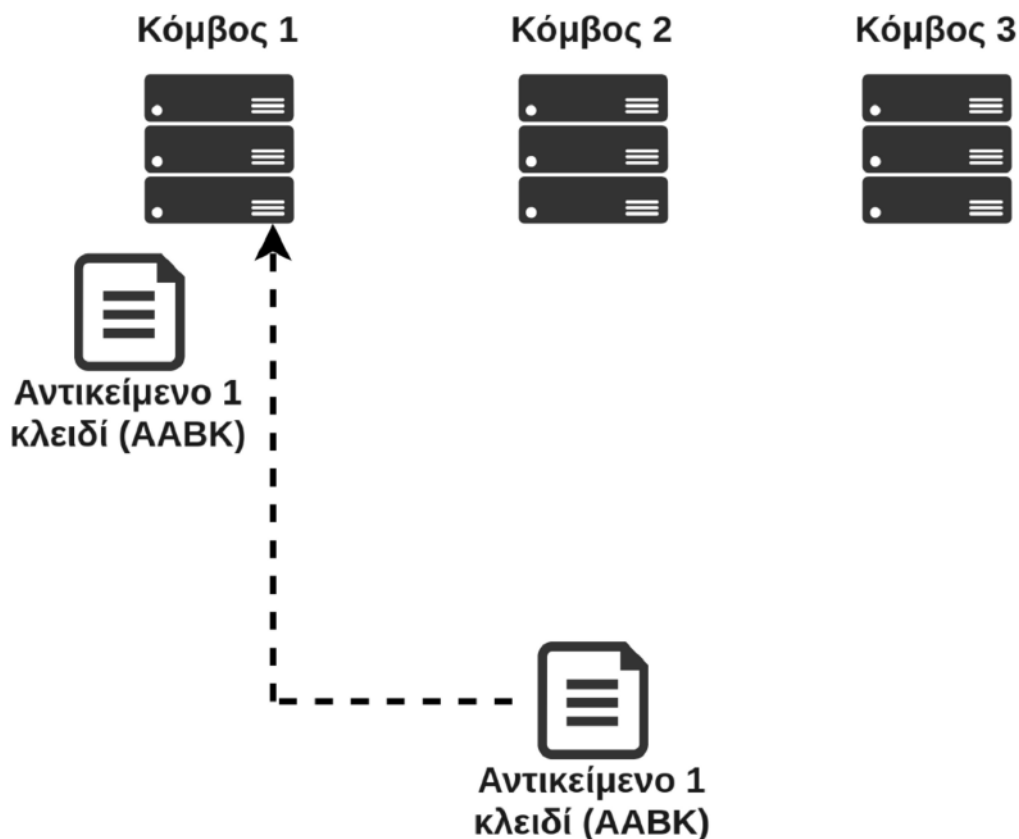
Σε αντίθεση με την κυκλική κατανομή (Round Robin) που περιγράφεται σε προηγούμενο κεφάλαιο, η οποία βασίζεται σε μια σειριακή και κυκλική κατανομή των δεδομένων στους κόμβους, η στατική κατανομή χρησιμοποιεί έναν προκαθορισμένο κανόνα για να αντιστοιχίσει κάθε αντικείμενο σε έναν συγκεκριμένο κόμβο. Αυτή η προσέγγιση διασφαλίζει μια πιο γρήγορη και άμεση προσπέλαση αντικειμένων από τους κόμβους του συστήματος, καθώς ο υπεύθυνος κόμβος για την αποθήκευση κάθε αντικειμένου υπολογίζεται με ντετερμινιστικό τρόπο βάσει κάποιου κριτηρίου. Δηλαδή, με βάση το κριτήριο ή τον κανόνα κατανομής, ο κόμβος στον οποίο θα αποθηκευτεί το αντικείμενο είναι πάντα ο ίδιος. Αυτό διευκολύνει την ταχεία προσπέλαση των δεδομένων, καθώς δεν απαιτείται αναζήτηση σε πολλούς κόμβους για να βρεθεί το αντικείμενο ή κάποια δομή ευρετηρίασης. Αντίθετα, στη συγκεκριμένη μέθοδο η θέση ή αλλιώς ο ο κόμβος ενός αντικειμένου υπολογίζεται άμεσα με βάση τον κανόνα χαρτογράφησης.

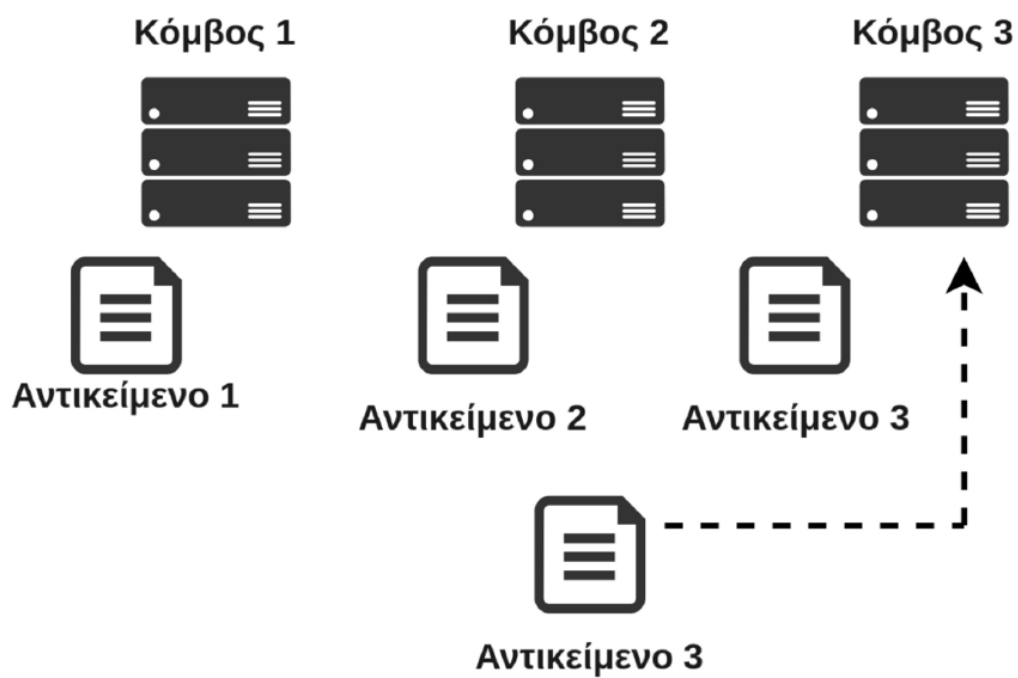
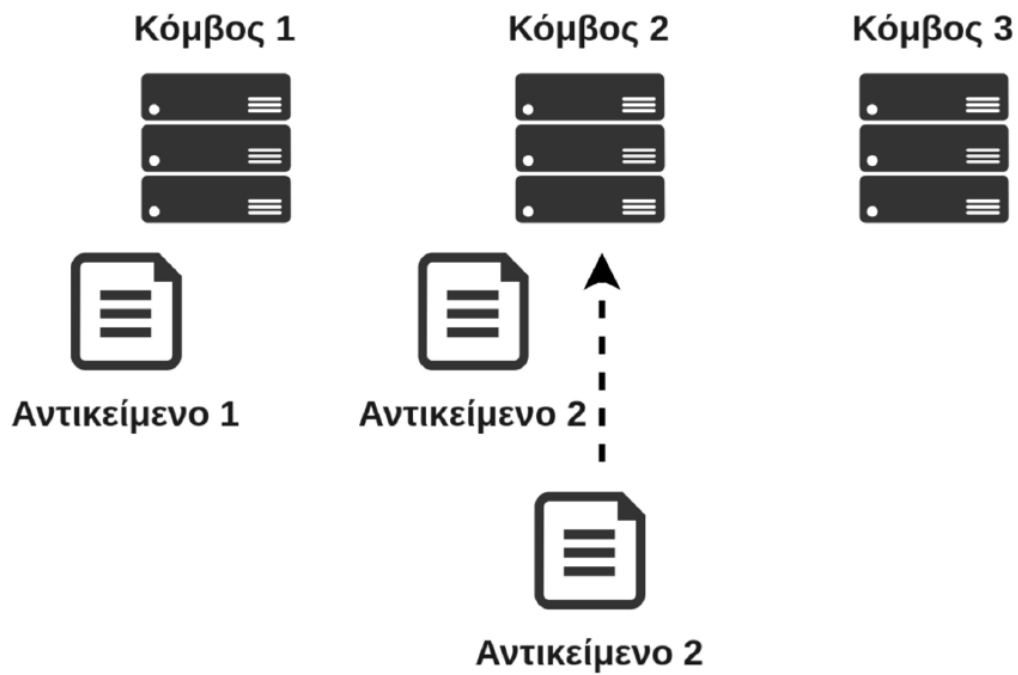
Επιπλέον, η υλοποίησή της συγκεκριμένης μεθόδου είναι επίσης απλή και εύκολη καθώς δεν περιλαμβάνει πολυπλοκότητα στον σχεδιασμό της, καθιστώντας την ιδανική για την λειτουργία συστημάτων αποθήκευσης μικρής κλίμακας και μη δυναμικά περιβάλλοντα. Ωστόσο σε περιβάλλοντα με αλλαγές στην σύσταση των κόμβων όπως για παράδειγμα σε περιπτώσεις προσθήκης ή αφαίρεσης ενός κόμβου, η υπάρχουσα κατανομή καθίσταται άκυρη και απαιτείται ανακατανομή των αντικειμένων.

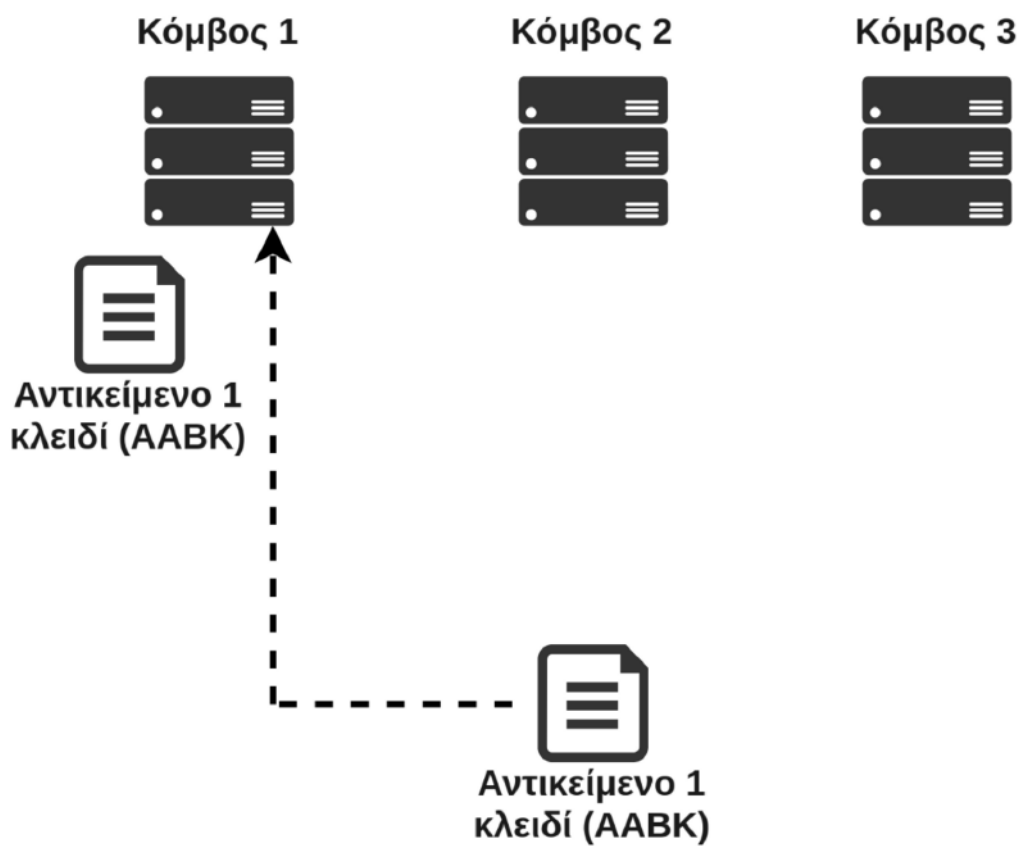
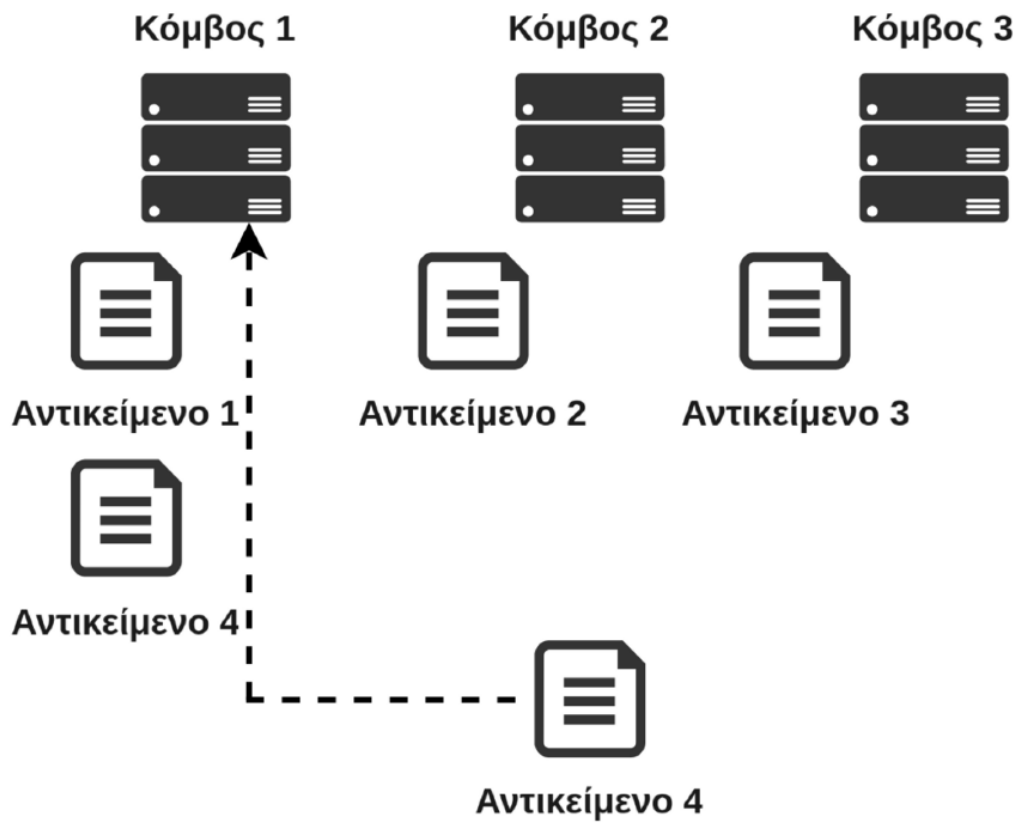
Η κατανομή των δεδομένων στην στατική κατανομή (Static Partitioning) δεν είναι τόσο ομοιόμορφη και ισότιμη σε σύγκριση με την μέθοδο της κυκλικής κατανομής (Round Robin).

Στην κυκλική κατανομή, τα δεδομένα κατανέμονται ομοιόμορφα στους κόμβους, καθώς κάθε νέο αντικείμενο τοποθετείται στον επόμενο διαθέσιμο κόμβο με κυκλική σειρά. Αυτό εξασφαλίζει μια ισορροπημένη κατανομή των αντικειμένων και αποτρέπει τη δημιουργία σημείων συμφόρησης.

Αντίθετα, στην συγκεκριμένη μέθοδο, η κατανομή των δεδομένων εξαρτάται σε μεγάλο βαθμό από τον αλγόριθμο, τα χαρακτηριστικά των δεδομένων καθώς και τα κριτήρια που χρησιμοποιούνται. Αν ο κανόνας δεν είναι σχεδιασμένος σωστά ή τα δεδομένα δεν κατανέμονται ομοιόμορφα βάσει των κριτηρίων, μπορεί να οδηγήσουν σε ανομοιόμορφη κατανομή των αντικειμένων στους κόμβους του συστήματος. Αυτό μπορεί να δημιουργήσει σημεία συμφόρησης, όπου ορισμένοι κόμβοι διαθέτουν πολλά αντικείμενα και να υπερφορτώνονται, ενώ άλλοι κόμβοι παραμένουν ανεκμετάλλετοι. Στη μέθοδο αυτή, εάν δεν εξασφαλίζεται μια σχετικά ομοιόμορφη κατανομή των αντικειμένων, τότε η απόδοση του συστήματος μπορεί να μειωθεί σε σημαντικό βαθμό.







Εικόνες 3.2.1, 3.2.2, 3.2.3, 3.2.4, 3.2.5: Αναπαράσταση εισαγωγής αντικειμένων σε ένα σύστημα αποθήκευσης που αποτελείται από 3 κόμβους και κατανέμει τα αντικείμενα σύμφωνα με την μέθοδο της στατικής κατανομής (Static Partitioning)

3.2.α Πλεονεκτήματα της στατικής κατανομής

- Απλότητα και ευκολία υλοποίησης: Η στατική κατανομή είναι εύκολη στην υλοποίηση, καθώς βασίζεται σε έναν απλό κανόνα αντιστοίχισης.
- Ταχεία προσπέλαση αντικειμένων: Εφόσον ο κόμβος στον οποίο αποθηκεύεται ένα αντικείμενο προσδιορίζεται άμεσα από τον κανόνα χαρτογράφησης, η προσπέλαση των δεδομένων είναι γρήγορη και αποδοτική.
- Δεν απαιτούνται πρόσθετες δομές ευρετηρίασης: Σε αντίθεση με άλλες μεθόδους, η στατική κατανομή δεν απαιτεί τη χρήση επιπλέον δομών ευρετηρίασης για την καταγραφή της θέσης των αντικειμένων.
- Αποδοτικότητα σε σταθερά συστήματα: Όταν ο αριθμός των κόμβων παραμένει σταθερός, η μέθοδος είναι πολύ αποδοτική, καθώς εξασφαλίζει γρήγορη και πολύ ισορροπημένη κατανομή των δεδομένων.

3.2.β Μειονεκτήματα της στατικής κατανομής

- Δεν εγγυάται ομοιόμορφη κατανομή: Η κατανομή των δεδομένων ενδέχεται να είναι μη ομοιόμορφη. Η ομοιομορφία της κατανομής εξαρτάται από τον κανόνα χαρτογράφησης, και τα χαρακτηριστικά των δεδομένων.
- Δυσκολία σε δυναμικά περιβάλλοντα: Η στατική κατανομή δεν προσαρμόζεται τόσο εύκολα σε αλλαγές στη σύνθεση του συστήματος και απαιτεί ανακατανομή των δεδομένων.

3.3 Αντιγραφή δεδομένων (Replication)

Η μέθοδος της αντιγραφής δεδομένων ή αλλιώς Replication, είναι μια από τις πιο σημαντικές τεχνικές για τη διασφάλιση της αξιοπιστίας του συστήματος αποθήκευσης, της αντοχής σε αποτυχίες ή σφάλματα καθώς και της διαθεσιμότητας των δεδομένων σε κατανεμημένα συστήματα αποθήκευσης. Βασίζεται στην ιδέα της δημιουργίας πολλαπλών αντιγράφων των αντικειμένων που απαιτείται να αποθηκευτούν και της αποθήκευσής τους σε διαφορετικούς κόμβους του συστήματος. Αυτό σημαίνει ότι κάθε αντικείμενο δεν αποθηκεύεται σε έναν μόνο κόμβο, αλλά αντιγράφεται σε δύο ή περισσότερους κόμβους, δημιουργώντας με τον τρόπο αυτό αντίγραφα ασφαλείας των αντικειμένων. Η αντιγραφή μπορεί να γίνει είτε πλήρης (όπου τα δεδομένα αντιγράφονται σε όλους τους κόμβους του συστήματος) είτε μερική (όπου τα δεδομένα αντιγράφονται σε ένα υποσύνολο του συνόλου των κόμβων).

Σε ένα σύστημα που λειτουργεί με τη μέθοδο της αντιγραφής δεδομένων, κάθε φορά που ένα αντικείμενο εισάγεται ή ενημερώνεται στο σύστημα, δημιουργούνται πολλαπλά αντίγραφα του αντικειμένου και αποθηκεύονται σε διαφορετικούς κόμβους του συστήματος. Για παράδειγμα, αν ένα αντικείμενο ζητηθεί να αποθηκευτεί σε έναν από τους κόμβους του συστήματος, μπορούν να δημιουργηθούν ίδια αντίγραφα του αντικειμένου, τα οποία να αποθηκευτούν σε άλλους κόμβους. Αυτό εξασφαλίζει ότι, σε περίπτωση που ένας από τους κόμβους αποτύχει, τα δεδομένα παραμένουν διαθέσιμα και προσβάσιμα στους υπόλοιπους κόμβους.

Η αντιγραφή δεδομένων είναι ιδανική για συστήματα όπου η αξιοπιστία και η διαθεσιμότητα των αποθηκευμένων δεδομένων είναι κρίσιμες. Ωστόσο, η εφαρμογή της αντιγραφής απαιτεί αυξημένο αποθηκευτικό χώρο, καθώς τα αντικείμενα πρέπει να αποθηκεύονται πολλαπλές φορές στο σύστημα. Ο συνολικός αποθηκευτικός χώρος που απαιτείται εξαρτάται από το μέγεθος ή τον όγκο των δεδομένων καθώς και τον αριθμό των αντιγράφων που διατηρούνται. Για παράδειγμα, αν ένα σύστημα αποθηκεύει δεδομένα συγκεκριμένου μεγέθους και διατηρεί 3 αντίγραφα, ο συνολικός αποθηκευτικός χώρος που απαιτείται είναι 3 φορές ο συνολικός όγκος των δεδομένων.

Σε ένα σύστημα που χρησιμοποιεί την αντιγραφή δεδομένων, οι λειτουργίες εγγραφής απαιτούν την ενημέρωση όλων των αντιγράφων ενός αντικειμένου σε πολλαπλούς κόμβους. Αυτή η διαδικασία μπορεί να είναι σχετικά μη αποδοτική, καθώς κάθε εγγραφή πρέπει να εκτελεστεί σε πολλαπλούς κόμβους, γεγονός που αυξάνει το φόρτο εργασίας του συστήματος. Ειδικά σε περιπτώσεις όπου ο αριθμός των αντιγράφων είναι μεγάλος, οι εγγραφές μπορεί να γίνουν αργές και να επηρεάσουν αρνητικά την απόδοση του συστήματος.

Από την άλλη πλευρά, οι λειτουργίες αναγνώσης είναι πολύ πιο αποδοτικές, καθώς τα δεδομένα μπορούν να ανακτηθούν και να προσπελαστούν από οποιονδήποτε από τους κόμβους που διατηρούν ένα αντίγραφο τους. Αυτό επιτρέπει την παράλληλη ανάγνωση δεδομένων από πολλαπλούς κόμβους, βελτιώνοντας σημαντικά την απόδοση του συστήματος στις αναγνώσεις.

Η αντιγραφή μπορεί να υλοποιηθεί με δύο κύριες προσεγγίσεις: την πλήρη αντιγραφή και την μερική αντιγραφή των δεδομένων. Κάθε μία από αυτές έχει τα δικά της πλεονεκτήματα

και τα μειονεκτήματα, ανάλογα με τις ανάγκες του συστήματος, τον αριθμό των κόμβων και το επιθυμητό επίπεδο ανοχής σε αποτυχίες.

3.3.α Πλήρης αντιγραφή δεδομένων

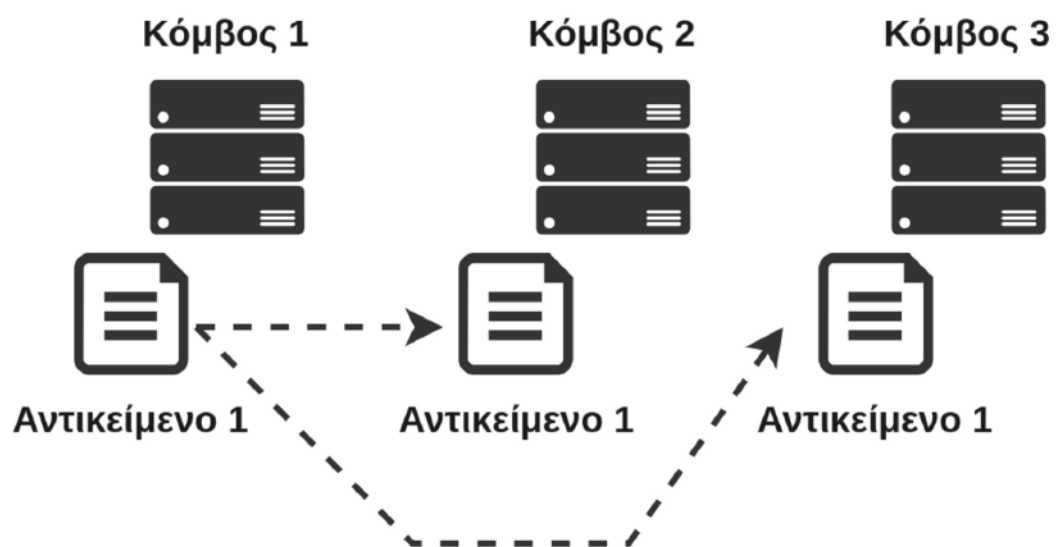
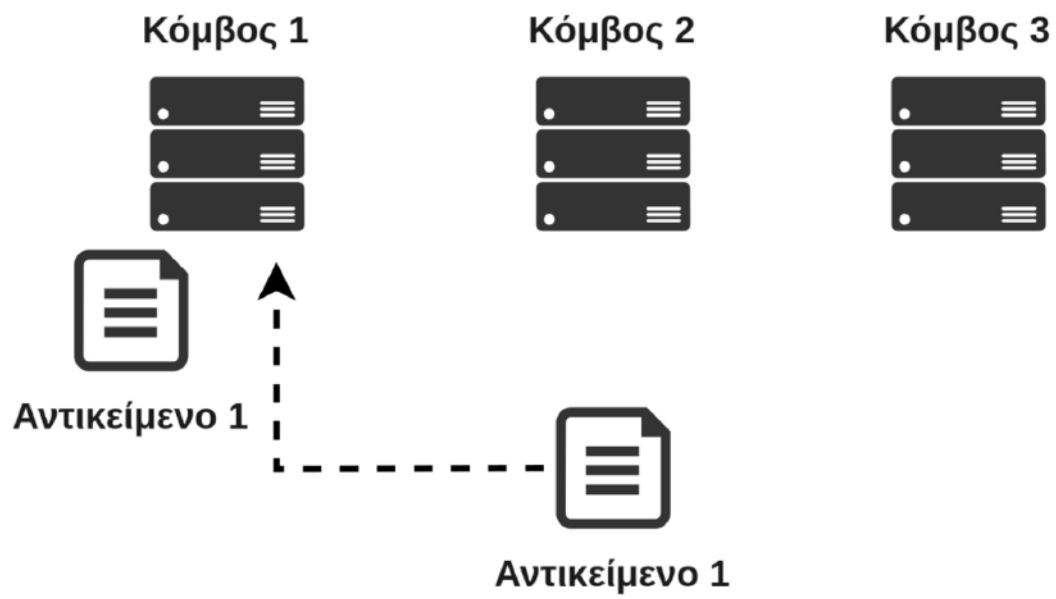
Στην προσέγγιση της πλήρους αντιγραφής, τα δεδομένα που αποθηκεύονται στο σύστημα αντιγράφονται σε όλους τους διαθέσιμους κόμβους. Αυτό σημαίνει ότι κάθε κόμβος διαθέτει ένα πλήρες αντίγραφο όλων των δεδομένων που είναι αποθηκευμένα στο σύστημα. Η πλήρης αντιγραφή εξασφαλίζει την υψηλότερη δυνατή διαθεσιμότητα στα δεδομένα και αντοχή σε αποτυχίες και βλάβες του συστήματος, καθώς ακόμα και αν πολλοί κόμβοι αποτύχουν, τα δεδομένα παραμένουν προσβάσιμα από οποιονδήποτε διαθέσιμο κόμβο. Για παράδειγμα, σε ένα σύστημα με 10 κόμβους, στη χειρότερη περίπτωση όπου οι 9 κόμβοι αποτύχουν ή παρουσιάσουν βλάβη, τα δεδομένα εξακολουθούν να είναι διαθέσιμα από τον έναν εναπομείναντα λειτουργικό κόμβο.

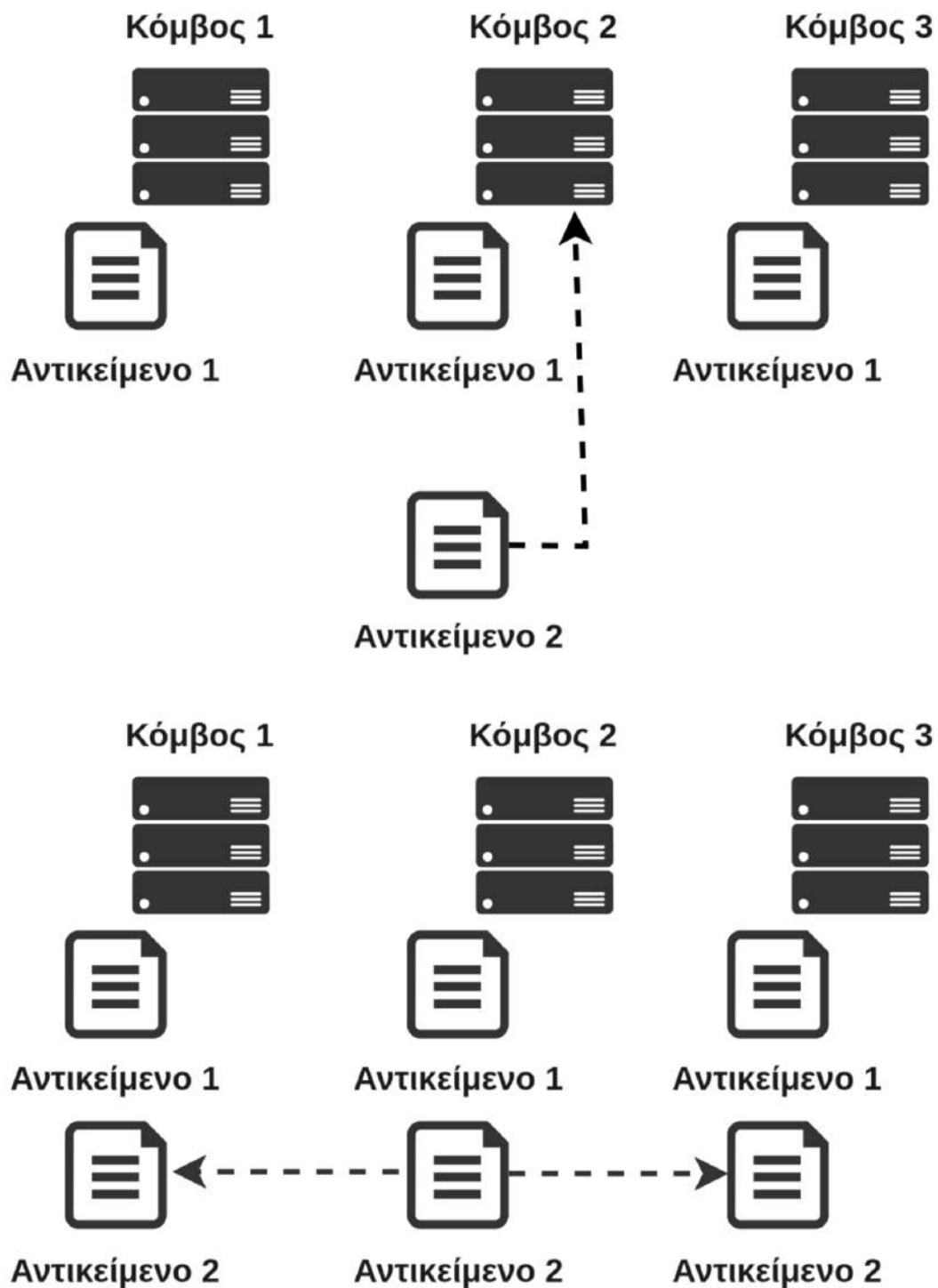
3.3.β Μερική αντιγραφή δεδομένων

Στην προσέγγιση της μερικής αντιγραφής, τα δεδομένα αντιγράφονται σε ένα υποσύνολο των κόμβων του συστήματος. Για παράδειγμα, σε ένα σύστημα με 10 κόμβους, τα αντικείμενα μπορεί να αντιγραφούν σε 2, 3 ή και περισσότερους κόμβους. Αυτή η προσέγγιση εξακολουθεί να εξασφαλίζει αυξημένη διαθεσιμότητα στα δεδομένα και ανθεκτικότητα του συστήματος σε βλάβες και αποτυχίες, αλλά με μειωμένες απαιτήσεις σε αποθηκευτικό χώρο σε σύγκριση με τη μέθοδο της πλήρους αντιγραφής των δεδομένων που παρουσιάζεται παραπάνω.

Η επιλογή του αριθμού των αντιγράφων εξαρτάται από τις απαιτήσεις του συστήματος και το επιθυμητό επίπεδο ανοχής σε αποτυχίες, σφάλματα και βλάβες. Για παράδειγμα, στην περίπτωση που διατηρούνται 3 αντίγραφα, το σύστημα μπορεί να αντέξει την αποτυχία έως 2 κόμβων χωρίς να υπάρχει κίνδυνος απώλειας δεδομένων. Αν διατηρούνται 5 αντίγραφα, το σύστημα μπορεί να αντέξει την αποτυχία έως 4 κόμβων. Αυτή η προσέγγιση είναι ιδανική για συστήματα όπου η διαθεσιμότητα είναι κρίσιμη, αλλά οι πόροι του συστήματος, όπως ο αποθηκευτικός χώρος, είναι πύο περιορισμένοι.

Η μερική αντιγραφή προσφέρει μια καλύτερη ισορροπία μεταξύ της διαθεσιμότητας των δεδομένων και του αποθηκευτικού χώρου που απαιτείται. Ωστόσο, η ανθεκτικότητα και η διαθεσιμότητα των δεδομένων εξαρτάται άμεσα από τον αριθμό των αντιγράφων που διατηρούνται στο σύστημα αποθήκευσης. Αν ο αριθμός των αντιγράφων είναι μικρός, όπως για παράδειγμα η διατήρηση 2 αντιγράφων, το σύστημα αποθήκευσης είναι λιγότερο αξιόπιστο, καθώς η αποτυχία περισσότερων κόμβων από τον αριθμό των αντιγράφων που διατηρούνται μπορεί να οδηγήσει σε απώλεια δεδομένων.





Εικόνες 3.3.1, 3.3.2: Αναπαράσταση εισαγωγής αντικειμένων σε ένα σύστημα αποθήκευσης 3 κόμβων που λειτουργεί με αντιγραφή των δεδομένων (Replication).

3.3.γ Πλεονεκτήματα της αντιγραφής δεδομένων

- Υψηλή διαθεσιμότητα δεδομένων: Η αντιγραφή δεδομένων εξασφαλίζει ότι τα δεδομένα παραμένουν προσβάσιμα ακόμα και σε περίπτωση αποτυχίας κάποιων κόμβων. Αυτό αυξάνει την αξιοπιστία του συστήματος.
- Ανθεκτικότητα σε αποτυχίες: Με την αντιγραφή των δεδομένων σε πολλαπλούς κόμβους, το σύστημα μπορεί να αντέξει την αποτυχία πολλών κόμβων χωρίς απώλεια δεδομένων, ανάλογα με τον αριθμό των αντιγράφων που διατηρούνται.
- Βελτιωμένη απόδοση στην ανάγνωση: Η ύπαρξη πολλαπλών αντιγράφων επιτρέπει την παράλληλη ανάγνωση δεδομένων από διαφορετικούς κόμβους, βελτιώνοντας την απόδοση του συστήματος.
- Αποδοτικότητα σε σταθερά συστήματα: Όταν ο αριθμός των κόμβων παραμένει σταθερός, η μέθοδος είναι πολύ αποδοτική, καθώς εξασφαλίζει γρήγορη και πολύ ισορροπημένη κατανομή των δεδομένων.

3.3.8 Μειονεκτήματα της αντιγραφής δεδομένων

- Αυξημένη ανάγκη αποθηκευτικού χώρου: Η αντιγραφή δεδομένων απαιτεί αυξημένο αποθηκευτικό χώρο, καθώς κάθε αντίγραφο ενός αντικειμένου καταλαμβάνει χώρο πολλαπλάσιο του μεγέθους του και του αριθμού των αντιγράφων που διατηρούνται.
- Εξάρτηση από τον αριθμό αντιγράφων: Όσο λιγότερα είναι τα αντίγραφα που διατηρούνται, τόσο μειώνεται η αξιοπιστία του συστήματος. Όσο περισσότερα αντίγραφα διατηρούνται, τόσο αυξάνονται οι ανάγκες σε αποθηκευτικό χώρο.
- Περιορισμένη δυνατότητα κλιμάκωσης: Η αύξηση του συνολικού αποθηκευτικού χώρου δεν είναι εύκολη, καθώς η προσθήκη νέων κόμβων δεν επεκτείνει τη διαθέσιμη χωρητικότητα, αλλά διατηρεί επιπλέον αντίγραφα των ίδιων δεδομένων.

3.4 Κατανομή με υπόλοιπο της τιμής κατακερματισμού (Hash Modulo)

Η μέθοδος κατανομής με βάση το υπόλοιπο της τιμής κατακερματισμού ή αλλιώς Hash Modulo είναι μια από τις βασικότερες τεχνικές για την κατανομή δεδομένων σε συστήματα αποθήκευσης με πολλαπλούς κόμβους. Βασίζεται στην ιδέα της αντιστοίχισης ενός αντικειμένου σε ένα κόμβο

βάσει μιας μαθηματικής συνάρτησης. Τα δεδομένα αντιστοιχίζονται σε κόμβους σύμφωνα με την τιμή του αποτελέσματος της συνάρτησης κατακερματισμού (hashing function). Η μέθοδος αυτή χρησιμοποιείται για να εξασφαλίσει μια ισορροπημένη κατανομή των δεδομένων στους διαθέσιμους κόμβους του συστήματος. Η απλότητα και η αποδοτικότητα της συγκεκριμένης μεθόδου την καθιστούν ιδανική για συστήματα όπου ο αριθμός των κόμβων παραμένει σταθερός, δηλαδή δεν υπάρχουν μεταβολές στη σύσταση των κόμβων του, καθώς επίσης σε περιπτώσεις όπου απαιτείται ταχύτητα στην προσπέλαση αποθηκευμένων δεδομένων.

Η λειτουργία της μεθόδου Hash Modulo βασίζεται σε δύο βασικά βήματα. Πρώτο βήμα είναι ο κατακερματισμός του κλειδιού του αντικειμένου και δεύτερο βήμα ο υπολογισμός του υπεύθυνου κόμβου στον οποίο θα αποθηκευτεί ή από τον οποίο θα προσπελαστεί το αντικείμενο.

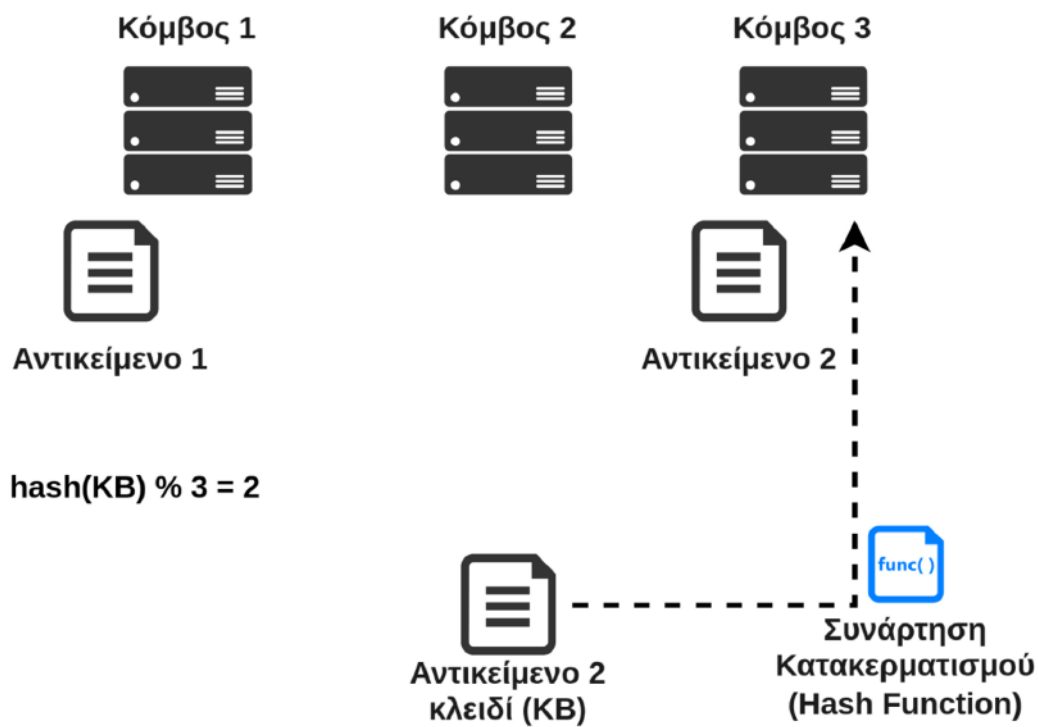
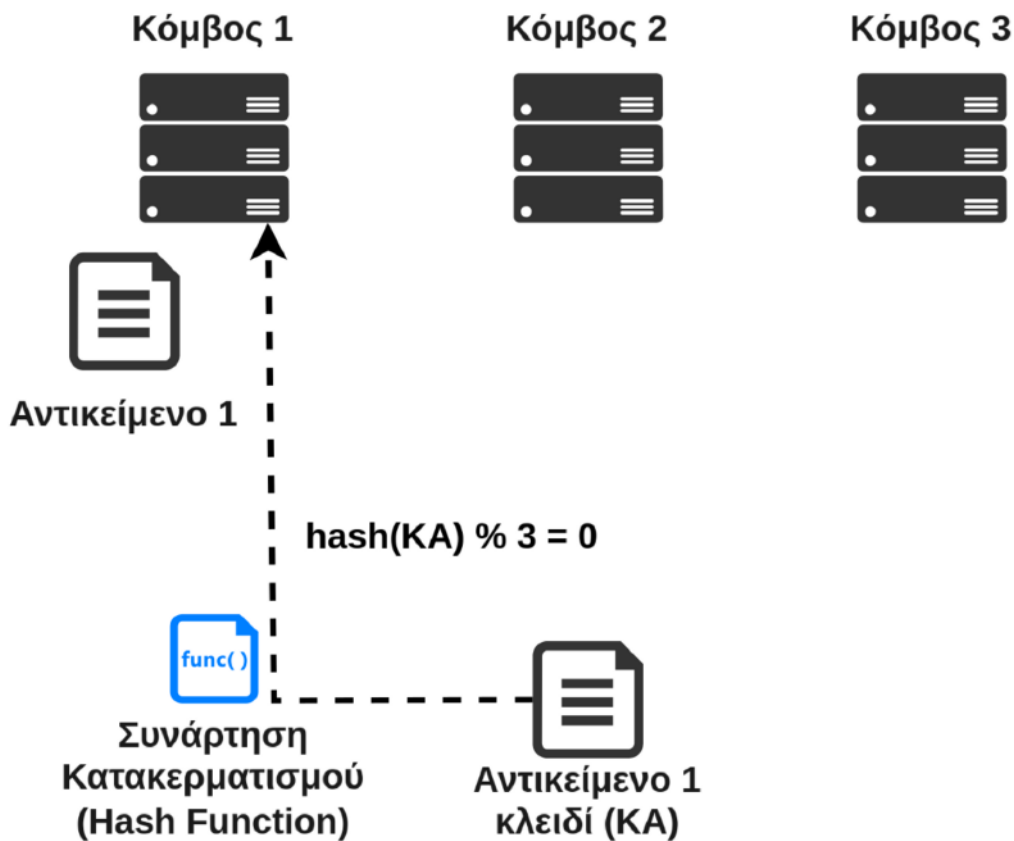
Πρώτο βήμα όπως περιγράφεται παραπάνω, είναι ο κατακερματισμός του κλειδιού του αντικειμένου. Όταν ένα αντικείμενο απαιτείται να αποθηκευτεί στο σύστημα, το κλειδί του αντικειμένου δίνεται ως είσοδος σε μια συνάρτηση κατακερματισμού (hash function). Η συνάρτηση αυτή μετατρέπει το κλειδί σε μια αριθμητική τιμή, γνωστή ως τιμή ή αποτέλεσμα κατακερματισμού (hash value), η οποία συνήθως είναι ένας μεγάλος ακέραιος αριθμός. Η συνάρτηση κατακερματισμού είναι σχεδιασμένη έτσι ώστε για διαφορετικές τιμές εισόδου ή κλειδιά να παράγει μια ομοιόμορφη κατανομή τιμών αποτελέσματος. Με τον τρόπο αυτό τα αντικείμενα που αποθηκεύονται στο σύστημα δεν συγκεντρώνονται σε συγκεκριμένους κόμβους και δεν παρουσιάζονται συχνά σημεία συμφόρησης.

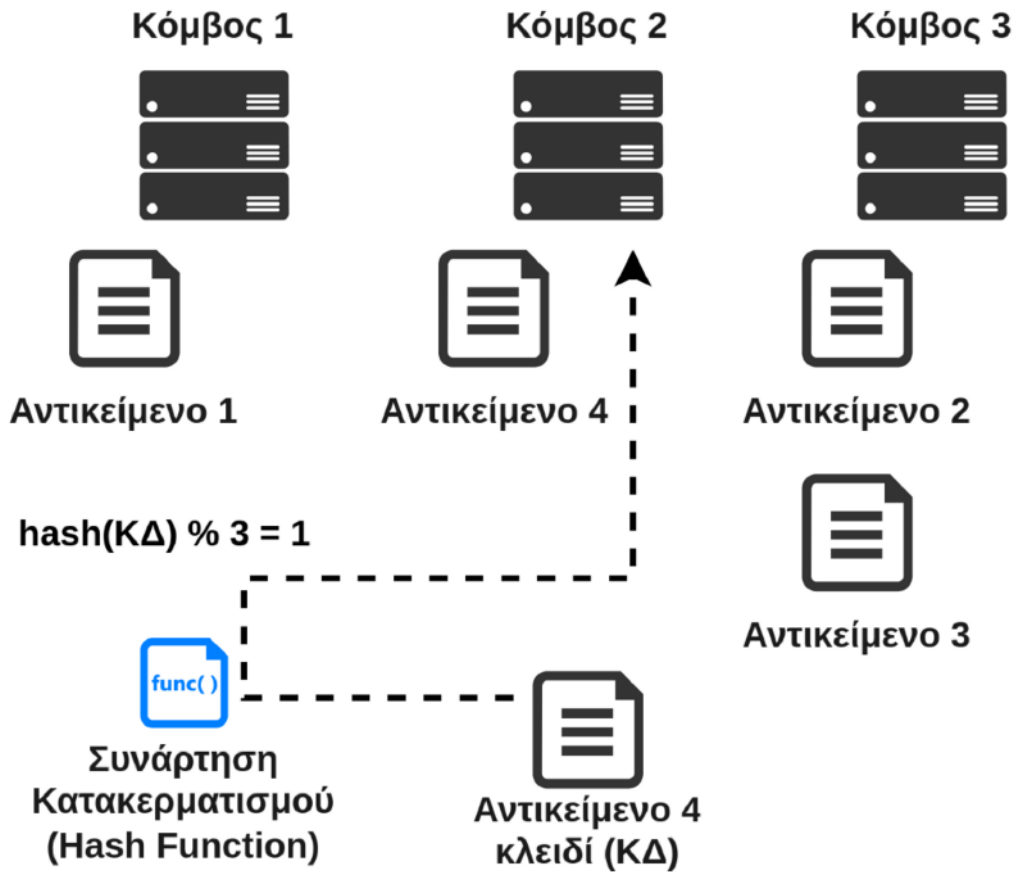
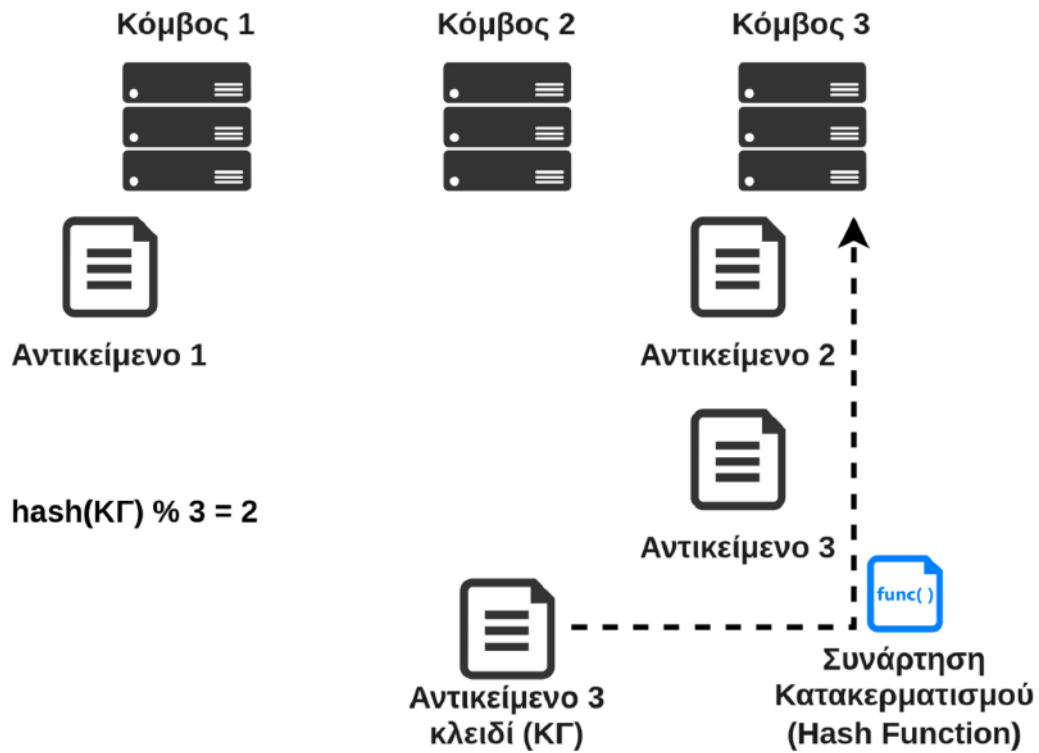
Δεύτερο βήμα αποτελεί ο υπολογισμός του υπεύθυνου κόμβου. Μόλις υπολογιστεί το αποτέλεσμα της τιμής ή αλλιώς αποτέλεσμα της συνάρτησης κατακερματισμού (hash value), το επόμενο βήμα είναι ο υπολογισμός του υπεύθυνου κόμβου στον οποίο θα αποθηκευτεί το αντικείμενο. Αυτό επιτυγχάνεται υπολογίζοντας το ακέραιο υπόλοιπο της διαίρεσης της τιμής κατακερματισμού (hash value) με τον συνολικό αριθμό των κόμβων του συστήματος (N), το οποίο καθορίζει τον υπεύθυνο κόμβο προορισμού του αντικειμένου. Για παράδειγμα, εάν το αποτέλεσμα κατακερματισμού του κλειδιού ενός αντικειμένου είναι ο αριθμός 25 και ο αριθμός των κόμβων του συστήματος είναι 10, τότε το αντικείμενο θα αποθηκευτεί στον έκτο κόμβο, καθώς το ακέραιο υπόλοιπο της διαίρεσης του αριθμού 25 με τον αριθμό 10 είναι η τιμή 5. Επιπλέον σημειώνεται ότι επιλέγεται ο έκτος κόμβος του συστήματος διότι για δέκα επιμέρους κόμβους η αρίθμηση των κόμβων ξεκινάει από το 0 και τελειώνει στο 9 και η τιμή 5 αναφέρεται στον έκτο κόμβο του συστήματος.

Μόλις προσδιοριστεί ο υπεύθυνος κόμβος για την αποθήκευση ενός αντικειμένου, το αντικείμενο αποθηκεύεται σε αυτόν. Όταν χρειάζεται να προσπελαστεί το αντικείμενο σε μία μελλοντική λειτουργία ανάγνωσης, ακολουθείται ακριβώς η ίδια διαδικασία, όπου εφαρμόζεται η συνάρτηση κατακερματισμού στο κλειδί του αντικειμένου. Στη συνέχεια υπολογίζεται το υπόλοιπο της ακέραιας διαίρεσης και προσδιορίζεται ο κόμβος που περιέχει το συγκεκριμένο αντικείμενο. Αυτή η προσέγγιση εξασφαλίζει ταχεία ανάκτηση ή προσπέλαση δεδομένων, καθώς η θέση του αντικειμένου προσδιορίζεται άμεσα από το κλειδί του, με ντετερμινιστικό τρόπο, χωρίς να απαιτείται η αναζήτηση του αντικειμένου σε πολλούς κόμβους ή με άλλα λόγια η ανάγκη σάρωσης πολλών κόμβων.

Η μέθοδος Hash Modulo είναι μια απλή και αποδοτική προσέγγιση για την κατανομή και ανάκτηση δεδομένων σε κατανεμημένα συστήματα. Βασίζεται στον κατακερματισμό του κλειδιού ενός αντικειμένου και στον υπολογισμό του κόμβου μέσω της πράξης της ακέραιας διαίρεσης, εξασφαλίζοντας ισορροπημένη κατανομή και γρήγορη ανάκτηση των δεδομένων. Ωστόσο, η μέθοδος αυτή έχει ορισμένους περιορισμούς, ιδιαίτερα σε δυναμικά περιβάλλοντα όπου ο αριθμός των κόμβων αλλάζει, καθώς απαιτεί πλήρη ανακατανομή των δεδομένων στους ξεχωριστούς κόμβους του συστήματος. Η μέθοδος μοιάζει αρκετά με την μέθοδο της στατικής κατανομής (Static Partitioning) που περιγράφεται σε προηγούμενο κεφάλαιο, καθώς και οι δύο βασίζονται σε έναν κανόνα αντιστοίχισης για την κατανομή των δεδομένων στους κόμβους του συστήματος. Ωστόσο, η Hash Modulo συνήθως εξασφαλίζει μια καλύτερη και πιο ομοιόμορφη κατανομή δεδομένων σε σύγκριση με την στατική κατανομή.

Στην στατική κατανομή, η κατανομή των δεδομένων εξαρτάται από έναν προκαθορισμένο κανόνα, ο οποίος μπορεί να βασίζεται σε συγκεκριμένα χαρακτηριστικά των δεδομένων, όπως το κλειδί ή άλλα κριτήρια. Σε περίπτωση που τα χαρακτηριστικά των αντικειμένων δεν είναι κατάλληλα ή τα κριτήρια αντιστοίχισης δεν επιλέγονται σωστά, υπάρχει κίνδυνος να μην επιτευχθεί αποδοτική και ομοιόμορφη κατανομή των δεδομένων. Αντίθετα, η μέθοδος Hash Modulo χρησιμοποιεί μια συνάρτηση κατακερματισμού (hash function). Η συνάρτηση κατακερματισμού είναι σχεδιασμένη ώστε να παράγει στις περισσότερες περιπτώσεις μια πιο ομοιόμορφη κατανομή τιμών.





Εικόνες 3.4.1, 3.4.2, 3.4.3, 3.4.4: Αναπαράσταση εισαγωγής αντικειμένων σε ένα σύστημα αποθήκευσης που αποτελείται από 3 κόμβους και κατανέμει τα αντικείμενα σύμφωνα με την μέθοδο του υπολοίπου διαίρεσης της τιμής κατακερματισμού (Hash Modulo)

3.4.α Πλεονεκτήματα της μεθόδου υπόλοιπου κατακερματισμού

- Απλότητα και ευκολία υλοποίησης: Η μέθοδος είναι σχετικά εύκολη στην υλοποίηση, καθώς βασίζεται σε μία μαθηματική συνάρτηση για την αντιστοίχιση αντικειμένων στους κόμβους του συστήματος.
- Ταχεία προσπέλαση αντικειμένων: Ο κόμβος στον οποίο αποθηκεύεται ένα αντικείμενο προσδιορίζεται άμεσα από το αποτέλεσμα της συνάρτησης κατακερματισμού, και έτσι η προσπέλαση των δεδομένων είναι γρήγορη και αποδοτική.
- Δεν απαιτούνται πρόσθετες δομές ευρετηρίασης: Δεν απαιτείται η χρήση δομών ευρετηρίασης για την καταγραφή των κόμβου αποθήκευσης των αντικειμένων στο σύστημα. Οι κόμβοι υπολογίζονται μέσω της συνάρτησης κατακερματισμού
- Αποδοτικότητα σε σταθερά συστήματα: Όταν ο αριθμός των κόμβων παραμένει σταθερός, η μέθοδος είναι πολύ αποδοτική, καθώς εξασφαλίζει γρήγορη προσπέλαση δεδομένων και την ομοιόμορφη κατανομή τους.

3.4.β Μειονεκτήματα της μεθόδου υπόλοιπου κατακερματισμού

- Δεν εγγυάται ομοιόμορφη κατανομή: Η κατανομή των δεδομένων ενδέχεται να είναι μη ομοιόμορφη. Ωστόσο η ομοιομορφία της κατανομής συνήθως είναι καλύτερη σε σχέση με την μέθοδο της στατικής κατανομής.
- Ανάγκη ανακατανομής σε δυναμικά περιβάλλοντα: Η κατανομή με τη χρήση της μεθόδου του υπόλοιπου κατακερματισμού δεν προσαρμόζεται τόσο εύκολα σε αλλαγές στη σύνθεση του συστήματος και απαιτεί πλήρη ανακατανομή των αντικειμένων.

3.5 Δυναμικός κατακερματισμός (Dynamic Hashing)

Η μέθοδος του δυναμικού κατακερματισμού (Dynamic Hashing) είναι μια τεχνική που επιτρέπει στο σύστημα να προσαρμόζεται δυναμικά σε αλλαγές που αφορούν τη δομή του, όπως η προσθήκη και η αφαίρεση κόμβων στο σύστημα. Σε αντίθεση με άλλες μεθόδους, όπως η μέθοδος της στατικής κατανομής ή της κυκλικής κατανομής, οι οποίες απαιτούν πλήρη ανακατανομή των δεδομένων όταν αλλάζει ο αριθμός των κόμβων, ο δυναμικός κατακερματισμός ελαχιστοποιεί την ανακατανομή που απαιτείται, μεταφέροντας μόνο τα δεδομένα που επηρεάζονται άμεσα από την αλλαγή που έχει πραγματοποιηθεί. Αυτή η προσέγγιση είναι ιδιαίτερα χρήσιμη σε δυναμικά περιβάλλοντα, όπου οι αλλαγές στον αριθμό των κόμβων και στη σύνθεση του συστήματος είναι συχνές.

Πιο αναλυτικά, τα δεδομένα που απαιτείται να αποθηκευτούν στο σύστημα κατανέμονται στους κόμβους του με βάση τη συνάρτηση κατακερματισμού όπως ακριβώς συμβαίνει στη μέθοδο κατανομής με βάση το υπόλοιπο της τιμής κατακερματισμού των αντικειμένων που περιγράφεται σε προηγούμενο κεφάλαιο. Κάθε κόμβος είναι υπεύθυνος για συγκεκριμένα αντικείμενα βάσει της τιμής κατακερματισμού τους. Η κατανομή αυτή καθορίζει σε ποιον κόμβο θα αποθηκευτεί το κάθε αντικείμενο, εξασφαλίζοντας μία σχετικά ομοιόμορφη κατανομή των αντικειμένων στους κόμβους του συστήματος.

Ο δυναμικός κατακερματισμός βασίζεται στην ιδέα της δυναμικής διαχείρισης της κατανομής των δεδομένων. Όταν προστίθεται ή αφαιρείται ένας κόμβος, το σύστημα δεν ανακατανομίζει όλα τα δεδομένα, αλλά επικεντρώνεται μόνο στα δεδομένα που επηρεάζονται άμεσα από την αλλαγή στον αριθμό των κόμβων. Αυτό επιτυγχάνεται μέσω δύο βασικών διαδικασιών του διαχωρισμού των δεδομένων ή αλλιώς split και της συγχώνευσης των δεδομένων ή αλλιώς merge.

Όταν προστίθεται ένας νέος κόμβος, το σύστημα δεν χρειάζεται να ανακατανομίζει όλα τα δεδομένα. Αντίθετα, ένα τμήμα των δεδομένων που ανήκει σε έναν από τους υπάρχοντες κόμβους διαχωρίζεται και μεταφέρεται στον νέο κόμβο. Αυτή η διαδικασία, εξασφαλίζει ότι η πλειονότητα των δεδομένων παραμένει σταθερή στον κόμβο που έχουν αρχικά αποθηκευτεί και μόνο ένα τμήμα τους μετακινείται στο νέο κόμβο. Αντίστοιχα όταν ένας κόμβος αφαιρείται από το σύστημα, τα αντικείμενα που έχει αποθηκευμένα ενώνονται με αυτά ενός γειτονικού κόμβου. Αυτό σημαίνει ότι τα δεδομένα που ανήκαν στον κόμβο που αφαιρέθηκε μεταφέρονται σε έναν άλλο υπάρχοντα κόμβο. Αυτή η διαδικασία είναι γνωστή και ως συγχώνευση ή merge.

Με τον τρόπο αυτό το σύστημα προσαρμόζεται στην αύξηση ή τη μείωση του αριθμού των κόμβων που υπάρχει κατά την προσθήκη ή αφαίρεση από το σύστημα αποθήκευσης χωρίς να απαιτείται η ανακατανομή όλων των αποθηκευμένων αντικειμένων. Αντίθετα μόνο ένα τμήμα των δεδομένων χρειάζεται να μεταφερθεί σε κάποιον άλλο κόμβο.

Ο δυναμικός κατακερματισμός, μέσω των διαδικασιών διαχωρισμού και συγχώνευσης, επιτρέπει στο σύστημα να προσαρμόζεται δυναμικά σε αλλαγές στον αριθμό των κόμβων, διατηρώντας παράλληλα μία ισορροπία ανάμεσα στην αποδοτικότητα και στην ομοιομορφία κατανομής των δεδομένων. Αυτή η προσέγγιση είναι ιδιαίτερα χρήσιμη σε δυναμικά περιβάλλοντα, όπου οι αλλαγές στη δομή του συστήματος είναι συχνές. Ωστόσο, η μέθοδος αυτή είναι πιο περίπλοκη στην υλοποίηση σε σύγκριση με άλλες τεχνικές, όπως η στατική

κατανομή ή η κυκλική κατανομή. Οι διαδικασίες του διαχωρισμού και συγχώνευσης των δεδομένων απαιτούν προσεκτικό σχεδιασμό. Αυτή η πολυπλοκότητα μπορεί να αυξήσει το υπολογιστικό φόρτο και τις απαιτήσεις σε πόρους στους κόμβους του συστήματος, ιδιαίτερα σε συστήματα μεγάλης κλίμακας. Παράλληλα τα πλεονεκτήματα της δυναμικής προσαρμογής και της ελαχιστοποίησης της ανακατανομής κάνουν την μέθοδο ιδανική για εφαρμογές όπου οι αλλαγές στον αριθμό των κόμβων είναι συχνές.

3.5.α Πλεονεκτήματα του δυναμικού κατακερματισμού

- Προσαρμοστικότητα: Προσαρμόζεται δυναμικά στις αλλαγές του αριθμού των κόμβων, επιτρέποντας την προσθήκη ή αφαίρεση κόμβων χωρίς να υπάρχει ανάγκη για πλήρη ανακατανομή των αποθηκευμένων δεδομένων.
- Μειωμένη μετακίνηση δεδομένων: Σε περιπτώσεις αλλαγών ή μεταβολών στο σύστημα, μόνο ένα τμήμα των αντικειμένων μετακινείται κατά την προσθήκη ή αφαίρεση κόμβων.
- Μεγαλύτερη αποδοτικότητα σε δυναμικά περιβάλλοντα: Όταν ο αριθμός των κόμβων μεταβάλλεται, η μέθοδος είναι πιο αποδοτική σε σχέση με άλλες, καθώς προσαρμόζεται με μεγαλύτερη ευκολία σε αλλαγές στη σύνθεση του συστήματος, καθιστώντας την πιο ιδανική για δυναμικά περιβάλλοντα.

3.5.β Μειονεκτήματα του δυναμικού κατακερματισμού

- Αυξημένη πολυπλοκότητα υλοποίησης: Απαιτεί πιο σύνθετους αλγορίθμους για τη διαχείριση των δεδομένων, σε σύγκριση με άλλες μεθόδους.
- Ανάγκη ανακατανομής σε δυναμικά περιβάλλοντα: Η κατανομή με τη χρήση της μεθόδου του υπόλοιπου κατακερματισμού δεν προσαρμόζεται τόσο εύκολα σε αλλαγές στη σύνθεση του συστήματος και απαιτεί πλήρη ανακατανομή των αντικειμένων.

ΚΕΦΑΛΑΙΟ 4 Συναρτήσεις κατακερματισμού

4.1 Εισαγωγή

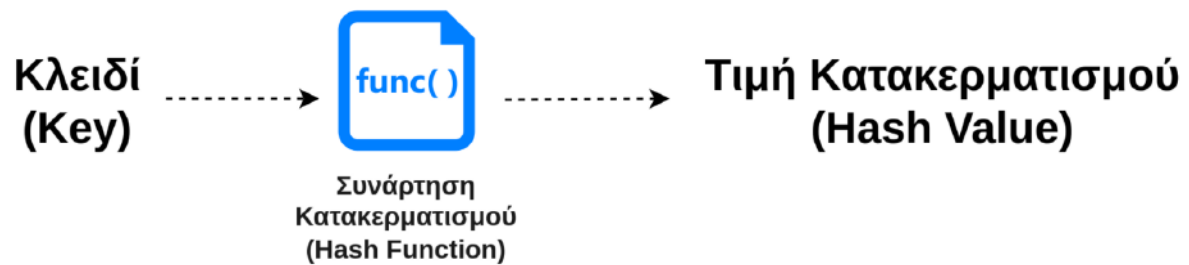
Ο κατακερματισμός (hashing) αποτελεί μία από τις πιο θεμελιώδεις και διαδεδομένες τεχνικές στην επιστήμη των υπολογιστών, με εφαρμογές σε πολλούς τομείς, όπως η κρυπτογραφία, η αποθήκευση και ανάκτηση δεδομένων, οι αλγόριθμοι αναζήτησης και οι δομές δεδομένων. Βασίζεται στη χρήση μιας μαθηματικής συνάρτησης, γνωστής ως συνάρτηση κατακερματισμού (hash function), η οποία δέχεται ως είσοδο δεδομένα οποιουδήποτε μεγέθους και παράγει μια σταθερού μεγέθους έξοδο, που αποκαλείται τιμή κατακερματισμού hash value ή digest.

Η διαδικασία αυτή προσφέρει σημαντικά πλεονεκτήματα, καθώς επιτρέπει τη γρήγορη αποθήκευση και ανάκτηση πληροφοριών, την ασφαλή αποθήκευση κωδικών πρόσβασης, την ανίχνευση αλλοιώσεων σε δεδομένα ή αρχεία και τη δημιουργία μοναδικών ψηφιακών υπογραφών. Ο κατακερματισμός αποτελεί τη βάση για πολλές σύγχρονες τεχνολογίες και αλγόριθμους σε πολλούς κλάδους της πληροφορικής.

Οι συναρτήσεις κατακερματισμού σχεδιάζονται με τέτοιο τρόπο ώστε να είναι ντετερμινιστικές, δηλαδή με άλλα λόγια να παράγουν πάντα την ίδια έξοδο για την μία συγκεκριμένη είσοδο, και να κατανέμουν τις τιμές τους με τρόπο που να αποφεύγονται συγκρούσεις. Αυτό σημαίνει ότι δύο διαφορετικές τιμές εισόδου δεν είναι επιθυμητό να παράγουν το ίδιο αποτέλεσμα ή την ίδια τιμή κατακερματισμού. Η αποδοτικότητα και η ασφάλεια μιας συνάρτησης κατακερματισμού είναι κρίσιμοι παράγοντες που καθορίζουν τη χρήση της σε διάφορες εφαρμογές, από την επιτάχυνση αναζητήσεων σε αρχεία, βάσεις δεδομένων μέχρι την κρυπτογράφηση δεδομένων και αρχείων.

4.2 Ορισμός και βασικές αρχές του κατακερματισμού

Ο κατακερματισμός είναι η διαδικασία μετατροπής δεδομένων με μή προκαθορισμένο ή μεταβλητό μήκου σε μία τιμή αποτελέσματος σταθερού μήκους, η οποία είναι συνήθως ένας ακέραιος αριθμός και αναπαρίσταται σε δεκαεξαδική μορφή. Η τιμή αυτή, γνωστή ως τιμή κατακερματισμού (hash value), αντιπροσωπεύει τα δεδομένα με τρόπο που επιτρέπει τη γρήγορη και αποδοτική σύγκριση τους. Ο κύριος στόχος αυτής της διαδικασίας είναι η δημιουργία μιας μοναδικής ταυτότητας για κάθε δεδομένο που δίνεται ως είσοδος στη συνάρτηση κατακερματισμού. Παρόλο που ο στόχος της συνάρτησης κατακερματισμού είναι η μοναδικότητα, σε πολύ σπάνιες περιπτώσεις μπορεί να παραχθεί το ίδιο αποτέλεσμα από διαφορετικές εισόδους, ένα φαινόμενο που ονομάζεται σύγκρουση (collision).



Εικόνα 4.1: Αναπαράσταση της έννοιας της συνάρτησης κατακερματισμού (Hash Function), η οποία μετασχηματίζει δεδομένα μεταβλητού μήκους σε μια μοναδική τιμή σταθερού μήκους.

4.3 Χρήσεις της συνάρτησης κατακερματισμού

Ο κατακερματισμός έχει εφαρμογές σε πολλούς τομείς, ιδιαίτερα στην κρυπτογραφία, την ασφάλεια δεδομένων και τη διαχείριση πληροφοριών. Παρακάτω παρουσιάζονται συνοπτικά μερικές από τις κύριες χρήσεις των συναρτήσεων κατακερματισμού:

4.3.α Κρυπτογραφία και ασφάλεια δεδομένων

Η χρήση του κατακερματισμού χρησιμοποιείται συχνά και έχει κρίσιμο ρόλο στην προστασία και στον έλεγχο ακεραιότητας των δεδομένων. Όταν εφαρμόζεται μια συνάρτηση κατακερματισμού σε δεδομένα, δημιουργείται ένα μοναδικό αποτέλεσμα ή τιμή, η οποία είναι πρακτικά αδύνατο να αντιστραφεί για την ανάκτηση των αρχικών δεδομένων που έχουν δοθεί ως είσοδος στη συνάρτηση. Αυτή η ιδιότητα καθιστά τον κατακερματισμό ιδανικό για τις εξής περιπτώσεις:

- Έλεγχος ακεραιότητας δεδομένων: Οι τιμές του αποτελέσματος του κατακερματισμού χρησιμοποιούνται για την ανίχνευση τυχόν αλλοιώσεων ή αλλαγών σε δεδομένα όπως για παράδειγμα αρχεία ή μηνύματα. Αν αλλάξουν τα αρχικά δεδομένα τότε αλλάζει και η τιμή κατακερματισμού των δεδομένων, γεγονός που σημαίνει ότι τα αρχικά δεδομένα έχουν τροποποιηθεί. Αυτό είναι ιδιαίτερα χρήσιμο σε περιπτώσεις που χρειάζεται να γίνει έλεγχος για την τροποποίηση κάποιων δεδομένων. Με τη χρήση του κατακερματισμού δίνεται η δυνατότητα να αποθηκευτεί η τιμή κατακερματισμού κάποιου αρχείου κατά τη δημιουργία του και να αποθηκευτεί. Στη συνέχεια για να πραγματοποιηθεί έλεγχος ότι τα δεδομένα δεν έχουν αλλοιωθεί ή τροποποιηθεί μελλοντικά, παράγεται εκ νέου η τιμή κατακερματισμού του αρχείου και συγκρίνεται με την αρχική τιμή. Σε περίπτωση που οι δύο τιμές είναι ίδιες σημαίνει ότι τα δεδομένα δεν έχουν αλλάξει ενώ σε αντίθετη περίπτωση έχει υπάρξει κάποια τροποποίηση και δεν είναι τα ίδια.
- Διαχείριση κωδικών πρόσβασης: Για λόγους ασφαλείας, οι κωδικοί πρόσβασης χρηστών μπορούν να αποθηκεύονται ως το αποτέλεσμα μιας συνάρτησης κατακερματισμού των αρχικών κωδικών. Αυτό σημαίνει ότι αντί να αποθηκεύονται οι κωδικοί πρόσβασης άμεσα σε αναγνώσιμη μορφή, μετατρέπονται σε μία μοναδική σειρά χαρακτήρων που δεν μπορούν να αντιστραφούν εύκολα και να βρεθούν οι αρχικοί κωδικοί πρόσβασης. Έτσι, ακόμη και αν κάποιος αποκτήσει πρόσβαση στην αποθήκη δεδομένων που περιέχει κωδικούς πρόσβασης, δεν θα μπορέσει να χρησιμοποιήσει τα αποθηκευμένα δεδομένα ή να ανακτήσει ή τους κωδικούς πρόσβασης, αφού αυτοί είναι αποθηκευμένοι σε μορφή που δεν μπορεί να αντιστραφεί. Αυτή η διαδικασία προσφέρει ένα επιπλέον επίπεδο προστασίας για την αποθήκευση δεδομένων.
- Ψηφιακές υπογραφές και ταυτοποίηση: Οι συναρτήσεις κατακερματισμού χρησιμοποιούνται για τη δημιουργία ψηφιακών υπογραφών, οι οποίες διασφαλίζουν ότι ένα έγγραφο ή αρχείο δεν έχει τροποποιηθεί και επιβεβαιώνουν την αυθεντικότητα του αποστολέα. Όταν ένα έγγραφο ή αρχείο υπογράφεται ψηφιακά, η συνάρτηση

κατακερματισμού παράγει μια μοναδική τιμή για το περιεχόμενο του, και αυτή η τιμή χρησιμοποιείται στη συνέχεια για να δημιουργηθεί η ψηφιακή υπογραφή. Αν το έγγραφο ή το αρχείο αλλάξει με οποιονδήποτε τρόπο, η τιμή κατακερματισμού του θα αλλάξει επίσης, και συνεπώς η υπογραφή δεν θα είναι έγκυρη. Με αυτόν τον τρόπο, οι ψηφιακές υπογραφές εξασφαλίζουν ότι τα δεδομένα παραμένουν αμετάβλητα και ότι προέρχονται από τον αναγνωρισμένο αποστολέα.

4.3.β Αποθήκευση και αναζήτηση δεδομένων

Η τεχνική του κατακερματισμού (hashing) χρησιμοποιείται συχνά και είναι μια από τις πιο βασικές μεθόδους στην επιστήμη των υπολογιστών, ιδιαίτερα όταν πρόκειται για ζητήματα αποθήκευσης και ανάκτησης δεδομένων και πληροφοριών με γρήγορο και αποδοτικό τρόπο. Βασίζεται στη χρήση μιας συνάρτησης κατακερματισμού (hash function), η οποία μετατρέπει τα δεδομένα εισόδου, γνωστά ως κλειδιά (keys), σε μοναδικές τιμές, γνωστές ως τιμές ή αποτελέσματα κατακερματισμού (hash values). Αυτές οι τιμές χρησιμοποιούνται στη συνέχεια ως δείκτες (indices) ή αλλιώς ευρετήρια για την αποθήκευση και την ανάκτηση των δεδομένων. Μία από τις πιο γνωστές και διαδεδομένες εφαρμογές του κατακερματισμού είναι οι πίνακες κατακερματισμού (hash tables), οι οποίοι αξιοποιούν αυτή τη διαδικασία για την αποδοτική οργάνωση και προσπέλαση δεδομένων και πληροφοριών.

Η βασική ιδέα στην οποία βασίζονται οι πίνακες κατακερματισμού είναι η δημιουργία μιας απεικόνισης (mapping) μεταξύ των κλειδιών (keys) και των θέσεων στον πίνακα. Όταν ένα κλειδί εισάγεται στη συνάρτηση κατακερματισμού, αυτή παράγει μια τιμή κατακερματισμού, η οποία καθορίζει τη θέση στον πίνακα κατακερματισμού όπου θα αποθηκευτεί ή θα αναζητηθεί το αντίστοιχο αντικείμενο. Αυτή η διαδικασία επιτρέπει την απευθείας πρόσβαση στα δεδομένα, χωρίς την ανάγκη για σειριακή αναζήτηση ή σάρωση, γεγονός που καθιστά τους πίνακες κατακερματισμού ιδανικούς για εφαρμογές που απαιτούν υψηλή ταχύτητα προσπέλασης στα αποθηκευμένα δεδομένα.

Ο κατακερματισμός είναι ιδιαίτερα αποτελεσματικός σε εφαρμογές που απαιτούν ταχύτητα και ευελιξία, προσφέροντας άμεση αναζήτηση, εισαγωγή και διαγραφή δεδομένων. Η ταχύτητα προσπέλασης και πρόσβασης στα δεδομένα είναι ιδιαίτερα σημαντική σε εφαρμογές όπου απαιτείται άμεση ανάκτηση πληροφοριών, όπως για παράδειγμα σε βάσεις δεδομένων, συστήματα αρχείων ή συστήματα προσωρινής αποθήκευσης δεδομένων. Επιπλέον, ο κατακερματισμός διακρίνεται για την ευελιξία του, καθώς μπορεί να εφαρμοστεί σε πολλές περιπτώσεις και σε διαφορετικά είδη δεδομένων. Χρησιμοποιείται για την αποθήκευση και ανάκτηση αριθμών, συμβολοσειρών, καθώς και πιο σύνθετων πληροφοριών, όπως αντικείμενα ή σύνθετες δομές δεδομένων. Αυτή η προσαρμοστικότητα και ευελιξία του τον καθιστά ιδανικό για πλήθος εφαρμογών και εξειδικευμένων περιπτώσεων χρήσης.

Η μέθοδος του κατακερματισμού χρησιμοποιείται σε αρκετές εφαρμογές, όπως για παράδειγμα:

- Σε βάσεις δεδομένων (Database Systems): Οι πίνακες κατακερματισμού χρησιμοποιούνται για την ταχεία ευρετηρίαση και προσπέλαση δεδομένων σε συστήματα βάσεων δεδομένων. Για παράδειγμα, όταν σε ένα σύστημα μιας βάσης δεδομένων χρειάζεται να βρεθεί και να προσπελαστεί γρήγορα μια συγκεκριμένη

εγγραφή, χρησιμοποιείται μια συνάρτηση κατακερματισμού για να καθορίσει την ακριβή θέση που περιέχει την εγγραφή στον πίνακα, μειώνοντας σημαντικά τον χρόνο αναζήτησης των δεδομένων.

- Σε συστήματα προσωρινής αποθήκευσης (Caching Systems): Σε συστήματα προσωρινής αποθήκευσης δεδομένων, όπως αυτά που χρησιμοποιούνται σε ιστοσελίδες ή διαδικτυακές εφαρμογές, οι πίνακες κατακερματισμού βοηθούν στην προσωρινή αποθήκευση δεδομένων για γρήγορη πρόσβαση. Για παράδειγμα, όταν ένας χρήστης ζητά συχνά χρησιμοποιούμενα δεδομένα, αυτά μπορούν να αποθηκευτούν προσωρινά σε έναν πίνακα κατακερματισμού για να εξυπηρετηθούν γρήγορα χωρίς να χρειάζεται να γίνει εκ νέου ανάκτηση του απο το σύστημα αρχείων από την βάση δεδομένων.

4.4 Ιδιότητες της συνάρτησης κατακερματισμού

Για να θεωρηθεί μια συνάρτηση κατακερματισμού ισχυρή και αξιόπιστη, πρέπει να διαθέτει μια σειρά ή ένα σύνολο από ιδιότητες που εξασφαλίζουν τόσο την αποδοτικότητα όσο και την αξιοπιστία της. Σε αυτή την ενότητα, θα εξετάσουμε τις βασικές ιδιότητες που πρέπει να έχει μια συνάρτηση κατακερματισμού, καθώς και επιπλέον χαρακτηριστικά που την καθιστούν κατάλληλη τη χρήση της σε εφαρμογές κρυπτογραφίας και αποθήκευση δεδομένων.

4.4.α Ντετερμινιστικότητα (Determinism)

Μια συνάρτηση κατακερματισμού πρέπει να είναι ντετερμινιστική, δηλαδή να παράγει το ίδιο αποτέλεσμα ή τιμή κατακερματισμού για την ίδια τιμή εισόδου κάθε φορά που εφαρμόζεται. Αυτή η ιδιότητα είναι κρίσιμη για την εξασφάλιση της ταυτοποίησης των δεδομένων, της ακεραιότητάς τους και της επαλήθευσης σε εφαρμογές κρυπτογράφησης και ασφάλειας, όπως η επαλήθευση δεδομένων και η ψηφιακή υπογραφή. Επιπλέον, σε εφαρμογές αποθήκευσης δεδομένων, αν η συνάρτηση δεν ήταν ντετερμινιστική, η ανάκτηση των δεδομένων με βάση ένα κλειδί δεν θα ήταν δυνατή, καθώς δεν θα υπήρχε σταθερή σχέση μεταξύ του κλειδιού και της θέσης αποθήκευσης των δεδομένων.

4.4.β Αντίσταση σε συγκρούσεις (Collision Resistance)

Η αντιστάθμιση συγκρούσεων είναι μια από τις πιο σημαντικές ιδιότητες μιας συνάρτησης κατακερματισμού. Μια σύγκρουση συμβαίνει όταν δύο διαφορετικά κλειδιά ή τιμές εισόδου παράγουν το ίδιο αποτέλεσμα ή την ίδια τιμή κατακερματισμού. Η συνάρτηση πρέπει να είναι σχεδιασμένη έτσι ώστε να είναι υπολογιστικά δύσκολο να βρεθούν δύο τέτοια διαφορετικά κλειδιά. Αυτή η ιδιότητα είναι κρίσιμη για την αποφυγή προβλημάτων σε εφαρμογές όπως οι ψηφιακές υπογραφές, τα συστήματα ασφαλείας και η επαλήθευση ακεραιότητας δεδομένων.

4.4.γ Αντίσταση σε προεικόνα (Preimage Resistance)

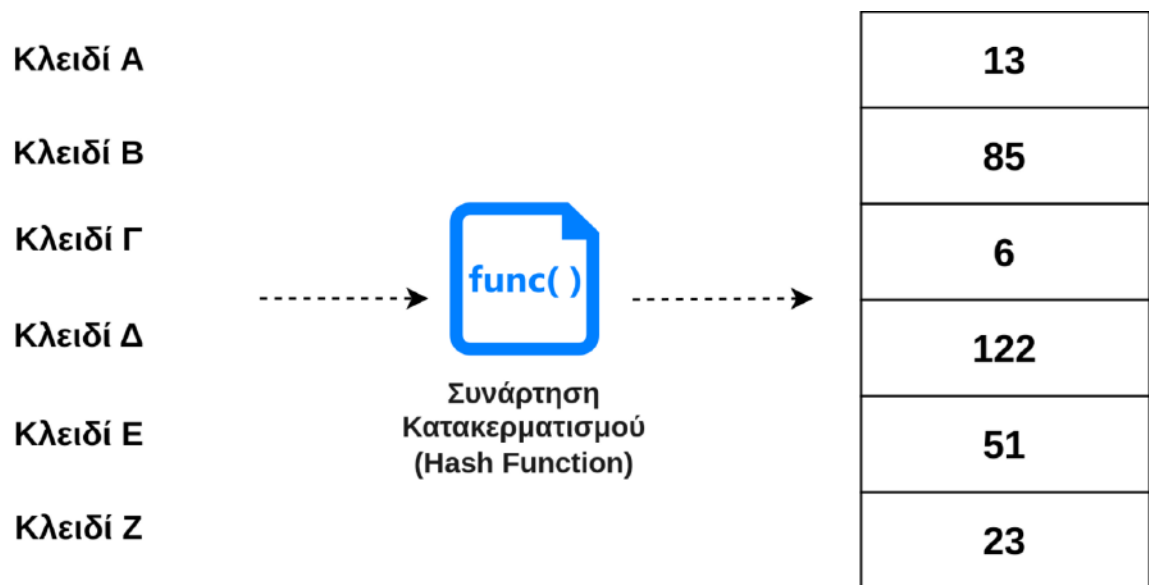
Η αντίσταση σε προεικόνα σημαίνει ότι, δεδομένης μιας τιμής κατακερματισμού, πρέπει να είναι υπολογιστικά δύσκολο να βρεθεί η αρχική είσοδος που δόθηκε ως είσοδος για την παραγωγή της. Με άλλα λόγια, η συνάρτηση πρέπει να είναι μονόδρομη ώστε να μην μπορεί κανείς να ανακτήσει τα αρχικά δεδομένα από την τιμή κατακερματισμού. Αυτή η ιδιότητα είναι ιδιαίτερα σημαντική για την προστασία των δεδομένων, ειδικά σε εφαρμογές κρυπτογραφίας, όπως η αποθήκευση κωδικών πρόσβασης και η κρυπτογράφηση αρχείων και δεδομένων.

4.4.δ Ομοιομορφία στην κατανομή (Uniform Distribution)

Οι τιμές κατακερματισμού που παράγονται από μια συνάρτηση κατακερματισμού πρέπει να είναι ομοιόμορφα κατανεμημένες στον χώρο εξόδου ή αλλιώς στο χώρο κατακερματισμού. Αυτό σημαίνει ότι κάθε τιμή στον χώρο κατακερματισμού πρέπει να έχει την ίδια πιθανότητα να επιλεγεί, ανεξαρτήτως της κατανομής των εισερχόμενων κλειδιών στην συνάρτηση κατακερματισμού. Η ομοιόμορφη κατανομή είναι ιδιαίτερα σημαντική ιδιαίτερα σε περιπτώσεις που χρησιμοποιείται σε εφαρμογές για αποθήκευση και αναζήτηση δεδομένων. Αυτό το γεγονός είναι απαραίτητο για τη μείωση των συγκρούσεων και τη βελτίωση της απόδοσης, καθώς εξασφαλίζει ότι τα δεδομένα και τα αντικείμενα κατανέμονται ομοιόμορφα στους κόμβους αποθήκευσης, αποτρέποντας περιοχές υπερφόρτωσης. Επιπλέον, αυτή η ιδιότητα είναι σημαντική για την αποδοτική κατανομή φόρτου σε κατανεμημένα συστήματα.

4.4.ε Απόδοση (Efficiency)

Μια συνάρτηση κατακερματισμού πρέπει να είναι γρήγορη και αποδοτική στην εκτέλεσή της, καθώς η ταχύτητα εκτέλεσης που απαιτείται είναι σημαντική για πολλές εφαρμογές που απαιτούν υψηλή απόδοση. Η ικανότητα της συνάρτησης να εκτελείται γρήγορα είναι ιδιαίτερα σημαντική σε συστήματα που χειρίζονται μεγάλες ποσότητες δεδομένων, όπως βάσεις δεδομένων, συστήματα προσωρινής αποθήκευσης και εφαρμογές κρυπτογραφίας. Σε αυτές τις περιπτώσεις, οι λειτουργίες κατακερματισμού χρειάζεται να πραγματοποιούνται συχνά σε πραγματικό χρόνο και δεν πρέπει να υπάρχει καθυστέρηση. Για παράδειγμα, σε βάσεις δεδομένων, η ταχύτητα του κατακερματισμού επηρεάζει άμεσα την ταχύτητα της αναζήτησης και της αποθήκευσης δεδομένων, καθώς οι τιμές κατακερματισμού χρησιμοποιούνται για την προσπέλαση των δεδομένων. Αντίστοιχα σε εφαρμογές κρυπτογράφησης, η ταχύτητα εκτέλεσης μιας συνάρτησης κατακερματισμού είναι απαραίτητη για την αποτελεσματική εκτέλεση αλγορίθμων ασφαλείας, όπως η παραγωγή μιας ψηφιακής υπογραφής καθώς και η κρυπτογράφηση ή επαλήθευση ενός κωδικού πρόσβασης σε ένα σύστημα.



Εικόνα 4.4.1: Οπτική αναπαράσταση της λειτουργίας μιας συνάρτησης κατακερματισμού (Hash Function). Τα δεδομένα εισόδου υποβάλλονται σε επεξεργασία μέσω της συνάρτησης, παράγοντας μοναδικές τιμές κατακερματισμού (hash values), οι οποίες μπορούν να χρησιμοποιηθούν για γρήγορη αναζήτηση, αποθήκευση και επαλήθευση ακεραιότητας δεδομένων.

ΚΕΦΑΛΑΙΟ 5 Σχεδιασμός του συστήματος αποθήκευσης

Σε αυτό το κεφάλαιο, παρουσιάζεται η ανάπτυξη ενός κατανεμημένου συστήματος αποθήκευσης που υποστηρίζει την αποθήκευση δεδομένων σε πολλαπλούς αυτόνομους κόμβους, εστιάζοντας στη συνεργατική αποθήκευση των δεδομένων στις παρυφές του δικτύου. Ο βασικός στόχος του συστήματος είναι η κατανομή των δεδομένων στους επιμέρους κόμβους του, επιδιώκοντας με τον τρόπο αυτό την αποδοτικότερη αξιοποίηση του αποθηκευτικού χώρου, την εξισορρόπηση του φορτίου και την αύξηση της διαθεσιμότητας των αποθηκευμένων δεδομένων.

Πιό συγκεκριμένα, αναλύονται οι προδιαγραφές και τα ζητούμενα του συστήματος, η επιλογή της κατάλληλης μεθόδου κατανομής των δεδομένων, καθώς και η τελική υλοποίηση του συστήματος, εστιάζοντας στις αρχιτεκτονικές επιλογές και τους μηχανισμούς διαχείρισης της κατανεμημένης αποθήκευσης.

Πιό συγκεκριμένα, στο κεφάλαιο αυτό γίνεται ανάλυση των προδιαγραφών και των απαιτήσεων του συστήματος, η επιλογή της καταλληλότερης μεθόδου που θα χρησιμοποιηθεί για την κατανομή των δεδομένων, καθώς και η τελική υλοποίηση του συστήματος αποθήκευσης. Επιπλέον, εξετάζονται οι αρχιτεκτονικές επιλογές όπως επίσης οι μηχανισμοί διαχείρισης της κατανεμημένης αποθήκευσης και οι τεχνικές που εφαρμόστηκαν.

5.1 Προδιαγραφές και ζητούμενα του συστήματος

Στην ενότητα αυτή καθορίζονται οι βασικές προδιαγραφές και τα ζητούμενα του κατανεμημένου συστήματος αποθήκευσης δεδομένων, τα οποία προκύπτουν από τους στόχους της εργασίας. Για να διασφαλιστεί η αποτελεσματική και αξιόπιστη λειτουργία του συστήματος σε ένα περιβάλλον με πολλαπλούς αυτόνομους κόμβους, είναι απαραίτητο να πληρούνται συγκεκριμένες απαιτήσεις και προδιαγραφές.

Οι βασικές προδιαγραφές και ζητούμενα του συστήματος που θα αναπτυχθεί στην διπλωματική εργασία περιλαμβάνουν τα εξής:

1. Κατανεμημένη αποθήκευση δεδομένων: Το σύστημα πρέπει να υποστηρίζει την κατανομή των αντικειμένων που χρειάζεται να αποθηκευτούν σε πολλαπλούς κόμβους. Κάθε κόμβος πρέπει να είναι σε θέση να αποθηκεύει και να διαχειρίζεται ένα μέρος ή τμήμα των αντικειμένων, χωρίς να υπάρχει κεντρικό σημείο αποτυχίας.
2. Ομοιόμορφη κατανομή δεδομένων: Το σύστημα πρέπει να εξασφαλίζει μια ομοιόμορφη κατανομή των δεδομένων και αντικειμένων στους διαθέσιμους κόμβους του, ώστε να αποφεύγεται η συγκέντρωση δεδομένων σε συγκεκριμένους κόμβους και να μην δημιουργούνται σημεία συμφόρησης στο σύστημα. Με τον τρόπο αυτό επιτυγχάνεται η αποδοτική χρήση και αξιοποίηση του αποθηκευτικού χώρου.

3. Διασφάλιση διαθεσιμότητας και αξιοπιστία: Το σύστημα πρέπει να εγγυάται υψηλή διαθεσιμότητα των δεδομένων και λειτουργία του συστήματος, ακόμα και σε περιπτώσεις αποτυχίας ενός ή περισσότερων κόμβων του. Πρέπει να έχει ανοχή σε σφάλματα και να μην υπάρχει κίνδυνος απώλειας δεδομένων σε κάποια πιθανή περίπτωση βλάβης ή αποτυχίας ενός κόμβου.
4. Δυνατότητα κλιμάκωσης: Το σύστημα αποθήκευσης πρέπει να είναι σχεδιασμένο με τρόπο που να επιτρέπει την κλιμάκωσή του, ώστε να είναι ικανό να καλύψει την ανάγκη για αύξηση του συνολικού αποθηκευτικού χώρου που διαθέτει. Δηλαδή να μπορεί να επεκτείνεται ομαλά, προσθέτοντας δυναμικά νέους κόμβους χωρίς να επηρεάζεται λειτουργία του.
5. Κατανομημένη λειτουργία του συστήματος: Οι κόμβοι του συστήματος πρέπει να λειτουργούν κατανομημένα, χωρίς την ανάγκη ύπαρξης ενός κεντρικού διαχειριστή. Κάθε κόμβος πρέπει να είναι ικανός να δέχεται και να εξυπηρετεί αιτήματα για προσπέλαση δεδομένων, διασφαλίζοντας ότι η εγγραφή ή η ανάγνωση ενός αντικειμένου μπορεί να πραγματοποιηθεί από οποιονδήποτε κόμβο του δικτύου.
6. Συνεργατική Λειτουργία: Το σύστημα πρέπει να υποστηρίζει τη συνεργατική αποθήκευση και διαχείριση δεδομένων μεταξύ των κόμβων. Οι επιμέρους κόμβοι πρέπει να έχουν τη δυνατότητα επικοινωνίας μεταξύ τους, ώστε να συνεργάζονται για την κατανομή και την διαχείριση των δεδομένων, διασφαλίζοντας την ομαλή λειτουργία του συστήματος.
7. Ευελιξία και προσαρμογή: Το σύστημα πρέπει να είναι ευέλικτο και να προσαρμόζεται δυναμικά σε αλλαγές στη σύνθεση του, όπως η προσθήκη ή η αφαίρεση κόμβων. Η προσαρμογή αυτή πρέπει να γίνεται με τρόπο που να μην επηρεάζεται η λειτουργία του και χωρίς να απαιτείται εκτεταμένη αναδιαμόρφωση ή διακοπή της υπηρεσίας.
8. Αποφυγή πλήρους ανακατανομής δεδομένων: Το σύστημα πρέπει να διαχειρίζεται δυναμικές αλλαγές, όπως η προσθήκη ή η αφαίρεση κόμβων, χωρίς να απαιτείται πλήρης ανακατανομή των αντικειμένων που είναι αποθηκευμένα σε αυτό. Η μετακίνηση ή ανακατανομή δεδομένων πρέπει να περιορίζεται μόνο στο απαραίτητο τμήμα, διασφαλίζοντας την αποδοτική λειτουργία του συστήματος και ελαχιστοποιώντας τον φόρτο στους κόμβους του.

5.2 Επιλογή της κατάλληλης μεθόδου για την κατανεμημένη αποθήκευση

Σύμφωνα με τις προδιαγραφές και τα ζητούμενα που παρουσιάστηκαν στην προηγούμενη ενότητα, η επιλογή της κατάλληλης μεθόδου για την κατανεμημένη αποθήκευση δεδομένων απαιτεί την ανάπτυξη ενός συστήματος που να είναι απλό, αποδοτικό και να μην βασίζεται σε πολύπλοκες δομές ή επιπλέον δεδομένα ευρετηρίασης. Αντίθετα, το σύστημα πρέπει να βασίζεται στην κατανομή των δεδομένων με τη χρήση μιας συνάρτησης κατακερματισμού (hash function), η οποία εξασφαλίζει γρήγορη και αποτελεσματική αντιστοίχιση των δεδομένων στους επιμέρους κόμβους του συστήματος.

Μέθοδοι όπως η κυκλική κατανομή (Round Robin) δεν είναι κατάλληλες για αυτό το σκοπό, καθώς δεν υποστηρίζουν πλήρως κατανεμημένη εκτέλεση του αλγορίθμου κατανομής στους ξεχωριστούς κόμβους. Επιπλέον, τέτοιες μέθοδοι απαιτούν επιπλέον πληροφορία για την ευρετηρίαση των αποθηκευμένων αντικειμένων. Αντίθετα, μια μέθοδος που βασίζεται στον κατακερματισμό επιτρέπει την πλήρως κατανεμημένη λειτουργία του συστήματος, χωρίς την ανάγκη για κεντρικό σημείο ελέγχου.

Για να διασφαλιστεί η υψηλή διαθεσιμότητα των δεδομένων και η ανοχή του συστήματος σε σφάλματα, είναι απαραίτητη η ενσωμάτωση της μεθόδου της αντιγραφής δεδομένων (replication). Με αυτήν την προσέγγιση, κάθε αντικείμενο αποθηκεύεται σε πολλαπλούς κόμβους του συστήματος αποθήκευσης, ώστε σε περίπτωση αποτυχίας ενός ή περισσότερων κόμβων, τα δεδομένα να παραμένουν προσβάσιμα από τους υπόλοιπους. Η αναπαραγωγή των δεδομένων σε περισσότερες από μία τοποθεσίες εγγυάται ότι το σύστημα συνεχίζει να λειτουργεί ακόμα και σε περιπτώσεις μερικής αποτυχίας.

Ο συνδυασμός της κατανομής δεδομένων μέσω συνάρτησης κατακερματισμού με τη τεχνική της αντιγραφής των δεδομένων, επιτρέπει στο σύστημα να διατηρεί μια ισορροπία μεταξύ αποδοτικότητας, αξιοπιστίας και ανθεκτικότητας σε σφάλματα.

Η επιλεγμένη μέθοδος πρέπει να διασφαλίζει επίσης ότι οι αλλαγές στη σύνθεση του συστήματος, όπως η προσθήκη ή η αφαίρεση κόμβων, δεν οδηγούν σε πλήρη ανακατανομή των δεδομένων. Αντίθετα, πρέπει να ελαχιστοποιείται η μετακίνηση των δεδομένων σε περιπτώσεις μεταβολών στους κόμβους του δικτύου.

Επιπλέον, η μέθοδος αποθήκευσης πρέπει να επιτρέπει την ομαλή κλιμάκωση του συστήματος, διασφαλίζοντας ότι η αύξηση των απαιτήσεων αποθήκευσης μπορεί να αντιμετωπιστεί χωρίς διακοπή της λειτουργίας του. Αυτή η ευελιξία είναι κρίσιμη για την υποστήριξη δυναμικών περιβαλλόντων, όπου οι αλλαγές στη σύνθεση των κόμβων είναι συχνές.

5.3 Η μέθοδος του συνεπή κατακερματισμού (Consistent Hashing)

Η μέθοδος του συνεπή κατακερματισμού ή αλλιώς Consistent Hashing είναι μια μέθοδος κατανομής δεδομένων σε κατανεμημένα συστήματα αποθήκευσης. Αυτή η μέθοδος επιλύει αρκετά από τα προβλήματα και τους περιορισμούς που προκύπτουν από απλούστερες προσεγγίσεις, όπως η μέθοδος κατανομής δεδομένων με βάση το υπόλοιπο της τιμής κατακερματισμού τους (hash modulo). Η συγκεκριμένη μέθοδος βασίζεται στην ιδέα της δημιουργίας ενός δακτυλίου κατακερματισμού (hash ring), στον οποίο αντιστοιχίζονται τόσο οι κόμβοι του συστήματος όσο και τα αντικείμενα που αποθηκεύονται. Ο συνεπής κατακερματισμός είναι ιδιαίτερα αποδοτικός σε δυναμικά περιβάλλοντα, όπου οι κόμβοι προστίθενται ή αφαιρούνται συχνά, καθώς ελαχιστοποιεί τις μετακινήσεις των αντικειμένων κατά τη διαδικασία ανακατανομής και εξασφαλίζει ότι μόνο ένα μικρό τμήμα των αντικειμένων απαιτεί μετακίνηση κατά την προσθήκη ή αφαίρεση κόμβων στο σύστημα.

Η βασική ιδέα της μεθόδου του συνεπή κατακερματισμού είναι η δημιουργία ενός κυκλικού δακτυλίου που ονομάζεται δακτύλιος κατακερματισμού ή αλλιώς hash ring . Ο δακτύλιος κατακερματισμού πρόκειται για μια αφηρημένη αναπαράσταση του χώρου κατακερματισμού, όπου οι τιμές κατακερματισμού τοποθετούνται σε έναν κύκλο. Η συνάρτηση κατακερματισμού εφαρμόζεται σε κάθε αντικείμενο που αποθηκεύεται και κάθε κόμβος του συστήματος, και παράγει μία μοναδική τιμή κατακερματισμού που καθορίζει τη θέση τους στον δακτύλιο. Ο δακτύλιος αυτός ξεκινά από την ελάχιστη τιμή και καταλήγει στην μέγιστη τιμή που μπορεί να παραχθεί από τη συνάρτηση κατακερματισμού. Με τον τρόπο αυτό, οι τιμές κατακερματισμού δεν κατανέμονται γραμμικά, αλλά κυκλικά, δηλαδή η μέγιστη τιμή του κατακερματισμού συνδέεται απευθείας με την ελάχιστη τιμή, δημιουργώντας έτσι έναν κύκλο.

Για να πραγματοποιηθεί η αντιστοίχιση των αντικειμένων στους κόμβους του δικτύου, κάθε αντικείμενο αποθηκεύεται στον κόμβο του οποίου η τιμή κατακερματισμού είναι η πλησιέστερη προς την τιμή κατακερματισμού του αντικειμένου, με την προϋπόθεση ότι η τιμή του κόμβου είναι μεγαλύτερη ή ίση από την τιμή του αντικειμένου στον δακτύλιο. Αν δεν υπάρχει κάποιος κόμβος με μεγαλύτερη τιμή, το αντικείμενο τοποθετείται στον πρώτο κόμβο στον δακτύλιο, ακολουθώντας τον κυκλικό χαρακτήρα του κατακερματισμού. Με άλλα λόγια, τα δεδομένα αντιστοιχίζονται στον πρώτο κόμβο που βρίσκεται μετά από την τιμή κατακερματισμού τους στον κυκλικό δακτύλιο κατακερματισμού.

Η μέθοδος του συνεπή κατακερματισμού είναι ιδιαίτερα αποδοτική στη διαχείριση δυναμικών περιβαλλόντων και αλλαγών στη σύνθεση των κόμβων του συστήματος αποθήκευσης, όπως για παράδειγμα η προσθήκη και η αφαίρεση κόμβων. Αυτή η δυναμικότητα επιτυγχάνεται χάρη στη χρήση του δακτυλίου, που διασφαλίζει ότι η ανακατανομή των αποθηκευμένων δεδομένων πραγματοποιείται σε ένα τμήμα των αποθηκευμένων αντικειμένων χωρίς να απαιτείται η πλήρης ανακατανομή τους. Το γεγονός αυτό καθιστά επίσης τη μέθοδο ιδανική για συστήματα που απαιτούν κλιμάκωση, δηλαδή την αύξηση του συνολικού αποθηκευτικού χώρου με την προσθήκη νέων κόμβων στο σύστημα. Παρακάτω εξηγούνται αναλυτικά οι διαδικασίες προσθήκης και αφαίρεσης των κόμβων στο σύστημα αποθήκευσης.

Όταν προστίθεται ένας νέος κόμβος στο σύστημα, υπολογίζεται η τιμή κατακερματισμού του και τοποθετείται στη σωστή θέση στον κυκλικό δακτύλιο. Η τιμή κατακερματισμού καθορίζει τη θέση του στον δακτύλιο και καθορίζει την περιοχή ή το τμήμα του δακτυλίου που περιλαμβάνει τα αντικείμενα για τα οποία είναι υπεύθυνος. Τα δεδομένα που είχαν τιμή κατακερματισμού ίση ή μικρότερη από αυτή του νέου κόμβου, αλλά μεγαλύτερη από την τιμή κατακερματισμού του αμέσως προηγούμενου κόμβου στον δακτύλιο, μεταφέρονται στο νέο κόμβο. Με άλλα λόγια, ο νέος κόμβος αναλαμβάνει τη διαχείριση των δεδομένων που βρίσκονται στην περιοχή μεταξύ της δικής του τιμής κατακερματισμού και της τιμής του προηγούμενου κόμβου. Τα υπόλοιπα δεδομένα παραμένουν στους ήδη υπάρχοντες κόμβους, χωρίς να επηρεάζεται η κατανομή τους.

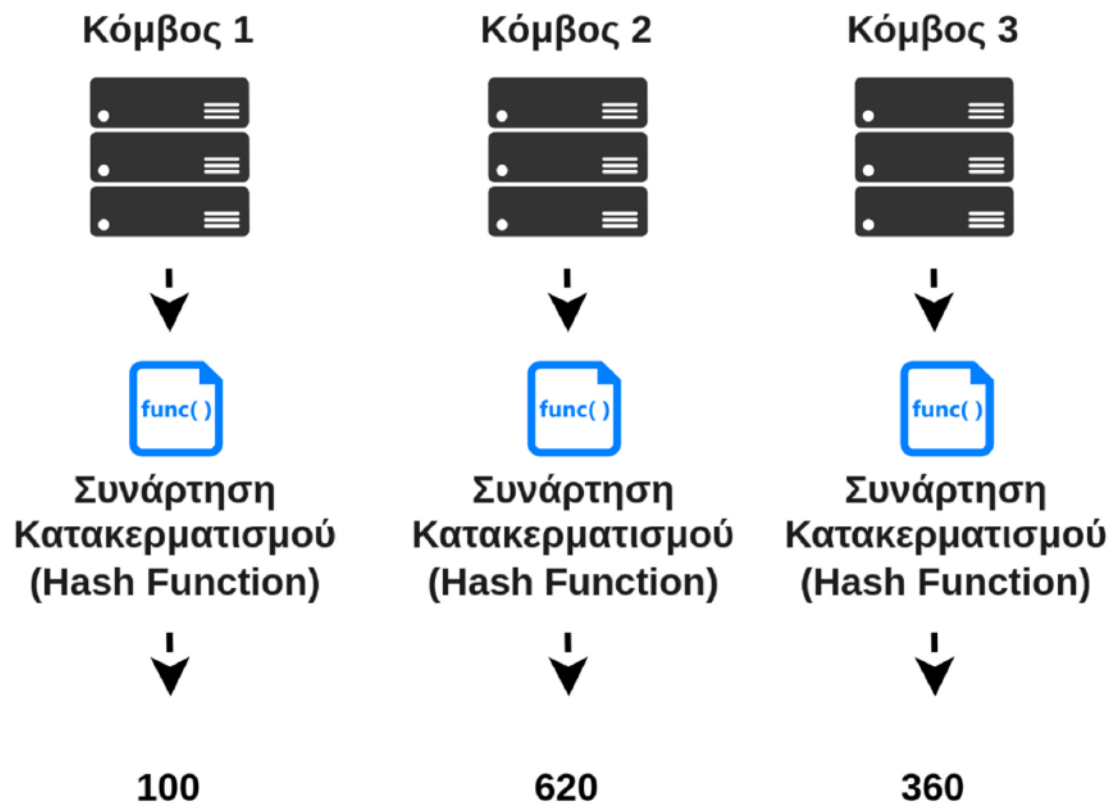
Για παράδειγμα, αν ο νέος κόμβος που προστίθεται έχει τιμή κατακερματισμού 150 και ο προηγούμενος κόμβος στον δακτύλιο έχει τιμή 100, τα δεδομένα με τιμές κατακερματισμού από 101 έως 150 μετακινούνται στον νέο κόμβο. Τα δεδομένα με τιμές έξω από αυτή την περιοχή παραμένουν στους κόμβους στους οποίους ήταν αποθηκευμένα.

Όταν αφαιρείται ένας κόμβος από το σύστημα, η διαδικασία είναι η αντίστροφη. Ο κόμβος που αφαιρείται αναγνωρίζεται και τα αντικείμενα που ήταν αποθηκευμένα σε αυτόν μετακινούνται στον αμέσως επόμενο κόμβο στον κυκλικό δακτύλιο, δηλαδή στον κόμβο με τη μεγαλύτερη τιμή κατακερματισμού από την τιμή του κόμβου που αφαιρείται. Αυτή η διαδικασία εξασφαλίζει ότι τα δεδομένα που ήταν αποθηκευμένα στον κόμβο που αφαιρείται αντιστοιχίζονται στον επόμενο διαθέσιμο κόμβο, όπου μετακινούνται και αποθηκεύονται, χωρίς να επηρεάζεται η υπόλοιπη κατανομή των δεδομένων στο σύστημα.

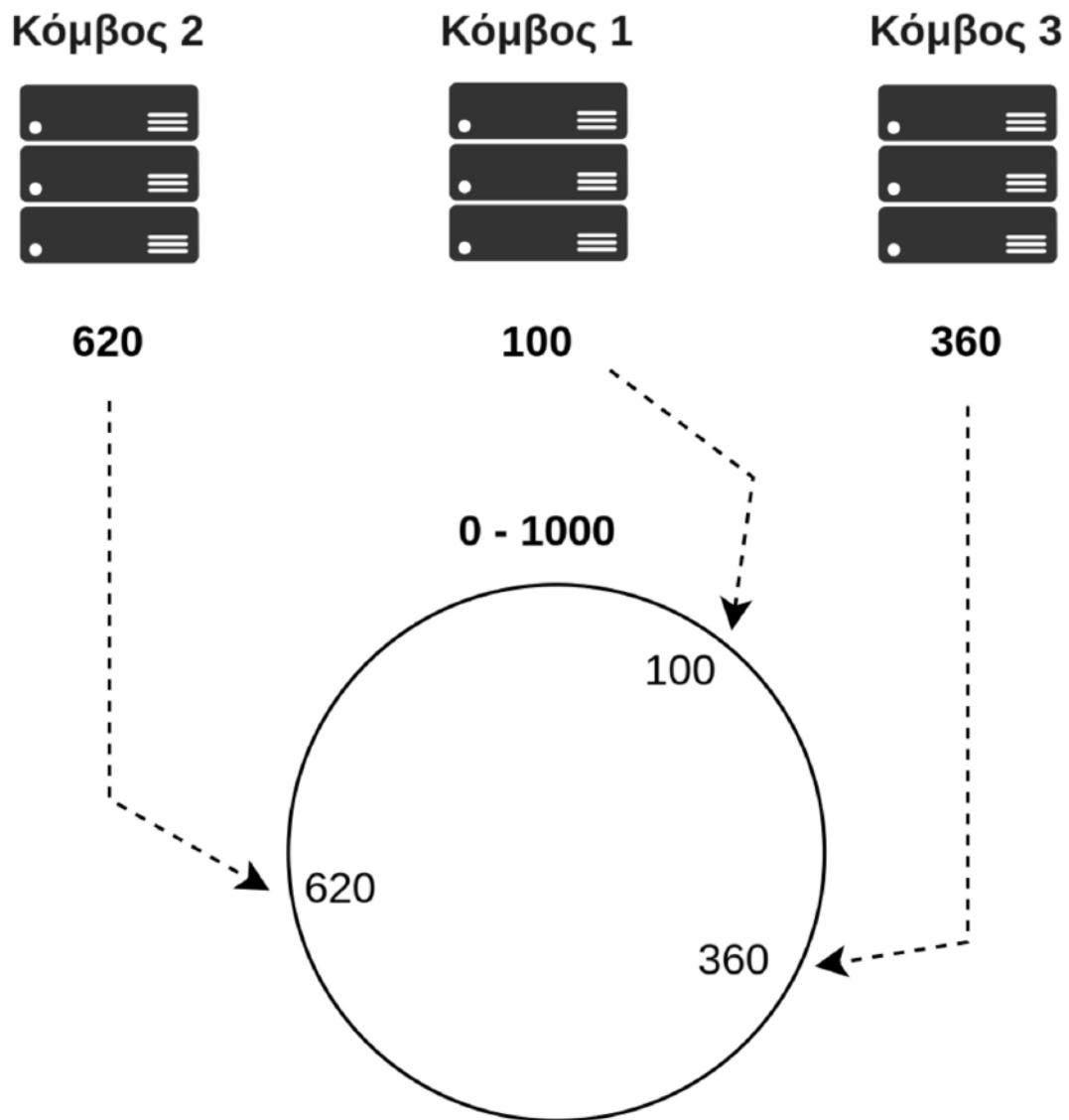
Για παράδειγμα, εάν ο κόμβος με τιμή κατακερματισμού 200 αφαιρεθεί, τα δεδομένα που ήταν αποθηκευμένα σε αυτόν μετακινούνται στον επόμενο κόμβο στον δακτύλιο, τον αμέσως επόμενο με τιμή μεγαλύτερη της τιμής 200 (π.χ. στον κόμβο με τιμή κατακερματισμού 300).

Αυτή η διαδικασία εγγυάται ότι οι αλλαγές στους κόμβους του συστήματος δεν απαιτούν πλήρη ανακατανομή των δεδομένων, γεγονός που καθιστά τον συνεπή κατακερματισμό ιδιαίτερα αποδοτικό για δυναμικά περιβάλλοντα με συχνές αλλαγές. Έτσι, το σύστημα προσαρμόζεται εύκολα στην προσθήκη ή αφαίρεση κόμβων, μειώνοντας στο ελάχιστο την ανάγκη για μετακίνηση δεδομένων και εξασφαλίζοντας ταυτόχρονα την αποδοτική κατανομή τους.

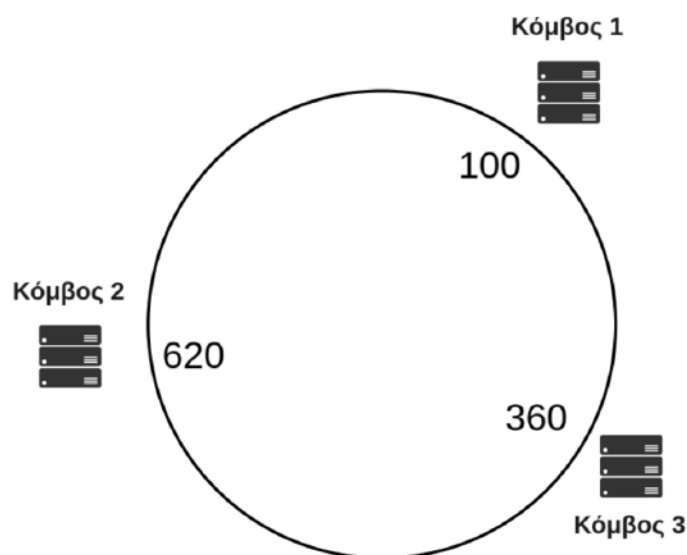
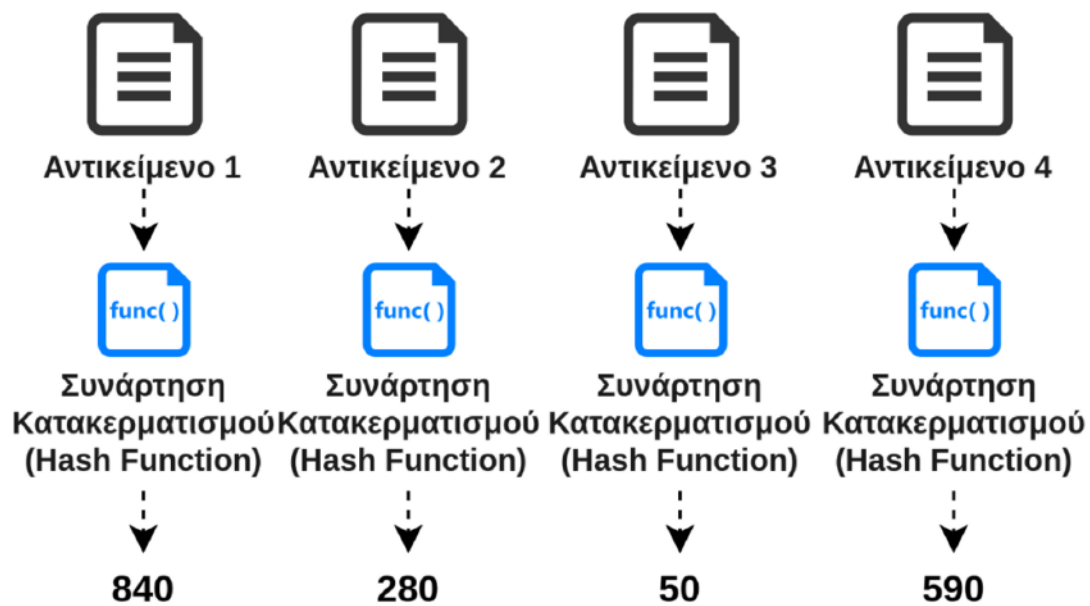
Η διαδικασία προσθήκης και αφαίρεσης κόμβων στον δακτύλιο εξασφαλίζει ότι οι αλλαγές στη σύνθεση του συστήματος δεν απαιτούν πλήρη ανακατανομή των αποθηκευμένων δεδομένων. Αντίθετα, μόνο τα αντικείμενα που ανήκουν στον νέο ή στον κόμβο που αφαιρείται επηρεάζονται, μειώνοντας σημαντικά τον αριθμό των αντικειμένων που απαιτείται να μεταφερθούν κατά τη διαδικασία ανακατανομής, διατηρώντας με τον τρόπο αυτό την αποδοτικότητα του συστήματος. Αυτή η δυναμική προσαρμογή είναι ένα από τα σημαντικότερα πλεονεκτήματα της μεθόδου του συνεπή κατακερματισμού.



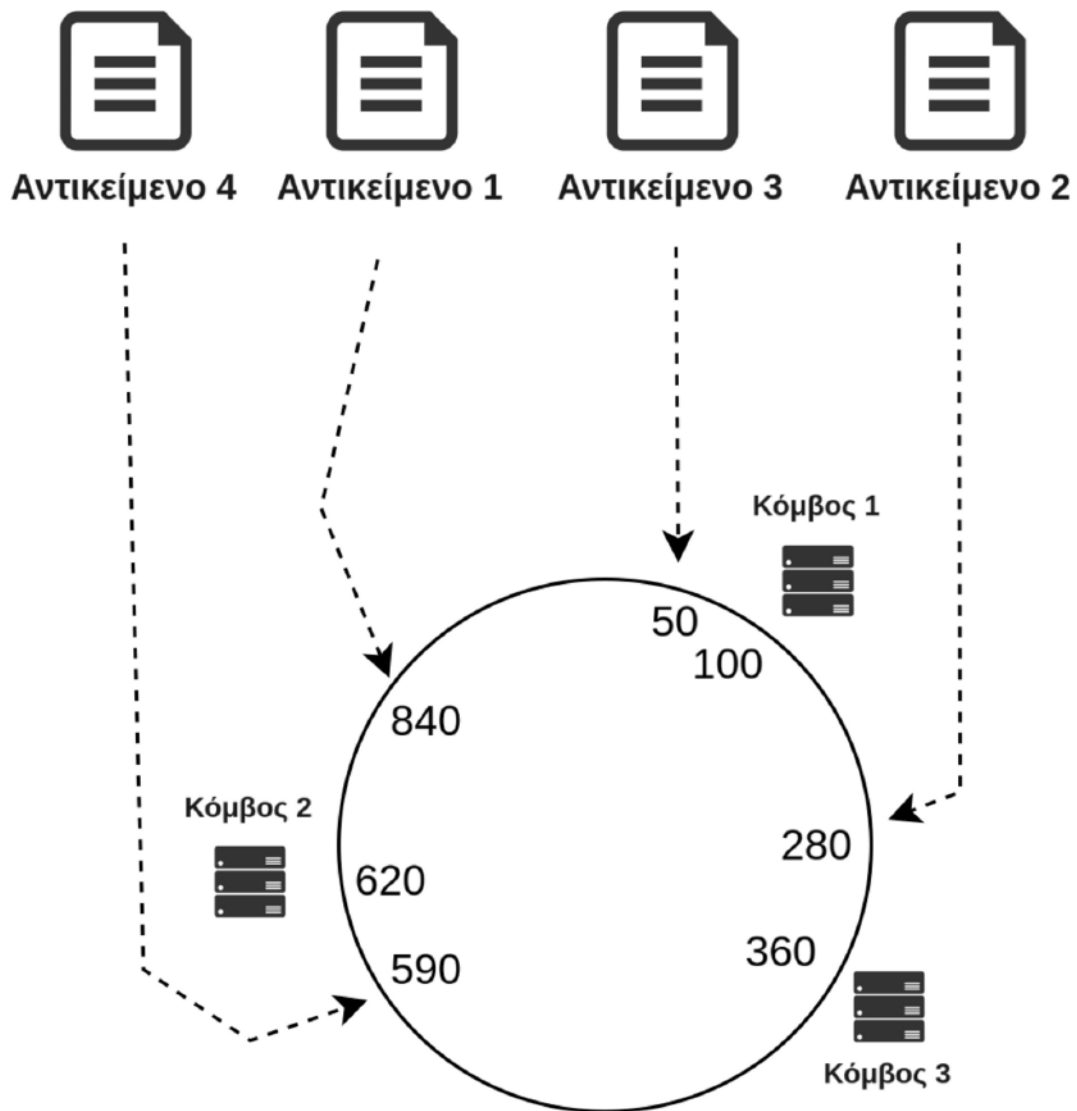
Εικόνα 5.3.1: Στην παραπάνω εικόνα αναπαρίστανται ένα παράδειγμα της διαδικασίας υπολογισμού των τιμών κατακερματισμού των κόμβων με τη χρήση της μεθόδου του συνεπή κατακερματισμού. Οι κόμβοι εισάγονται στο σύστημα αποθήκευσης και παράγονται οι τιμές κατακερματισμού τους μέσω μιας συνάρτησης κατακερματισμού. Αυτές οι τιμές στη συνέχεια θα χρησιμοποιηθούν για την τοποθέτηση των κόμβων στο δακτύλιο.



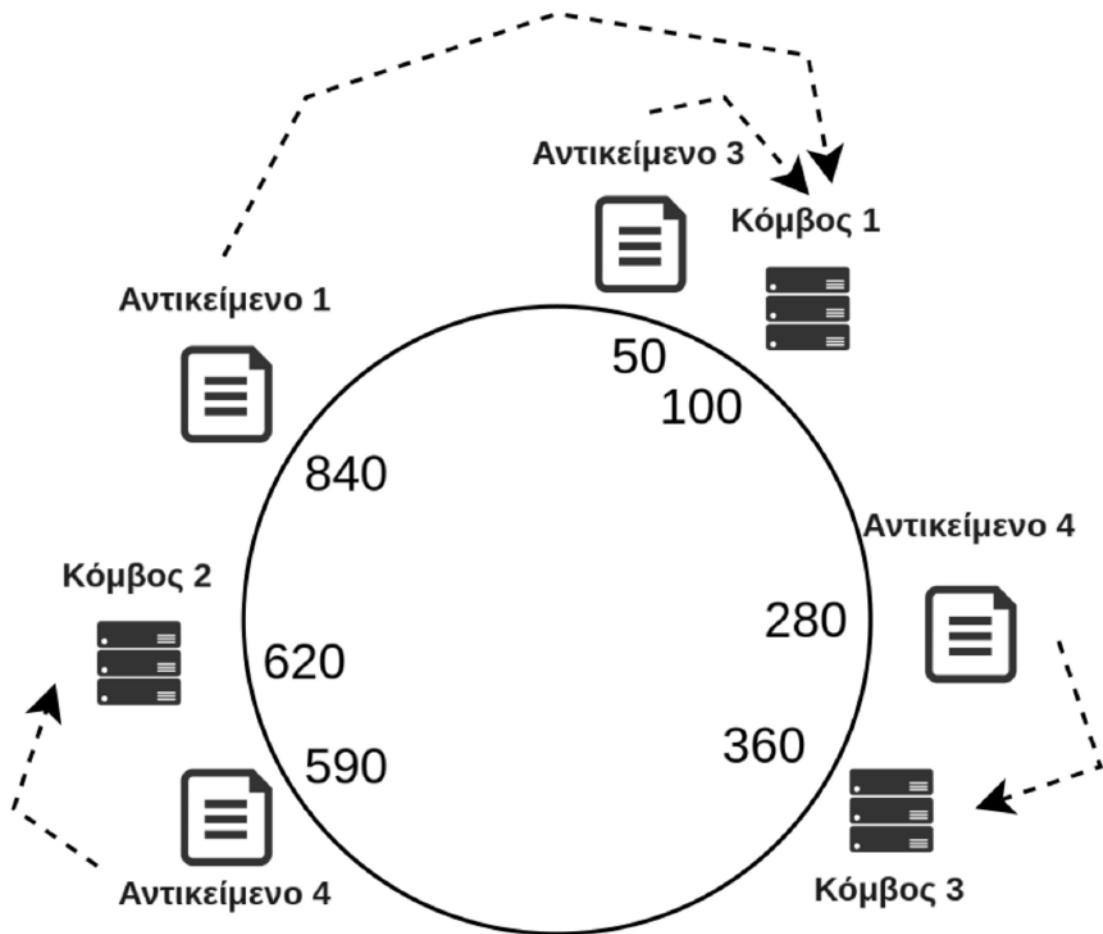
Εικόνα 5.3.2: Αφού για τους κόμβους του συστήματος υπολογιστεί η τιμή κατακερματισμού τους, τοποθετούνται στο δακτύλιο κατακερματισμού σύμφωνα με την τιμή κατακερματισμού. Στην παραπάνω εικόνα ο δακτύλιος κατακερματισμού έχει εύρος τιμών από 0 έως 1000. Κάθε κόμβος τοποθετείται στη θέση του κύκλου που αντιστοιχεί στην τιμή κατακερματισμού του.



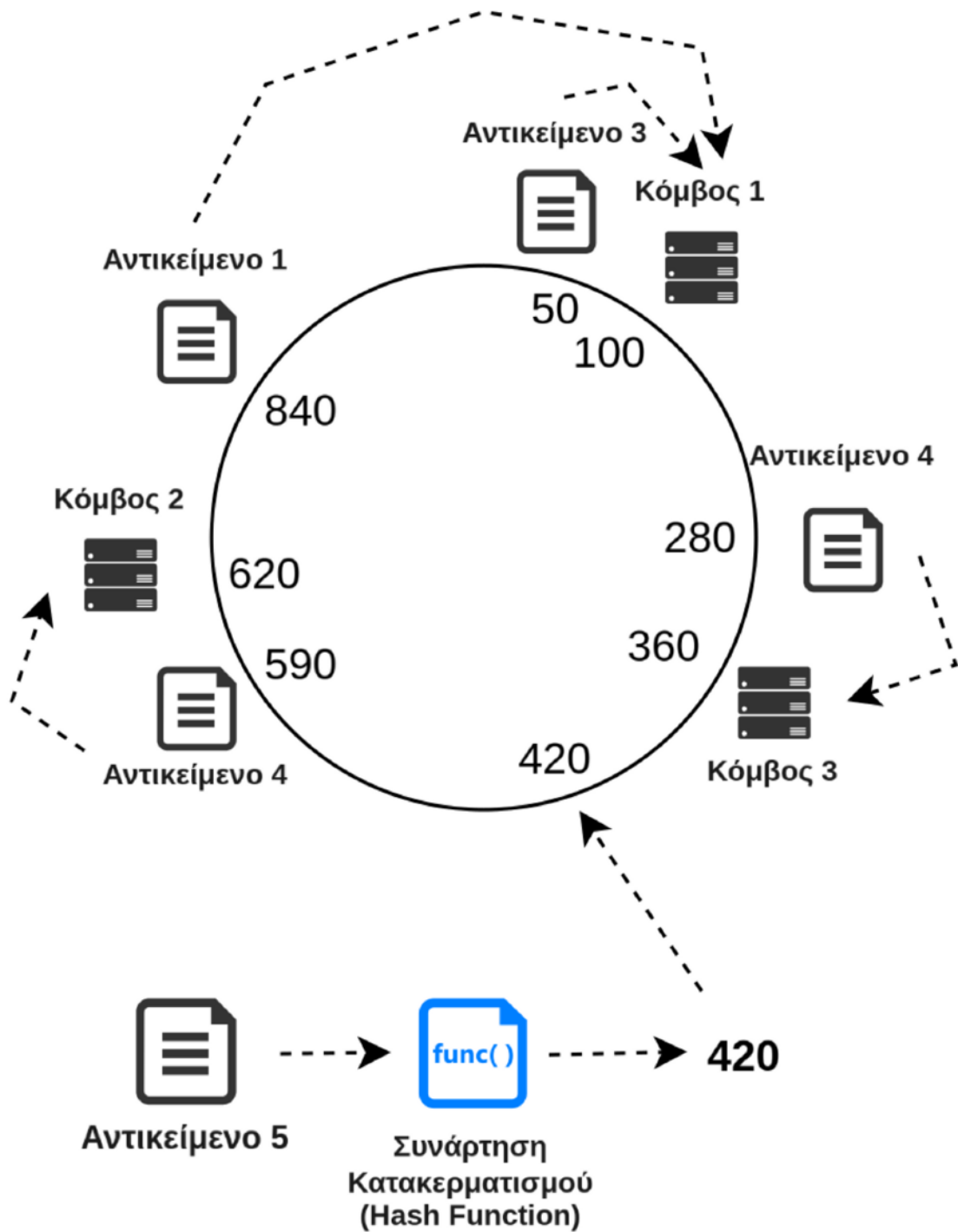
Εικόνα 5.3.3: Στην παραπάνω εικόνα γίνεται αναπαράσταση του υπολογισμού της τιμής κατακερματισμού για πέντε νέα αντικείμενα που εισάγονται για αποθήκευση στο σύστημα. Σε κάθε αντικείμενο εφαρμόζεται η συνάρτηση κατακερματισμού στο κλειδί του αντικειμένου και παράγεται το αποτέλεσμα ή η τιμή κατακερματισμού του.



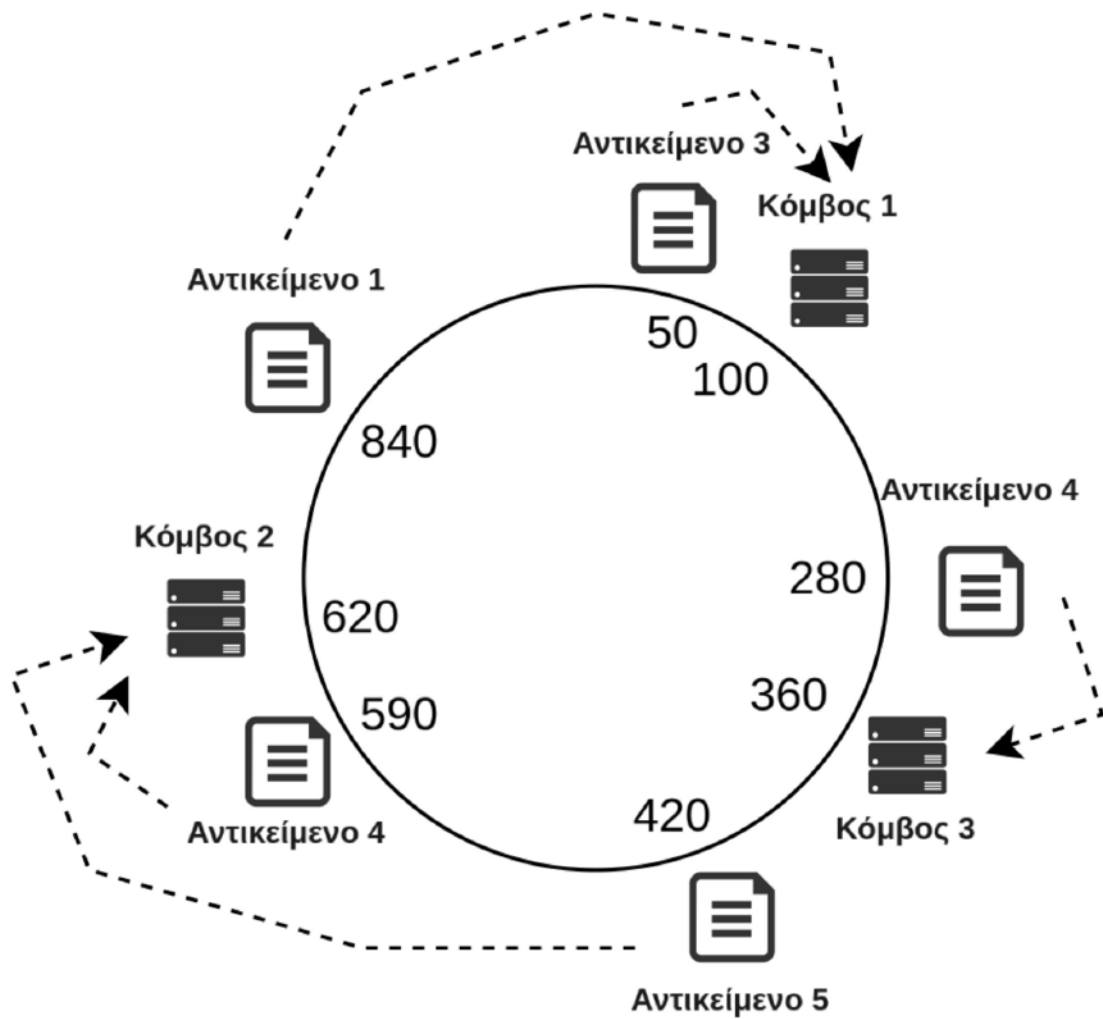
Εικόνα 5.3.4: Στην παραπάνω εικόνα απεικονίζεται ενδεικτικά η διαδικασία εισαγωγής των αντικειμένων στον δακτύλιο κατακερματισμού, σύμφωνα με τις τιμές κατακερματισμού τους. Οι κόμβοι και οι αντίστοιχες τιμές κατακερματισμού τους τοποθετούνται σε έναν κυκλικό χώρο τιμών, σχηματίζοντας τον δακτύλιο κατακερματισμού. Τα αντικείμενα εισάγονται στον κύκλο και αντιστοιχίζονται σε σημεία του δακτυλίου βάσει των τιμών κατακερματισμού τους.



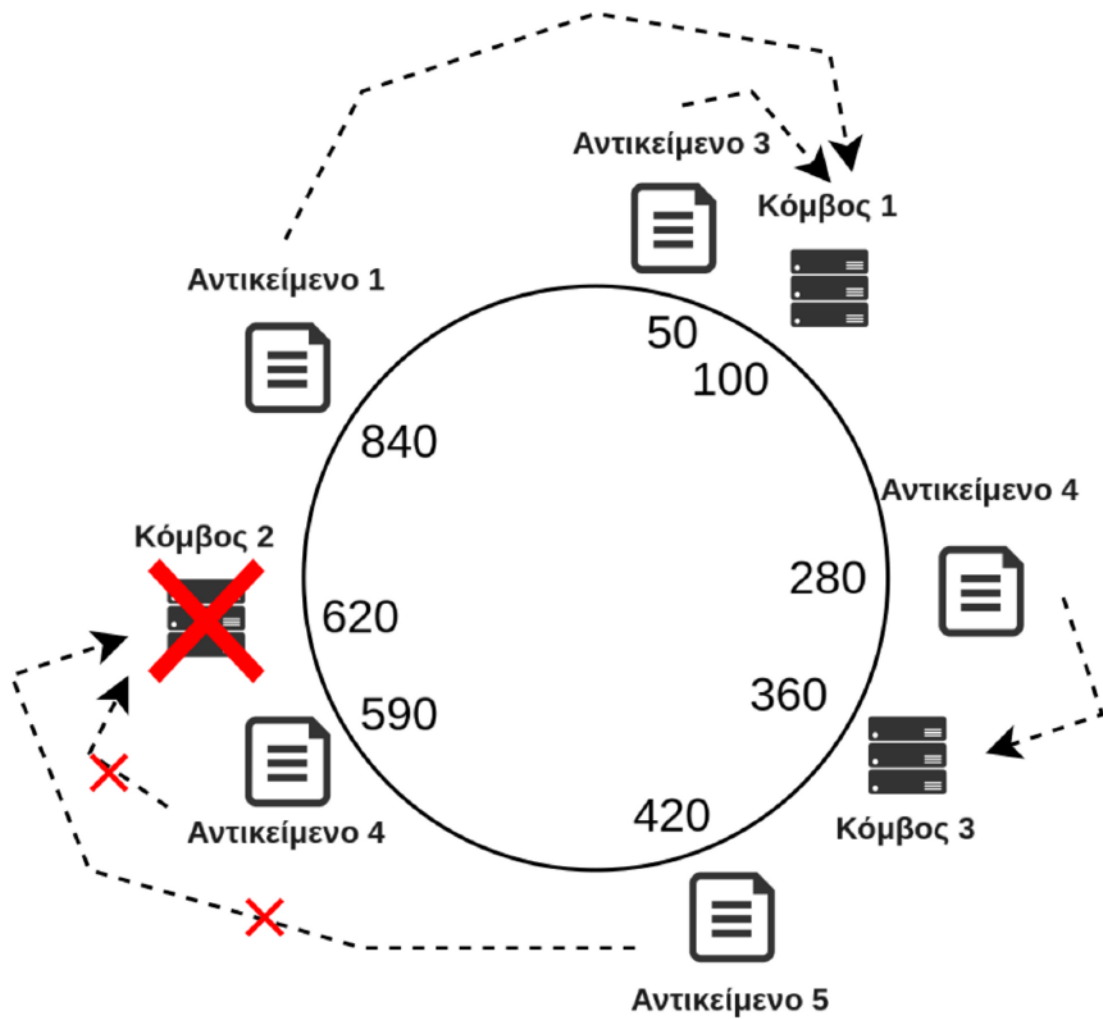
Εικόνα 5.3.5: Τα αντικείμενα που εισάγονται στο σύστημα για αποθήκευση τοποθετούνται στον δακτύλιο κατακερματισμού σύμφωνα με τις τιμές κατακερματισμού τους. Ο υπεύθυνος κόμβος για κάθε αντικείμενο προσδιορίζεται εντοπίζοντας τον πρώτο κόμβο στον δακτύλιο που έχει τιμή κατακερματισμού μεγαλύτερη από αυτή του αντικειμένου. Η αναζήτηση του υπεύθυνου κόμβου πραγματοποιείται με κατεύθυνση σύμφωνα με τη φορά του ρολογιού πάνω στον δακτύλιο κατακερματισμού.



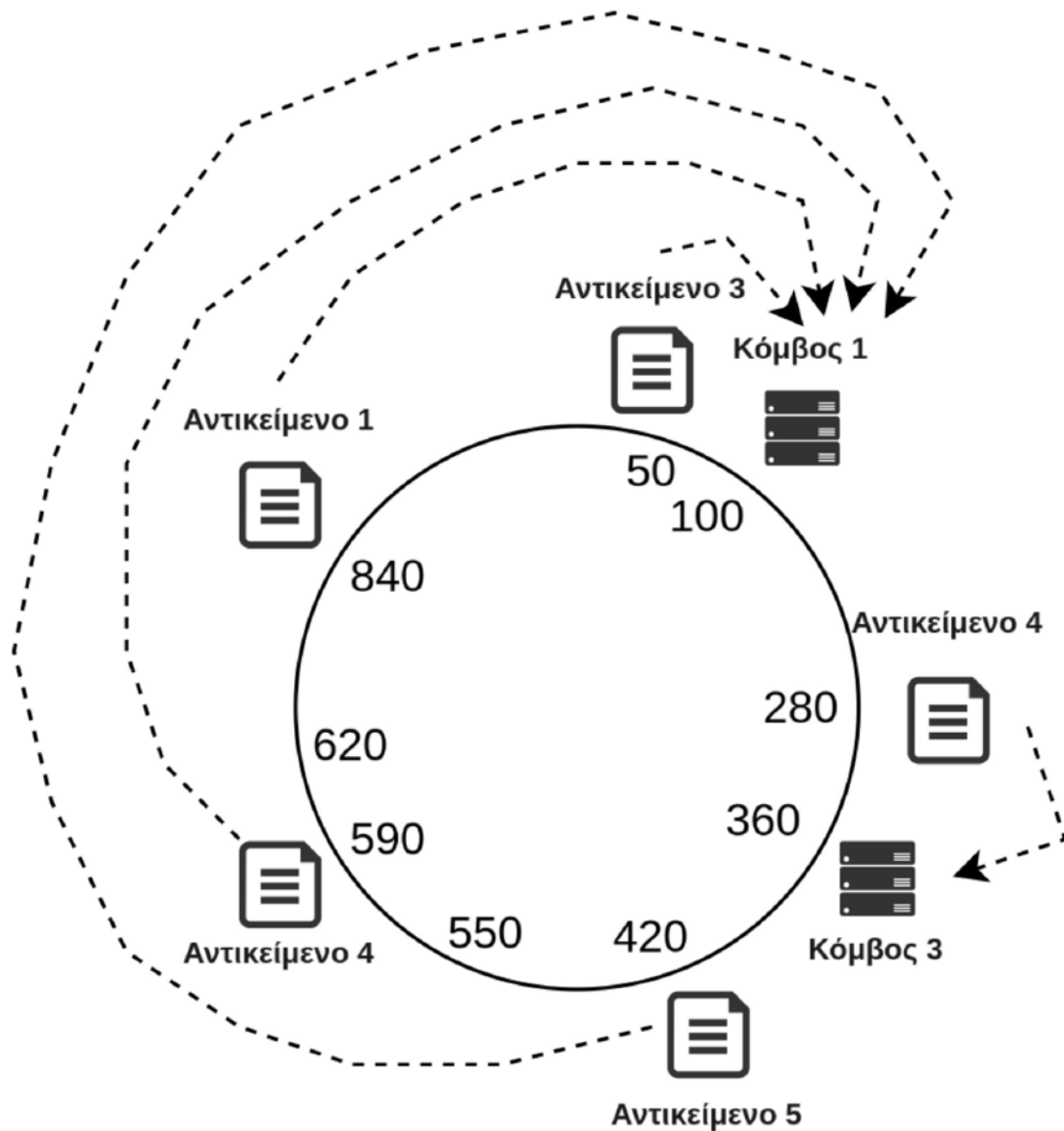
Εικόνα 5.3.6: Στο παράδειγμα πραγματοποιείται η εισαγωγή ενός νέου αντικειμένου στο σύστημα. Αρχικά, η τιμή κατακερματισμού του υπολογίζεται μέσω της συνάρτησης κατακερματισμού. Στη συνέχεια, το αντικείμενο τοποθετείται στη σωστή θέση στον νοητό δακτύλιο κατακερματισμού, ακολουθώντας τη διαδικασία εκχώρησης.



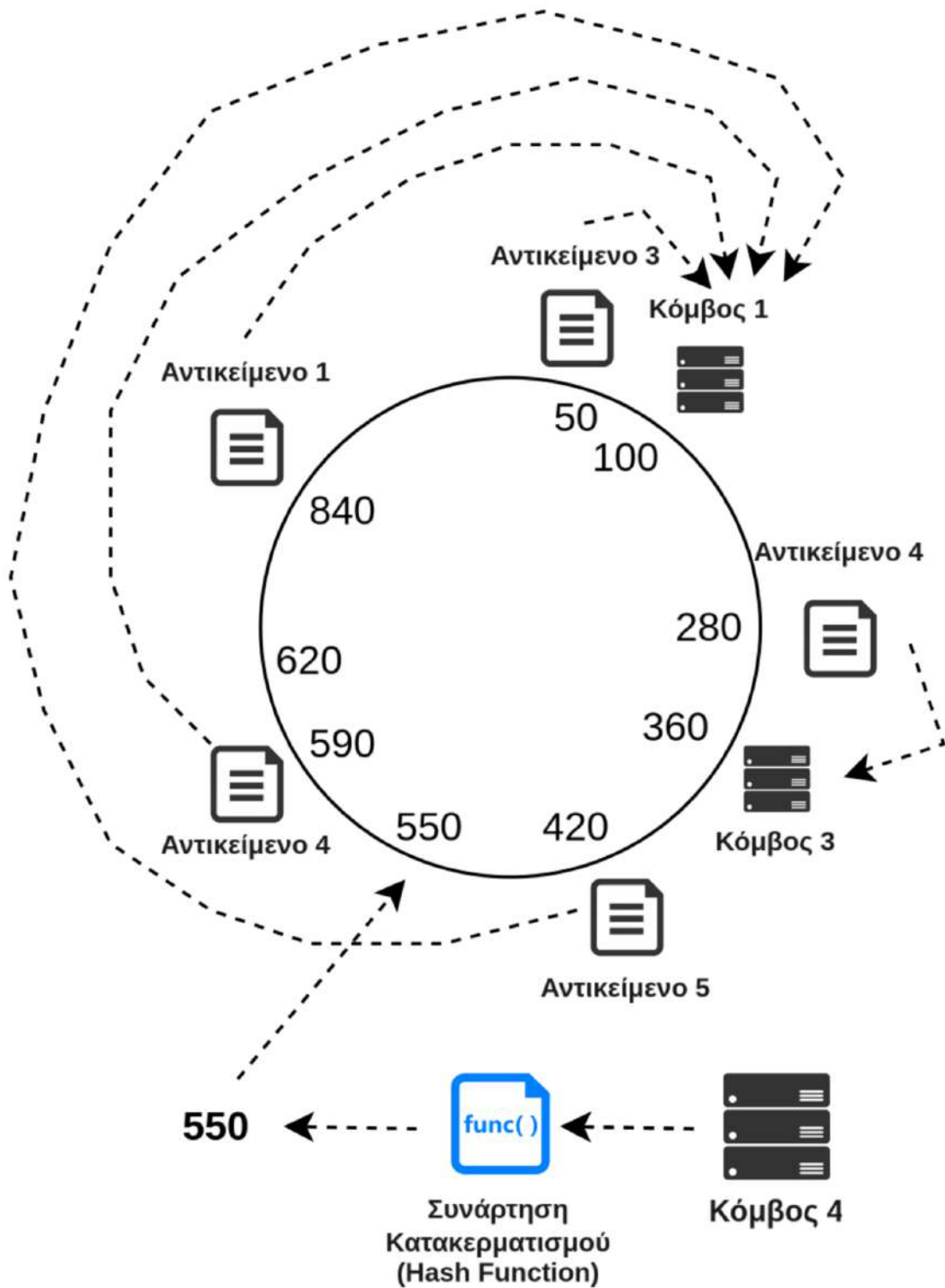
Εικόνα 5.3.7: Αφού το νέο αντικείμενο τοποθετηθεί στο δακτύλιο αντιστοιχίζεται στον πρώτο διαθέσιμο κόμβο του συστήματος με μεγαλύτερη τιμή κατακερματισμού από αυτή του αντικειμένου.



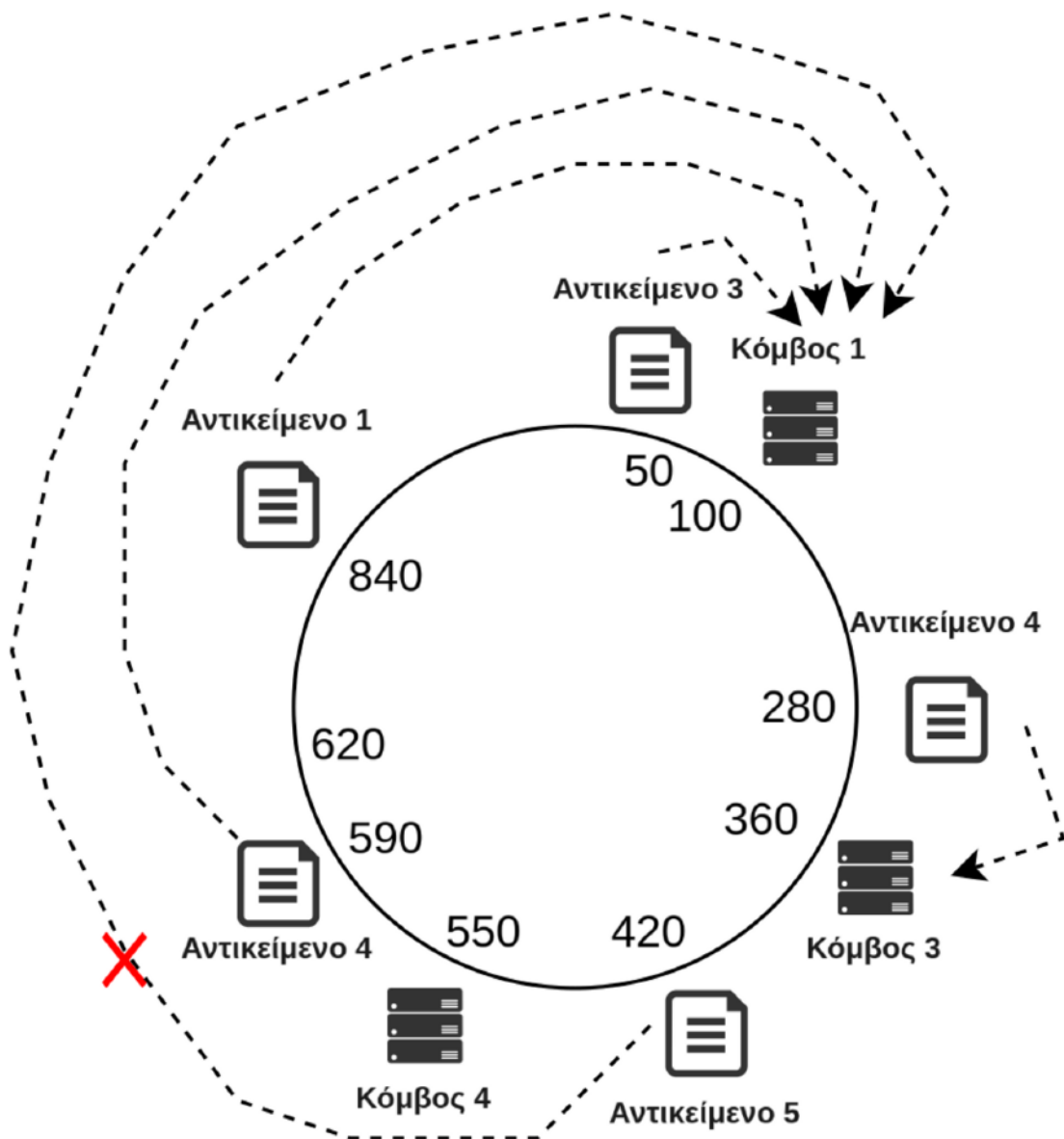
Εικόνα 5.3.8: Στην παραπάνω εικόνα παρουσιάζεται ένα παράδειγμα όπου ένας από τους κόμβους του συστήματος αποθήκευσης αφαιρείται, είτε λόγω αποχώρησης είτε λόγω βλάβης. Σε αυτή την περίπτωση, τα αντικείμενα που ήταν αποθηκευμένα στον συγκεκριμένο κόμβο πρέπει να μετακινηθούν σε άλλο διαθέσιμο κόμβο του συστήματος. Στο παράδειγμα, τα αντικείμενα 4 και 5 ήταν αποθηκευμένα στον κόμβο 2, ο οποίος αφαιρείται. Μετά την απομάκρυνση του κόμβου 2, τα αντικείμενα αναζητούν τον επόμενο διαθέσιμο κόμβο στον δακτύλιο κατακερματισμού, στον οποίο και μεταφέρονται, διασφαλίζοντας τη συνέχεια της αποθήκευσης.



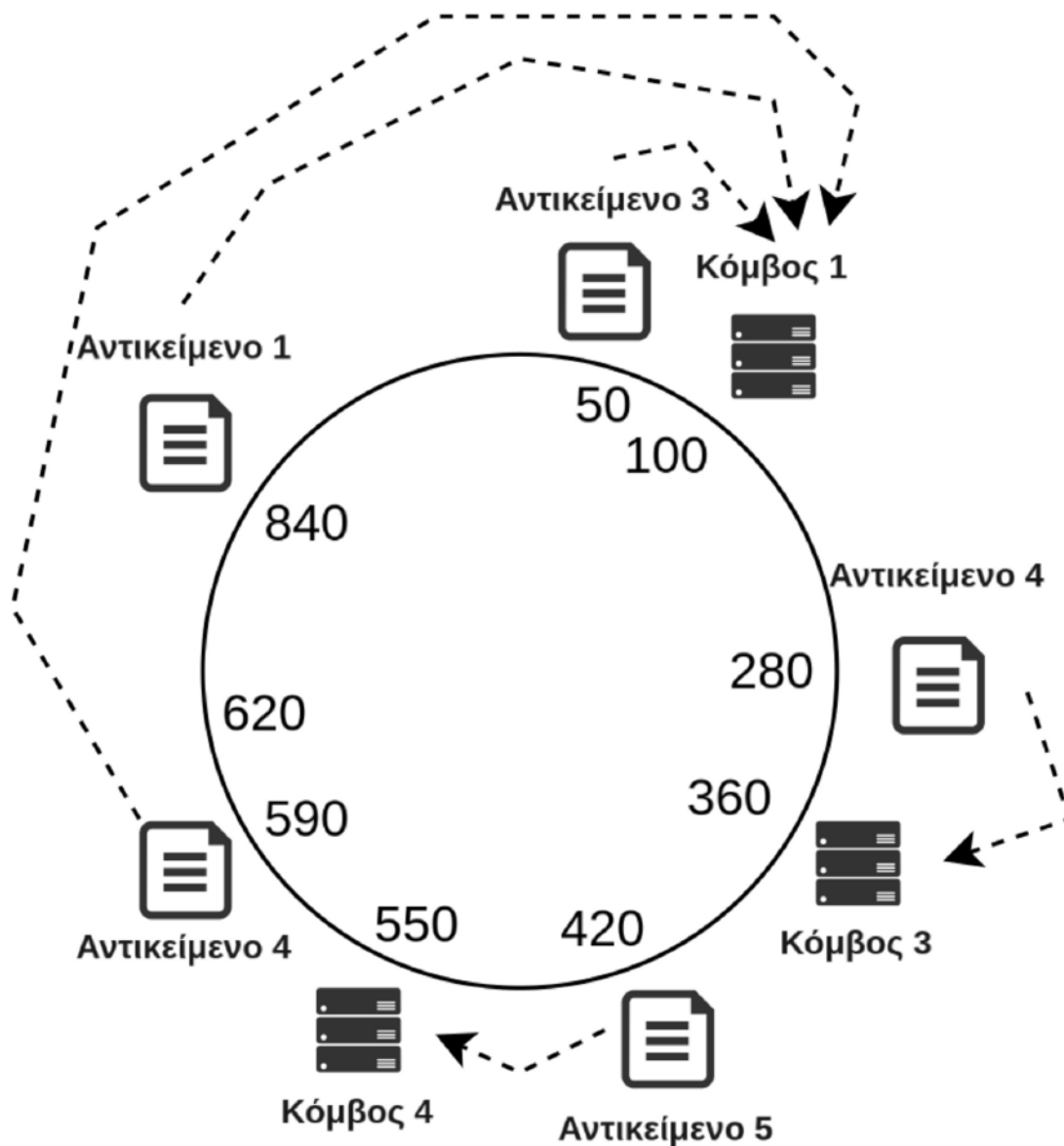
Εικόνα 5.3.9: Μετά την αφαίρεση του κόμβου 2 από το σύστημα, τα αντικείμενα μετακινούνται στον αμέσως επόμενο διαθέσιμο κόμβο στον δακτύλιο κατακερματισμού. Σύμφωνα με τη φορά του ρολογιού, ο πρώτος διαθέσιμος κόμβος που εντοπίζεται στο παράδειγμα είναι ο κόμβος 1, όπου και μεταφέρονται τα αντικείμενα.



Εικόνα 5.3.10: Οπτική αναπαράσταση της εισαγωγής ενός νέου κόμβου σε ένα σύστημα αποθήκευσης που λειτουργεί με τη μέθοδο του συνεπή κατακερματισμού. Στην παραπάνω εικόνα, ένας νέος κόμβος, ο κόμβος 4, προστίθεται στο σύστημα. Αρχικά, υπολογίζεται η τιμή κατακερματισμού του κόμβου και στη συνέχεια τοποθετείται στη θέση που αντιστοιχεί στον δακτύλιο κατακερματισμού.



Εικόνα 5.3.11: Αφού ο νέος κόμβος προστεθεί στο σύστημα και τοποθετηθεί στη σωστή θέση στον δακτύλιο κατακερματισμού, πραγματοποιείται έλεγχος για τυχόν αντικείμενα που εμπίπτουν στην περιοχή ευθύνης του αλλά είναι αποθηκευμένα σε άλλον κόμβο. Όπως φαίνεται στο σχήμα, στην περιοχή του δακτυλίου που αντιστοιχεί στον νέο κόμβο βρίσκεται το αντικείμενο 5, το οποίο είναι ήδη αποθηκευμένο στον κόμβο 1. Ωστόσο, μετά την προσθήκη του κόμβου 4, η αντιστοίχιση του αντικειμένου 5 στον κόμβο 1 δεν είναι πλέον έγκυρη, καθώς πρέπει να μετακινηθεί στον νέο κόμβο που προστέθηκε.



Εικόνα 5.3.11: Μετά τη διαδικασία ελέγχου για ανακατανομή των αντικειμένων λόγω της προσθήκης του κόμβου 4 στο σύστημα αποθήκευσης, το αντικείμενο 5 μεταφέρεται στον κόμβο 4, ο οποίος είναι ο επόμενος κόμβος στον δακτύλιο κατακερματισμού και πλέον υπεύθυνος για το αντικείμενο.

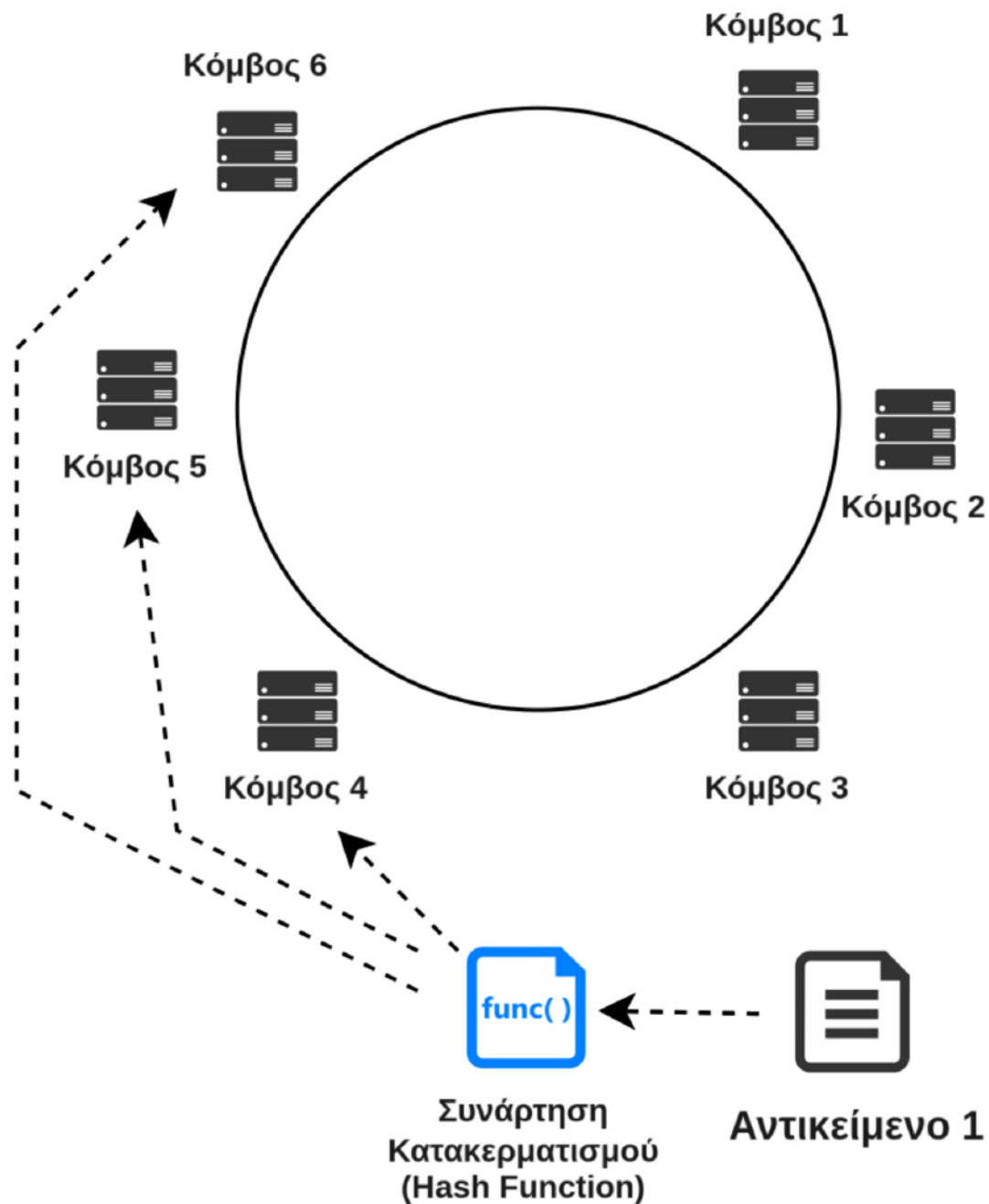
5.4 Επέκταση του συνεπή κατακερματισμού για τη διατήρηση αντιγράφων

Όπως περιγράφεται σε προηγούμενη ενότητα, οι προδιαγραφές και τα ζητούμενα που σχετίζονται με την υλοποίηση του συστήματος αποθήκευσης, απαιτείται η κατανεμημένη εκτέλεσης του αλγορίθμου κατανομής, η ικανότητα προσαρμογής του συστήματος σε δυναμικά περιβάλλοντα καθώς η δυνατότητα ομαλής κλιμάκωση του συστήματος. Αυτά τα ζητούμενα έχουν επιτευχθεί με τη χρήση της μεθόδου του συνεπή κατακερματισμού ή αλλιώς Consistent Hashing, η οποία εξασφαλίζει ομοιόμορφη κατανομή των δεδομένων, δυναμική προσαρμογή σε αλλαγές των κόμβων και αποδοτική διαχείριση των δεδομένων χωρίς την ανάγκη για πλήρη ανακατανομή των αντικειμένων σε περιπτώσεις προσθήκης ή αφαίρεσης κάποιου κόμβου.

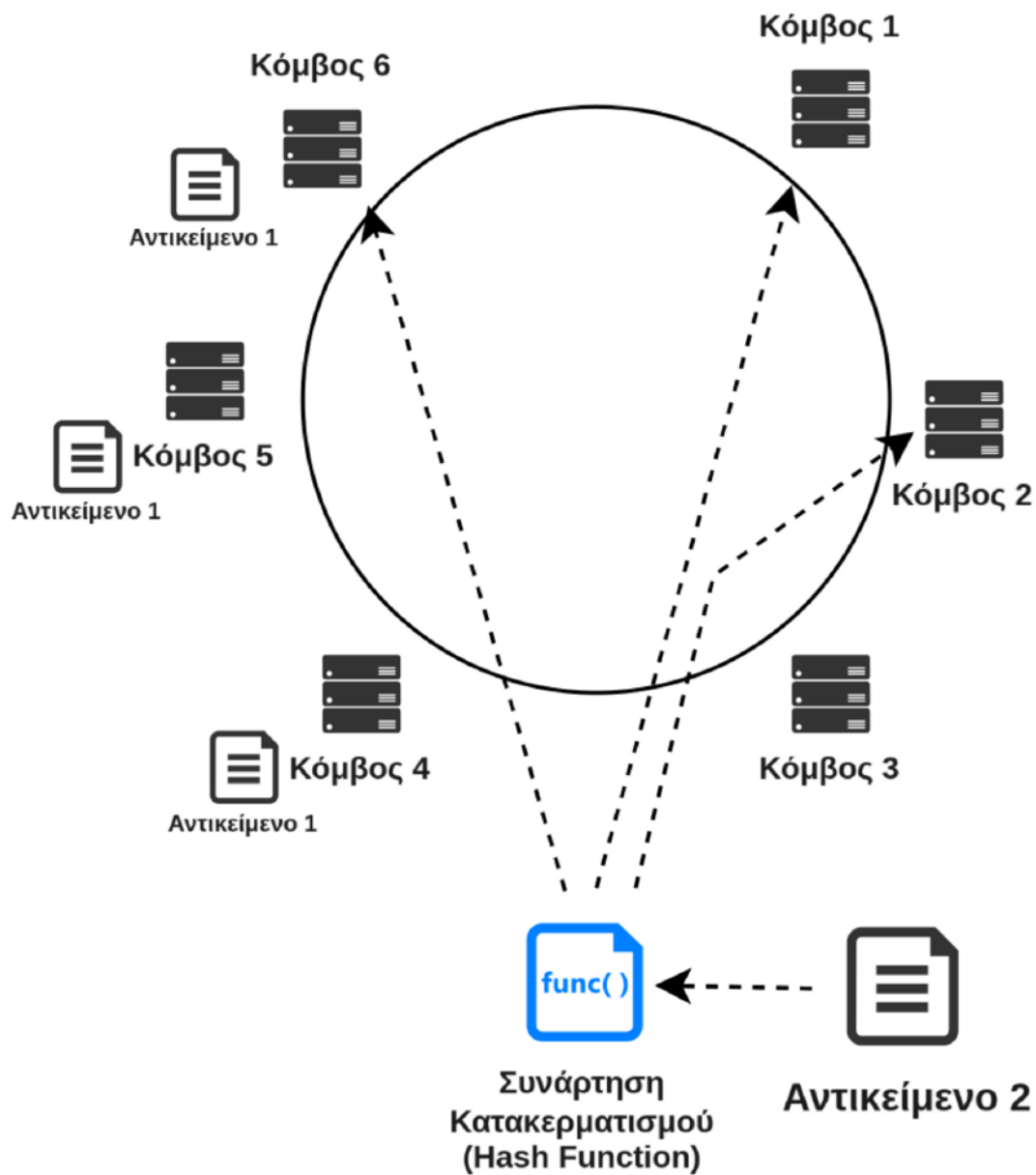
Ωστόσο, ακόμα ένα σημαντικό ζητούμενο που πρέπει να διασφαλιστεί στην ανάπτυξη του συστήματος είναι η διατήρηση της διαθεσιμότητας των δεδομένων, καθώς επίσης η ανθεκτικότητα του συστήματος σε βλάβες και αποτυχίες κόμβων. Σε ένα σύστημα με πολλαπλούς κόμβους, είναι απαραίτητο να εξασφαλιστεί ότι τα δεδομένα παραμένουν προσβάσιμα ακόμα και σε περίπτωση αποτυχίας ενός ή περισσότερων κόμβων, ενώ ταυτόχρονα να αποφεύγεται ο κίνδυνος της οριστικής απώλεια αποθηκευμένων δεδομένων.

Αυτό μπορεί να επιτευχθεί με τη χρήση της μεθόδου αντιγραφής δεδομένων (replication), όπου κάθε αντικείμενο αποθηκεύεται σε πολλαπλούς κόμβους του συστήματος. Στόχος είναι η τροποποίηση της μεθόδου του συνεπή κατακερματισμού (Consistent Hashing) ώστε να υποστηρίξει τη διατήρησης αντιγράφων. Η συγκεκριμένη τροποποίηση βασίζεται στην ιδέα ότι κάθε αντικείμενο δεν αποθηκεύεται μόνο στον υπεύθυνο κόμβο που αντιστοιχίζεται με τη συνάρτηση κατακερματισμού, αλλά και σε άλλους κόμβους του συστήματος στον δακτύλιο. Με αυτόν τον τρόπο, στο σύστημα εξασφαλίζεται ότι τα αντικείμενα παραμένουν προσβάσιμα ακόμα και σε περίπτωση αποτυχίας ενός ή περισσότερων κόμβων, μειώνοντας τον κίνδυνο απώλειας δεδομένων και βελτιώνοντας τη συνολική ανθεκτικότητα του συστήματος αποθήκευσης.

Όταν ένα αντικείμενο εισάγεται προς αποθήκευση στο σύστημα, αρχικά υπολογίζεται η τιμή κατακερματισμού του, προκειμένου να προσδιοριστεί ο υπεύθυνος κόμβος στον οποίο θα αποθηκευτεί, όπως συμβαίνει και στη μέθοδο του συνεπή κατακερματισμού. Ωστόσο, σε αντίθεση με την κλασική προσέγγιση, το αντικείμενο δεν αποθηκεύεται αποκλειστικά στον υπεύθυνο κόμβο που έχει αντιστοιχιστεί, αλλά αναπαράγεται ή αλλιώς διατηρεί αντίγραφα και σε επιπλέον κόμβους εντός του δακτυλίου κατακερματισμού. Για παράδειγμα, εάν το σύστημα έχει ρυθμιστεί να διατηρεί τρία αντίγραφα για κάθε αντικείμενο, τότε το αρχικό αντίγραφο αποθηκεύεται στον κύριο κόμβο, ενώ τα υπόλοιπα δύο αντίγραφα τοποθετούνται στους επόμενους κόμβους με τις αμέσως υψηλότερες τιμές κατακερματισμού.



Εικόνα 5.4.1: Στην παραπάνω εικόνα απεικονίζεται η διαδικασία προσθήκης ενός νέου αντικειμένου σε ένα σύστημα αποθήκευσης που χρησιμοποιεί συνεπή κατακερματισμό, ενώ ταυτόχρονα διατηρεί πολλαπλά αντίγραφα των αντικειμένων μέσω της μεθόδου της αντιγραφής. Στο παράδειγμα, το αντικείμενο 1 εισάγεται και αποθηκεύεται στον κόμβο 4 του συστήματος, ο οποίος είναι υπεύθυνος για την αποθήκευση του συγκεκριμένου αντικειμένου. Ταυτόχρονα, δημιουργούνται και αποθηκεύονται τρία αντίγραφα του αντικειμένου, τα οποία τοποθετούνται στον υπεύθυνο κόμβο καθώς και στους δύο επόμενους κόμβους του δακτυλίου κατακερματισμού, δηλαδή στον κόμβο 5 και στον κόμβο 6 του συστήματος.



Εικόνα 5.4.2: Στη συνέχεια, στο σύστημα αποθήκευσης που χρησιμοποιεί τη μέθοδο του συνεπή κατακερματισμού με διατήρηση πολλαπλών αντιγράφων, εισάγεται ακόμα ένα αντικείμενο, το αντικείμενο 2. Ο υπεύθυνος κόμβος για την αποθήκευση του αντικειμένου είναι ο κόμβος 6. Ωστόσο, λόγω της μεθόδου της διατήρησης πολλαπλών αντιγράφων, το αντικείμενο αποθηκεύεται και στους επόμενους δύο κόμβους του δακτυλίου κατακερματισμού, τον κόμβο 1 και τον κόμβο 2, όπως απεικονίζεται στην εικόνα.

ΚΕΦΑΛΑΙΟ 6 Υλοποίηση του συστήματος αποθήκευσης

Σε αυτό το κεφάλαιο, παρουσιάζεται η υλοποίηση του κατανεμημένου συστήματος αποθήκευσης δεδομένων που αναπτύχθηκε για την παρούσα διπλωματική εργασία. Το σύστημα έχει σχεδιαστεί με στόχο την αποθήκευση και τη διαχείριση αντικειμένων σε ένα κατανεμημένο περιβάλλον, όπου πολλαπλοί αυτόνομοι κόμβοι ενός δικτύου συνεργάζονται μεταξύ τους και σχηματίζουν το σύστημα αποθήκευσης.

Η υλοποίηση περιλαμβάνει τη διαχείριση των κόμβων του συστήματος, την επικοινωνία μεταξύ τους, καθώς και τη διαδικασία αποθήκευσης, ανάκτησης και διαγραφής των αντικειμένων. Κάθε κόμβος του συστήματος είναι υπεύθυνος για τη διατήρηση των τοπικών του δεδομένων, ενώ παράλληλα ανταλλάσσει πληροφορίες με τους υπόλοιπους κόμβους για να διασφαλιστεί η συνέπεια της κοινής κατάστασης στους ξεχωριστούς κόμβους του συστήματος.

Αρχικά, παρουσιάζεται η επιλογή της γλώσσας προγραμματισμού που χρησιμοποιήθηκε για την ανάπτυξη του συστήματος, καθώς οι λόγοι και παράγοντες που οδήγησαν σε αυτήν την επιλογή. Στη συνέχεια, αναλύονται τα βασικά χαρακτηριστικά της υλοποίησης, περιγράφοντας τις δομές δεδομένων που χρησιμοποιούν οι κόμβοι και τις βασικές λειτουργίες που εκτελούν. Έμφαση δίνεται στον τρόπο με τον οποίο οι κόμβοι του συστήματος διαχειρίζονται την προσθήκη και αφαίρεση κόμβων και αντικειμένων, την ενημέρωση της λίστας των ενεργών κόμβων, καθώς και τη διαδικασία της ανακατανομής των αποθηκευμένων αντικειμένων.

Επιπλέον, το κεφάλαιο καλύπτει τον τρόπο με τον οποίο οι κόμβοι χειρίζονται την προσθήκη και αφαίρεση κόμβων και αντικειμένων, τη δυναμική ενημέρωση της λίστας των κόμβων και αποθηκευμένων δεδομένων, καθώς και τη διαδικασία ανακατανομής των αντικειμένων όταν προκύπτουν αλλαγές στη σύνθεση της ομάδας των κόμβων.

Επιπλέον, στο κεφάλαιο εξετάζεται και αναλύεται ο τρόπος λειτουργίας των κόμβων, εστιάζοντας στη διαχείριση της προσθήκης και αφαίρεσης κόμβων, καθώς και στη διαδικασία αποθήκευσης, ανάκτησης και διαγραφής αντικειμένων. Παρουσιάζεται, επίσης, ο μηχανισμός δυναμικής ενημέρωσης της κατάστασης των κόμβων και των αποθηκευμένων αντικειμένων, καθώς και η διαδικασία ανακατανομής των αντικειμένων όταν μεταβάλλεται η σύνθεση της ομάδας των κόμβων του συστήματος.

6.1 Επιλογή της γλώσσας προγραμματισμού

Η υλοποίηση του συστήματος έγινε με τη γλώσσα προγραμματισμού Python, η οποία επιλέχθηκε για τους εξής λόγους:

1. Γλώσσα υψηλού επιπέδου: Η Python είναι μια γλώσσα υψηλού επιπέδου, η οποία επιτρέπει την ανάπτυξη εφαρμογών με απλό και ευανάγνωστο κώδικα. Αυτό την καθιστά ιδανική για την υλοποίηση εφαρμογών που απαιτούν γρήγορη ανάπτυξη και ευκολία στην υλοποίηση πολύπλοκων λειτουργιών. Με τη χρήση της Python, δεν απαιτείται η διαχείριση χαμηλού επιπέδου λειτουργιών, όπως η διαχείριση μνήμης, επιτρέποντας στους προγραμματιστές να εστιάσουν στη λογική και τη λειτουργικότητα των εφαρμογών. Αυτό το χαρακτηριστικό καθιστά την Python κατάλληλη τόσο για την ανάπτυξη ολοκληρωμένων συστημάτων όσο και για τη δημιουργία πρωτοτύπων.
2. Ευρεία χρήση και πληθώρα βιβλιοθηκών: Η Python χρησιμοποιείται ευρέως στην ανάπτυξη εφαρμογών και συστημάτων. Μερικά βασικά παραδείγματα χρήσης της περιλαμβάνουν την ανάλυση δεδομένων, τη μηχανική μάθηση, την ανάπτυξη εφαρμογών διαδικτύου και τη δημιουργία εργαλείων για τη διαχείριση και επεξεργασία δεδομένων. Επιπλέον, η Python παρέχει μια πληθώρα διαθέσιμων βιβλιοθηκών και εργαλείων, τα οποία διευκολύνουν την υλοποίηση εξειδικευμένων λειτουργιών, όπως η επεξεργασία δεδομένων, η διαχείριση αιτήσεων δικτύου και η διαχείριση αρχείων. Αυτές οι δυνατότητες καθιστούν την Python μια ιδανική επιλογή για την υλοποίηση της συγκεκριμένης εργασίας.
3. Υποστήριξη από πολλαπλές πλατφόρμες: Ένα από τα βασικότερα πλεονεκτήματα της Python είναι η υποστήριξή της από πολλαπλές πλατφόρμες. Η Python έχει τη δυνατότητα να εκτελείται σε όλα τα σύγχρονα λειτουργικά συστήματα (Windows, Linux, macOS) χωρίς επιπλέον απαιτήσεις και χωρίς την ανάγκη για σημαντικές τροποποιήσεις στον κώδικα. Αυτή η δυνατότητα διασφαλίζει τη συμβατότητα των εφαρμογών σε διαφορετικά περιβάλλοντα, επιτρέποντας την εύκολη ανάπτυξη και μεταφορά εφαρμογών μεταξύ διαφορετικών συστημάτων.
4. Ταχεία ανάπτυξη και ευελιξία: Η Python επιτρέπει την ταχεία ανάπτυξη εφαρμογών, καθιστώντας την ιδιαίτερα χρήσιμη για τη δημιουργία πρωτοτύπων και τη δοκιμή διαφορετικών προσεγγίσεων στην υλοποίηση. Χάρη στην ευκολία σύνταξής της, επιτρέπει την ανάπτυξη λειτουργικών εφαρμογών με απλούστερο και πιο κατανοητό τρόπο συγκριτικά με άλλες γλώσσες προγραμματισμού. Αυτό μειώνει σημαντικά τον χρόνο ανάπτυξης καθώς και παρέχει μεγαλύτερη ευκολία σε αλλαγές και τροποποιήσεις στη λειτουργικότητα των εφαρμογών.

6.2 Βιβλιοθήκες και εργαλεία που χρησιμοποιούνται

Παρακάτω παρουσιάζονται οι βιβλιοθήκες της Python που χρησιμοποιήθηκαν για την ανάπτυξη του συστήματος, καθώς και ο ρόλος τους στην υλοποίηση:

Βιβλιοθήκη Bisect

Η βιβλιοθήκη `bisect` χρησιμοποιήθηκε για τη διαχείριση των ταξινομημένων των λιστών των κόμβων και των αντικειμένων. Η `bisect` παρέχει λειτουργίες για την εισαγωγή και αναζήτηση στοιχείων σε ταξινομημένες λίστες με αποδοτικό τρόπο. Στο σύστημα, χρησιμοποιήθηκε στις λειτουργίες εισαγωγής εγγραφών σε λίστες ώστε να διατηρείται η ταξινόμηση των κόμβων και των αντικειμένων βάσει των τιμών κατακερματισμού τους, γεγονός που επιταχύνει την αποδοτικότερη και ταχύτερη αναζήτηση και την εισαγωγή νέων αντικειμένων.

Βιβλιοθήκη JSON

Η βιβλιοθήκη `json` της Python χρησιμοποιείται για την αναπαράσταση δεδομένων σε μορφή JSON (JavaScript Object Notation), η οποία είναι μια συχνά χρησιμοποιούμενη μέθοδος για την αναπαράσταση δομημένων δεδομένων και την μεταφορά ή την αποθήκευσή τους. Στο πλαίσιο της υλοποίησης του συστήματος, η βιβλιοθήκη `json` επιτρέπει την ανταλλαγή πληροφοριών και δεδομένων μεταξύ των κόμβων, καθώς και στις αιτήσεις των χρηστών στο σύστημα και στις απαντήσεις στα αιτήματα τους αιτήματα. Όλα τα δεδομένα που ανταλλάσσονται μέσω δικτύου κωδικοποιούνται και αποκωδικοποιούνται σε μορφή JSON. Η βιβλιοθήκη αυτή επιτρέπει τη μετατροπή μεταβλητών και δομών δεδομένων όπως λεξικά, λίστες καθώς και άλλες δομές της Python σε μία συμβατή αναπαράστασή τους, επιτρέποντας την αποστολή των δεδομένων μέσω δικτύου.

Βιβλιοθήκη Requests

Η βιβλιοθήκη `requests` χρησιμοποιήθηκε για την αποστολή αιτημάτων τύπου HTTP μεταξύ των κόμβων του συστήματος. Αυτή η βιβλιοθήκη απλοποιεί την επικοινωνία μέσω δικτύου, επιτρέποντας στους κόμβους να στέλνουν και να λαμβάνουν δεδομένα από άλλους κόμβους. Για παράδειγμα, όταν ένας κόμβος χρειάζεται να ενημερώσει τους υπόλοιπους για μια αλλαγή στη λίστα των κόμβων ή των αντικειμένων, χρησιμοποιεί την βιβλιοθήκη `requests` για να στείλει τα αντίστοιχα δεδομένα.

Βιβλιοθήκη Flask

Η βιβλιοθήκη `Flask` της γλώσσας προγραμματισμού Python χρησιμοποιήθηκε για τη δημιουργία ενός εξυπηρετητή ή αλλιώς `server` που διαχειρίζεται αιτήματα τύπου HTTP. Το `Flask` είναι ένα ελαφρύ και εύχρηστο εργαλείο που χρησιμοποιείται για την ανάπτυξη εφαρμογών διαδικτύου, το οποίο επιτρέπει τη δημιουργία διαδρομών (`endpoints`) για την εκτέλεση διαφόρων λειτουργιών. Στο πλαίσιο του συστήματος, η `Flask` χρησιμοποιήθηκε για να δημιουργηθούν διαδρομές που επιτρέπουν στους κόμβους να επικοινωνούν μεταξύ τους και να εκτελούν λειτουργίες όπως για παράδειγμα η προσθήκη και αφαίρεση κόμβων, η προσθήκη, η ανάκτηση και διαγραφή αντικειμένων, καθώς και η διαχείριση της κατάστασης του συστήματος.

Η βιβλιοθήκη Flask επιτρέπει τη δημιουργία διαδρομών ή αλλιώς endpoint που αντιστοιχούν σε συγκεκριμένες λειτουργίες. Κάθε διαδρομή συνδέεται με μια συνάρτηση που εκτελείται όταν ο εξυπηρετητής λαμβάνει ένα αίτημα τύπου HTTP στην αντίστοιχη διαδρομή ή endpoint. Για παράδειγμα, μία διαδρομή όπως για παράδειγμα `/add_node` μπορεί να χρησιμοποιηθεί για την προσθήκη ενός νέου κόμβου στο σύστημα, ενώ μια διαδρομή `/get_item` μπορεί να χρησιμοποιηθεί για την ανάκτηση ενός αντικειμένου από το σύστημα αποθήκευσης.

Η βιβλιοθήκη Flask υποστηρίζει διάφορες μεθόδους αιτημάτων τύπου HTTP, όπως οι GET, POST, PUT και DELETE. Στο σύστημα, οι μέθοδοι POST και GET χρησιμοποιήθηκαν ευρέως για την αποστολή και λήψη δεδομένων. Για παράδειγμα, ένα αίτημα POST μπορεί να χρησιμοποιηθεί για την προσθήκη ενός νέου κόμβου, ενώ ένα αίτημα GET μπορεί να χρησιμοποιηθεί για την προσπέλαση κάποιου αντικειμένου.

Όλες οι απαντήσεις στις αιτήσεις των κόμβων επιστρέφονται σε μορφή JSON. Αυτό διευκολύνει την κατανόηση και την επεξεργασία των αποτελεσμάτων αιτήσεων από άλλους κόμβους ή αιτήσεων από εξωτερικές εφαρμογές.

Βιβλιοθήκη OS

Η βιβλιοθήκη `os` της Python χρησιμοποιήθηκε για τη διαχείριση φακέλων και αρχείων στο σύστημα αρχείων των κόμβων. Στην υλοποίηση του συστήματος, η βιβλιοθήκη `os` χρησιμοποιήθηκε για τη δημιουργία του φακέλου αποθήκευσης δεδομένων για κάθε ξεχωριστό κόμβο, καθώς και για τη διαχείριση των αρχείων που περιέχουν τα αποθηκευμένα αντικείμενα στους κόμβους. Για παράδειγμα, η συνάρτηση `os.makedirs()` χρησιμοποιήθηκε για τη δημιουργία του φακέλου για κάθε κόμβο του συστήματος.

Βιβλιοθήκη Threading

Η βιβλιοθήκη `threading` της Python χρησιμοποιήθηκε για να επιτρέπει την ταυτόχρονη εκτέλεση πολλαπλών νημάτων (`threads`). Πιο συγκεκριμένα η βιβλιοθήκη `threading` χρησιμοποιήθηκε για να εξομοιώσει πολλαπλούς κόμβους ως ξεχωριστά νήματα σε ένα πρόγραμμα. Κάθε κόμβος του συστήματος εκτελείται ως ένα ξεχωριστό νήμα (`thread`). Αυτό σημαίνει ότι πολλοί κόμβοι μπορούν να λειτουργούν ταυτόχρονα στο ίδιο πρόγραμμα, χωρίς να χρειάζεται να εκτελούνται σε διαφορετικές διεργασίες ή ξεχωριστές μηχανές. Η εκτέλεση πολλαπλών κόμβων ως νήματα σε ένα πρόγραμμα απλοποιεί σημαντικά τις διαδικασίες δοκιμών και ανάπτυξης. Αντί να εκτελούνται πολλαπλές διεργασίες ή να χρησιμοποιούνται πολλαπλές μηχανές, όλοι οι κόμβοι μπορούν να προσομοιωθούν σε μία διεργασία. Αυτή η προσέγγιση διευκόλυνε τη δοκιμή και την εκτέλεση του συστήματος, καθώς επιτρέπει την προσομοίωση μιας ομάδας κόμβων σε ένα μόνο περιβάλλον εκτέλεσης.

6.3 Δομή και Λειτουργία του συστήματος

Το σύστημα αποθήκευσης δεδομένων που αναπτύσσεται στην παρούσα εργασία αποτελείται από πολλαπλούς κόμβους (Nodes), οι οποίοι συνδέονται μέσω δικτύου και συνεργάζονται για την αποθήκευση αντικειμένων. Όλοι οι κόμβοι της ομάδας είναι ισότιμοι, χωρίς την ύπαρξη κάποιου κεντρικού κόμβου ή κόμβου διαχειριστή, γεγονός που καθιστά το σύστημα αποκεντρωμένο. Οι κόμβοι έχουν τη δυνατότητα να επικοινωνούν μεταξύ τους, ανταλλάσσοντας πληροφορίες και δεδομένα, προκειμένου να ενημερώνουν τους υπόλοιπους για αλλαγές στη σύνθεση της ομάδας και να διασφαλίζουν την σωστή κατανομή και αποθήκευση των αντικειμένων στο σύστημα.

Όλοι οι κόμβοι του συστήματος εκτελούν τον ίδιο κώδικα, γεγονός που τους επιτρέπει να λειτουργούν αυτόνομα και να μπορούν να πραγματοποιήσουν οποιαδήποτε απαραίτητη ενέργεια ή λειτουργία. Συγκεκριμένα, κάθε κόμβος έχει τη δυνατότητα να προσθέτει ή να αφαιρεί άλλους κόμβους από το σύστημα, προσαρμόζοντας τη σύνθεση της ομάδας. Επιπλέον, μπορεί να εκτελεί λειτουργίες αποθήκευσης, ανάκτησης και διαγραφής αντικειμένων καθώς και να επικοινωνεί με άλλους κόμβους και να ανταλλάσει πληροφορίες σχετικά με αλλαγές ή μεταβολές στο σύστημα και τα αποθηκευμένα αντικείμενα.

6.4 Δεδομένα και μεταβλητές που διατηρεί κάθε κόμβος

Οι κόμβοι διατηρούν πληροφορίες που είναι απαραίτητες για τη λειτουργία τους, την επικοινωνία με τους υπόλοιπους κόμβους και τη σωστή διαχείριση των αποθηκευμένων

αντικειμένων. Αυτές οι πληροφορίες περιλαμβάνουν τόσο δεδομένα σχετικά με τον ίδιο τον κόμβο όσο και στοιχεία για τους υπόλοιπους κόμβους της ομάδας και τα αποθηκευμένα αντικείμενα.

- Τα στοιχεία του κόμβου: Κάθε κόμβος διατηρεί βασικές πληροφορίες για τον εαυτό του. Αυτές οι πληροφορίες περιλαμβάνουν το όνομα του κόμβου, την τιμή κατακερματισμού του κόμβου η οποία είναι ένα μοναδικό αναγνωριστικό για τον κόμβο, καθώς επίσης και τη διεύθυνση του κόμβου στο δίκτυο που χρησιμοποιείται για την επικοινωνία με τους υπόλοιπους κόμβους και για την εξυπηρέτηση αιτημάτων χρηστών.
- Λίστα όλων των κόμβων της ομάδας: Κάθε κόμβος διατηρεί μια λίστα που περιέχει όλους τους κόμβους της ομάδας, συμπεριλαμβανομένου και του εαυτού του. Για κάθε κόμβο στη λίστα αποθηκεύεται η τιμή κατακερματισμού του κόμβου, η οποία χρησιμοποιείται για την ταξινόμηση και την αναζήτηση των κόμβων στο σύστημα, το όνομα του κόμβου καθώς και τη διεύθυνση του κόμβου στο δίκτυο. Αυτή η λίστα επιτρέπει στον κάθε κόμβο να γνωρίζει ποιοι είναι οι υπόλοιποι κόμβοι της ομάδας και πώς να επικοινωνήσει μαζί τους.
- Λίστα των αποθηκευμένων αντικειμένων: Οι κόμβοι διατηρούν μια λίστα με τα αντικείμενα που είναι αποθηκευμένα στην ομάδα. Κάθε αντικείμενο περιλαμβάνει την τιμή κατακερματισμού του, η οποία χρησιμοποιείται για την ταξινόμηση και την αναζήτηση του αντικειμένου. Επίσης, περιλαμβάνει το κλειδί του αντικειμένου το οποίο είναι το μοναδικό αναγνωριστικό του αντικειμένου, όπως επίσης και τους κόμβους που έχουν αποθηκευμένο το αντικείμενο ή διατηρούν αντίγραφά του. Αυτή η λίστα εξασφαλίζει ότι ο κάθε κόμβος γνωρίζει ποια αντικείμενα είναι αποθηκευμένα στο σύστημα και σε ποιους κόμβους βρίσκονται και είναι διαθέσιμα.
- Τον αριθμό αντιγράφων που διατηρούνται: Ο αριθμός αντιγράφων καθορίζει σε πόσους κόμβους του συστήματος αποθηκεύεται κάθε αντικείμενο. Με άλλα λόγια, εκφράζει τον αριθμό των αντιγράφων που διατηρεί το σύστημα.

6.5 Λειτουργίες κάθε κόμβου

Αρχικοποίηση κόμβου:

Κατά την έναρξη λειτουργίας ενός κόμβου, εκτελείται η διαδικασία της αρχικοποίησής του. Πρόκειται για μια συνάρτηση που δέχεται ως παραμέτρους τα στοιχεία των κόμβων της ομάδας. Τα στοιχεία αυτά περιλαμβάνουν τα ονόματα των κόμβων, τη διεύθυνσή τους στο δίκτυο, καθώς και την τιμή κατακερματισμού τους για την τοποθέτησή τους στον δακτύλιο κατακερματισμού. Επιπλέον, ως παράμετροι δίνονται τα αντικείμενα που είναι ήδη αποθηκευμένα στην ομάδα του συστήματος αποθήκευσης ως μια λίστα. Η λίστα περιλαμβάνει ένα σύνολο αντικειμένων που αποτελούνται από το κλειδί τους που είναι και το μοναδικό χαρακτηριστικό τους, την τιμή κατακερματισμού τους, καθώς και τους κόμβους στους οποίους είναι αποθηκευμένα ή διατηρούνται αντίγραφά του.

Αρχικά, κατά τη διαδικασία αρχικοποίησης ενός κόμβου κατά την πρώτη του εκκίνηση, ο κόμβος δέχεται ως παράμετρο τα στοιχεία μιας ομάδας που περιλαμβάνει μόνο τον εαυτό του. Με άλλα λόγια, η ομάδα αποτελείται εξ ολοκλήρου από τον ίδιο τον κόμβο, χωρίς να περιλαμβάνει άλλα μέλη. Η λίστα των αντικειμένων κατά την αρχικοποίηση του κόμβου, κατά την πρώτη εκκίνηση της λειτουργίας του, είναι αρχικά άδεια. Επιπλέον, ορίζεται ο αριθμός των αντιγράφων που θα διατηρεί το σύστημα, όπως καθορίζεται κατά τη δημιουργία του. Με αυτόν τον τρόπο, ο κόμβος ξεκινά τη λειτουργία του ως αυτόνομη οντότητα, χωρίς να ανήκει σε κάποια ομάδα με άλλους κόμβους.

Στη συνέχεια, ο κόμβος μπορεί να λάβει αίτηση είτε για να προσκαλέσει έναν άλλο κόμβο και να σχηματίσουν μια νέα ομάδα, είτε για να ενταχθεί ο ίδιος σε μια ήδη υπάρχουσα ομάδα με άλλους κόμβους. Στην περίπτωση που ο κόμβος ζητηθεί να ενταχθεί σε μια υπάρχουσα ομάδα, η συνάρτηση αρχικοποίησης καλείται του ξανά από τον κόμβο που τον προσκαλεί. Κατά τη διάρκεια αυτής της διαδικασίας, ο κόμβος ενημερώνεται με τα στοιχεία όλων των κόμβων της ομάδας, καθώς και με τα αντικείμενα που είναι ήδη αποθηκευμένα στο σύστημα, καθώς και τον αριθμό αντιγράφων που διατηρούνται. Αυτή η ενημέρωση εξασφαλίζει ότι ο νέος κόμβος γνωρίζει την τρέχουσα κατάσταση της ομάδας στην οποία εντάσσεται.

Επιπλέον, στην περίπτωση που έχει ήδη σχηματιστεί μια ομάδα και πραγματοποιηθεί αίτηση σε έναν από τους κόμβους να προσθέσει έναν νέο κόμβο στην ομάδα, εκτός από τον νέο κόμβο που προστίθεται, ο κόμβος που αναλαμβάνει την προσθήκη καλεί τη συνάρτηση αρχικοποίησης και για τους υπόλοιπους κόμβους της ομάδας. Αυτό πραγματοποιείται για να αρχικοποιήσουν εκ νέου την κατάστασή τους και να ενημερώσουν την τοπική τους λίστα των κόμβων της ομάδας, συμπεριλαμβανομένου του νέου μέλους που προστέθηκε.

Προσθήκη νέου κόμβου:

Η διαδικασία προσθήκης ενός νέου κόμβου πραγματοποιείται μέσω μιας συνάρτησης που λαμβάνει ως είσοδο τα στοιχεία του νέου κόμβου, τα οποία περιλαμβάνουν το όνομα του κόμβου και τη διεύθυνση του κόμβου στο δίκτυο. Κατά την εκτέλεση της λειτουργίας της προσθήκης, υπολογίζεται η τιμή κατακερματισμού του νέου κόμβου που εισάγεται, ώστε να προσδιοριστεί η θέση του στον δακτύλιο κατακερματισμού. Τα στοιχεία του νέου κόμβου προστίθενται στην τοπική λίστα των κόμβων, η οποία ταξινομείται εκ νέου με βάση την τιμή κατακερματισμού των κόμβων.

Μόλις ο νέος κόμβος προστεθεί στην τοπική λίστα του κόμβου που έλαβε την αίτηση για την προσθήκη, ακολουθεί η διαδικασία ενημέρωσης όλων των υπόλοιπων κόμβων της ομάδας για την αλλαγή. Ο κόμβος που πραγματοποίησε την προσθήκη στέλνει σε κάθε μέλος της ομάδας, συμπεριλαμβανομένου και του νέου κόμβου, ένα αίτημα τύπου HTTP για την εκ νέου αρχικοποίηση τους με τη νέα σύσταση της ομάδας. Το αίτημα αυτό περιλαμβάνει την ενημερωμένη λίστα όλων των κόμβων (όπου συμπεριλαμβάνεται και ο νέος κόμβος), καθώς και τον αριθμό των αντιγράφων που διατηρεί το σύστημα. Επιπλέον, στον νέο κόμβο αποστέλλεται και η λίστα των αντικειμένων που είναι ήδη αποθηκευμένα στο σύστημα, ώστε να ενημερωθεί για την τρέχουσα κατάσταση των αποθηκευμένων δεδομένων.

Τέλος, αφού ολοκληρωθεί η αρχικοποίηση όλων των κόμβων, ακολουθεί η διαδικασία ανακατανομής των αποθηκευμένων αντικειμένων. Ο κόμβος που προσέθεσε το νέο μέλος στην ομάδα αποστέλλει σε κάθε κόμβο της ομάδας, συμπεριλαμβανομένου και του εαυτού του μια αίτηση τύπου HTTP, ενημερώνοντάς τους να επανυπολογίσουν την κατανομή των δεδομένων σύμφωνα με τη νέα δομή του δακτυλίου κατακερματισμού. Με την παραλαβή της αίτησης για ανακατανομή, κάθε κόμβος αναλαμβάνει να μεταφέρει συγκεκριμένα αντικείμενα που δεν είναι σωστά αποθηκευμένα στον κατάλληλο κόμβο της ομάδας.

Αφαίρεση κόμβου:

Η διαδικασία αφαίρεσης ενός κόμβου από την ομάδα πραγματοποιείται μέσω μιας συνάρτησης που λαμβάνει ως είσοδο το όνομα του κόμβου που πρόκειται να αφαιρεθεί. Κατά την εκτέλεση της διαδικασίας, ο κόμβος που έλαβε την αίτηση της αφαίρεσης αναζητά τον κόμβο που θα αφαιρεθεί στην τοπική του λίστα με τους κόμβους. Ο κόμβος αφαιρείται από τη τοπική λίστα και η λίστα ταξινομείται εκ νέου με βάση την τιμή κατακερματισμού των υπόλοιπων κόμβων.

Μόλις ο κόμβος αφαιρεθεί από την τοπική λίστα, ακολουθεί η διαδικασία ενημέρωσης όλων των υπόλοιπων κόμβων της ομάδας για την αλλαγή. Ο κόμβος που πραγματοποίησε την αφαίρεση στέλνει σε κάθε μέλος της ομάδας ένα αίτημα τύπου HTTP για την εκ νέου αρχικοποίηση τους με τη νέα σύσταση της ομάδας. Το αίτημα αυτό περιλαμβάνει την ενημερωμένη λίστα όλων των κόμβων, χωρίς να περιλαμβάνει τον κόμβο που αφαιρέθηκε. Επιπλέον, στον κόμβο που αφαιρέθηκε αποστέλλεται ένα αίτημα αρχικοποίησης που περιλαμβάνει μόνο τον εαυτό του ως μέλος της ομάδας και μια κενή λίστα αντικειμένων, ώστε να επανέλθει στην αρχική κατάσταση ενός αυτόνομου κόμβου.

Μετά την ενημέρωση όλων των κόμβων, ο κόμβος που αφαίρεσε το μέλος προχωρά στο τελευταίο βήμα που περιλαμβάνει την ανακατανομή των αποθηκευμένων αντικειμένων. Ο κόμβος στέλνει σε κάθε μέλος της ομάδας, και στον εαυτό του ένα αίτημα

τύπου HTTP ειδοποιώντας τους να επανυπολογίσουν την κατανομή των δεδομένων σύμφωνα με τη νέα δομή του δακτυλίου κατακερματισμού. Κάθε κόμβος, λαμβάνοντας το αίτημα και πραγματοποιεί τη διαδικασία ανακατανομής όπως ακριβώς συμβαίνει και στην νέου προσθήκη κόμβου στην ομάδα.

Ανάκτηση αντικειμένου από το σύστημα:

Η διαδικασία ανάκτησης ενός αντικειμένου από το σύστημα πραγματοποιείται μέσω μιας συνάρτησης που λαμβάνει ως είσοδο το κλειδί (key) του αντικειμένου που ζητείται. Αρχικά, υπολογίζεται η τιμή κατακερματισμού του κλειδιού, η οποία χρησιμοποιείται για τον προσδιορισμό των υπευθύνων κόμβων που διαθέτουν το αντικείμενο. Εάν ο τρέχων κόμβος είναι ένας από τους υπεύθυνους κόμβους, αναζητά το αντικείμενο στον τοπικό του χώρο αποθήκευσης και στη συνέχεια επιστρέφεται το αντικείμενο με τα δεδομένα του. Σε αντίθετη περίπτωση, εάν ο κόμβος που έλαβε το αίτημα ανάκτησης του αντικειμένου δεν είναι υπεύθυνος και δεν διατηρεί το αντικείμενο, ανακατευθύνει το αίτημα για ανάκτηση στον κατάλληλο υπεύθυνο κόμβο.

Είναι σημαντικό να σημειωθεί ότι ο κόμβος από τον οποίο θα γίνει η ανάκτηση του αντικειμένου προσδιορίζεται αποκλειστικά με βάση την τιμή κατακερματισμού του αντικειμένου. Δεν χρησιμοποιείται καμία πληροφορία από τις τοπικές λίστες των αντικειμένων που διατηρούνται από τους ξεχωριστούς κόμβους του συστήματος. Αυτή η προσέγγιση εξασφαλίζει ότι η διαδικασία ανάκτησης βασίζεται αποκλειστικά στη μέθοδο κατακερματισμού, χωρίς να απαιτείται επιπλέον πληροφορία.

Προσθήκη αντικειμένου στο σύστημα:

Η διαδικασία προσθήκης ενός νέου αντικειμένου στο σύστημα πραγματοποιείται μέσω μιας συνάρτησης που λαμβάνει ως είσοδο το κλειδί και τα δεδομένα του αντικειμένου που πρόκειται να αποθηκευτεί. Η διαδικασία ξεκινά με τον υπολογισμό της τιμής κατακερματισμού του κλειδιού του αντικειμένου. Η τιμή αυτή χρησιμοποιείται για την εύρεση των υπευθύνων κόμβων που θα διατηρήσουν αντίγραφο του αντικειμένου.

Σύμφωνα με τη μέθοδο του συνεπή κατακερματισμού (consistent hashing), για κάθε αντικείμενο που εισάγεται στο σύστημα, υπολογίζεται η τιμή κατακερματισμού του κλειδιού του. Στη συνέχεια, με βάση την ταξινομημένη λίστα των κόμβων που διατηρεί κάθε μέλος της ομάδας, προσδιορίζεται ο κόμβος στον οποίο αντιστοιχίζεται το αντικείμενο. Αυτός ο κόμβος είναι ο πρωτεύων υπεύθυνος κόμβος για το αντικείμενο. Ωστόσο, με την τροποποίηση της μεθόδου του συνεπή κατακερματισμού για την υποστήριξη πολλαπλών αντιγράφων (replication), επιλέγονται και επόμενοι κόμβοι στον δακτύλιο που θα διατηρήσουν αντίγραφα του συγκεκριμένου αντικειμένου. Ο αριθμός των επιπλέον κόμβων που επιλέγονται βασίζεται στον αριθμό των αντιγράφων που διατηρεί το σύστημα, ο οποίος ορίζεται ως μια τοπική μεταβλητή σε κάθε κόμβο κατά την διαδικασία αρχικοποίησης που περιγράφεται παραπάνω.

Μόλις βρεθούν οι υπεύθυνοι κόμβοι για την αποθήκευση του αντικειμένου, το αντικείμενο αποστέλλεται σε κάθε έναν από αυτούς μέσω μιας κλήσης τύπου HTTP. Η κλήση αυτή καλεί

τη συνάρτηση που είναι χρησιμοποιείται για την αποθήκευση του αντικειμένου στον τοπικό χώρο αποθήκευσης του κάθε κόμβου. Αυτό εξασφαλίζει ότι το αντικείμενο αποθηκεύεται σε όλους τους υπεύθυνους κόμβους που αντιστοιχίζεται

Επιπλέον, όλοι οι κόμβοι της ομάδας ενημερώνονται για την προσθήκη του νέου αντικειμένου στο σύστημα. Αυτή η ενημέρωση πραγματοποιείται μέσω μιας άλλης κλήσης τύπου HTTP, η οποία στέλνει σε κάθε κόμβο της ομάδας και περιέχει τις πληροφορίες του νέου αντικειμένου που εισάγεται. Οι πληροφορίες αυτές περιλαμβάνουν το κλειδί του αντικειμένου, την τιμή κατακερματισμού του κλειδιού του αντικειμένου καθώς και τους κόμβους που διατηρούν αντίγραφο του αντικειμένου. Οι κόμβοι ενημερώνουν τις τοπικές τους λίστες με τα αποθηκευμένα αντικείμενα, προσθέτοντας το νέο αντικείμενο και τις σχετικές πληροφορίες που σχετίζονται με αυτό. Αυτή η ενημέρωση εξασφαλίζει ότι όλοι οι κόμβοι έχουν μια συνεπή εικόνα της τρέχουσας κατάστασης των αντικειμένων του συστήματος.

Αφαίρεση αντικειμένου από το σύστημα:

Η διαδικασία αφαίρεσης ενός αντικειμένου από το σύστημα πραγματοποιείται μέσω μιας συνάρτησης που λαμβάνει ως είσοδο το κλειδί του αντικειμένου που πρόκειται να αφαιρεθεί. Αρχικά, υπολογίζεται η τιμή κατακερματισμού του κλειδιού, η οποία χρησιμοποιείται για τον υπολογισμό των υπευθύνων κόμβων που διατηρούν το αντικείμενο. Μόλις προσδιοριστούν οι υπεύθυνοι κόμβοι, στέλνεται σε κάθε έναν από αυτούς μια κλήση τύπου HTTP για την αφαίρεση του αντικειμένου από τον τοπικό τους χώρο αποθήκευσης.

Επιπλέον, όλοι οι κόμβοι της ομάδας ενημερώνονται για την αφαίρεση του αντικειμένου. Αυτή η ενημέρωση πραγματοποιείται μέσω μιας άλλης κλήσης τύπου HTTP, η οποία στέλνει σε κάθε κόμβο τις πληροφορίες του αντικειμένου που αφαιρέθηκε. Οι κόμβοι ενημερώνουν τις τοπικές τους λίστες με τα αποθηκευμένα αντικείμενα, αφαιρώντας το αντικείμενο και τις σχετικές πληροφορίες.

Προσθήκη αντικειμένου σε κόμβο:

Η διαδικασία προσθήκης ενός αντικειμένου σε έναν συγκεκριμένο κόμβο του συστήματος πραγματοποιείται μέσω μιας συνάρτησης που λαμβάνει ως είσοδο το κλειδί και τα δεδομένα του αντικειμένου που πρόκειται να αποθηκευτεί. Αρχικά, δημιουργείται η διαδρομή του αρχείου στον τοπικό χώρο αποθήκευσης του κόμβου, όπου θα αποθηκευτεί το αντικείμενο. Στη συνέχεια, τα δεδομένα του αντικειμένου γράφονται στο αρχείο που αντιστοιχεί στο κλειδί. Η διαδικασία ολοκληρώνεται με την επιστροφή ενός μηνύματος επιτυχίας που επιβεβαιώνει την προσθήκη του αντικειμένου στον κόμβο.

Σε αυτή τη διαδικασία, δεν υπολογίζεται κάποια τιμή κατακερματισμού του αντικειμένου ούτε πραγματοποιούνται επιπλέον λειτουργίες. Αυτό οφείλεται στο γεγονός ότι η συγκεκριμένη λειτουργία είναι σχεδιασμένη να καλείται από άλλο κόμβο του συστήματος, ο οποίος έχει

ήδη υπολογίσει τους υπεύθυνους κόμβους και έχει πραγματοποιήσει όλες τις απαραίτητες ενέργειες. Η λειτουργία αυτή αποσκοπεί αποκλειστικά στην αποθήκευση των δεδομένων του αντικειμένου στον τοπικό αποθηκευτικό χώρο του κόμβου.

Αφαίρεση αντικειμένου από κόμβο:

Η διαδικασία αφαίρεσης ενός αντικειμένου από έναν συγκεκριμένο κόμβο πραγματοποιείται μέσω μιας συνάρτησης που λαμβάνει ως είσοδο το κλειδί του αντικειμένου που πρόκειται να αφαιρεθεί. Αρχικά, ελέγχεται αν το αντικείμενο υπάρχει στον τοπικό χώρο αποθήκευσης του κόμβου, αναζητώντας το αρχείο που αντιστοιχεί στο κλειδί. Εάν το αρχείο που αντιστοιχεί στο αντικείμενο βρεθεί διαγράφεται από τον αποθηκευτικό χώρο του κόμβου.

Παρόμοια με τη διαδικασία προσθήκης, δεν υπολογίζεται κάποια τιμή κατακερματισμού ούτε πραγματοποιούνται επιπλέον λειτουργίες. Η λειτουργία αυτή είναι σχεδιασμένη να καλείται από άλλο κόμβο του συστήματος, ο οποίος έχει ήδη προσδιορίσει τους υπεύθυνους κόμβους και έχει πραγματοποιήσει τις απαραίτητες ενέργειες. Η συγκεκριμένη λειτουργία αποσκοπεί αποκλειστικά στη διαγραφή ενός αντικειμένου από τον τοπικό αποθηκευτικό χώρο ενός κόμβου.

Ενημέρωση της τοπικής λίστας των αποθηκευμένων αντικειμένων:

Η διαδικασία ενημέρωσης της τοπικής λίστας αντικειμένων ενός κόμβου πραγματοποιείται μέσω μιας συνάρτησης που λαμβάνει ως είσοδο τον τύπο της λειτουργίας (operation) που πρόκειται να εκτελεστεί, καθώς και τα δεδομένα ή τις πληροφορίες που σχετίζονται με αυτή τη λειτουργία. Οι διαθέσιμες λειτουργίες είναι οι εξής: ορισμός όλων των αντικειμένων (set_all), προσθήκη αντικειμένου (add), αφαίρεση αντικειμένου (remove) και ενημέρωση αντικειμένου (update). Κάθε λειτουργία εκτελεί διαφορετικές ενέργειες για τη διαχείριση της τοπικής λίστας αντικειμένων που διατηρεί ο κόμβος.

Η συγκεκριμένη συνάρτηση έχει υλοποιηθεί με σκοπό να καλείται από άλλους κόμβους του συστήματος, όταν προστίθενται, αφαιρούνται ή ενημερώνονται αντικείμενα στο σύστημα. Αυτό εξασφαλίζει ότι όλοι οι κόμβοι διατηρούν μια συνεπή και ενημερωμένη εικόνα της κατάστασης των αποθηκευμένων δεδομένων.

- Λειτουργία ορισμού όλων των αντικειμένων (set_all): Αυτή η λειτουργία χρησιμοποιείται για να ορίσει την πλήρη λίστα αντικειμένων του κόμβου. Τα δεδομένα που λαμβάνονται περιλαμβάνουν μια λίστα με όλα τα αντικείμενα, τα οποία

αποτελούνται από την τιμή κατακερματισμού τους, το κλειδί τους καθώς και τους υπεύθυνους κόμβους στους οποίους διατηρούνται και είναι αποθηκευμένα. Η λίστα αντικειμένων του κόμβου ενημερώνεται με τα νέα δεδομένα και ταξινομείται με βάση την τιμή κατακερματισμού των αντικειμένων. Η συγκεκριμένη λειτουργία έχει υλοποιηθεί για την αρχικοποίηση της λίστας αποθηκευμένων αντικειμένων κατά τη διαδικασία αρχικοποίησης ενός κόμβου.

- **Λειτουργία προσθήκης αντικειμένου (add):** Αυτή η λειτουργία χρησιμοποιείται για την προσθήκη ενός νέου αντικειμένου στη λίστα. Τα δεδομένα που λαμβάνονται περιλαμβάνουν την τιμή κατακερματισμού, το κλειδί του αντικειμένου και τους υπεύθυνους κόμβους για το αντικείμενο. Το αντικείμενο εισάγεται στην τοπική λίστα, ενώ στη συνέχεια η λίστα ταξινομείται με βάση την τιμή κατακερματισμού των αντικειμένων που περιέχει. Η λειτουργία αυτή καλείται όταν προστίθεται και αποθηκεύεται ένα νέο αντικείμενο στο σύστημα
- **Λειτουργία αφαίρεσης αντικειμένου (remove):** Αυτή η λειτουργία χρησιμοποιείται για την αφαίρεση ενός αντικειμένου από τη λίστα. Τα δεδομένα που λαμβάνονται περιλαμβάνουν την τιμή κατακερματισμού του αντικειμένου και το κλειδί του. Εάν το αντικείμενο βρεθεί στη λίστα αφαιρείται. Η λειτουργία αυτή καλείται όταν αφαιρείται ή διαγράφεται ένα αντικείμενο από το σύστημα
- **Λειτουργία ενημέρωσης αντικειμένου (update):** Αυτή η λειτουργία χρησιμοποιείται για την ενημέρωση των πληροφοριών ενός αντικειμένου που υπάρχει ήδη στη λίστα. Τα δεδομένα που λαμβάνονται περιλαμβάνουν την τιμή κατακερματισμού, το κλειδί καθώς και τους υπεύθυνους κόμβους του αντικειμένου. Εάν το αντικείμενο βρεθεί στη λίστα, οι πληροφορίες του ενημερώνονται με τα νέα δεδομένα που λαμβάνονται. Η λειτουργία αυτή καλείται κυρίως σε περιπτώσεις όπου ένας κόμβος προστίθεται ή αφαιρείται από το σύστημα αποθήκευσης. Σε τέτοιες περιπτώσεις πραγματοποιείται ανακατανομή των αντικειμένων μεταξύ των ξεχωριστών κόμβων τις ομάδας. Λόγω αυτού οι υπεύθυνοι κόμβοι που διατηρούν τα αντικείμενα ενδέχεται να αλλάξουν και χρειάζεται να ενημερωθούν. Μέσω αυτής της λειτουργίας, είναι δυνατή η ενημέρωση των υπευθύνων κόμβων που διατηρούν αντίγραφα ενός αντικειμένου στις τοπικές λίστες των κόμβων της ομάδας.

Ανακατανομή των αποθηκευμένων αντικειμένων

Η διαδικασία ανακατανομής και αντιγραφής των αποθηκευμένων αντικειμένων είναι μια πολύ σημαντική λειτουργία του συστήματος αποθήκευσης, η οποία πραγματοποιείται όταν αλλάζουν οι κόμβοι της ομάδας και η δομή του δακτυλίου κατακερματισμού, είτε λόγω προσθήκης ενός νέου κόμβου, είτε λόγω αφαίρεσης ενός υπάρχοντος κόμβου από ομάδα. Στόχος της συγκεκριμένης διαδικασίας είναι να εξασφαλιστεί ότι κάθε αντικείμενο είναι αποθηκευμένο και διατηρείται στους σωστούς κόμβους του συστήματος, μετά από μεταβολές στη σύστασή του, σύμφωνα με τη συνάρτηση κατακερματισμού και το δακτύλιο κατακερματισμού.

Όπως έχει ήδη αναφερθεί, σε περιπτώσεις όπου προστίθενται ή αφαιρούνται κόμβοι από την ομάδα του συστήματος, απαιτείται ανακατανομή ή μετακίνηση ορισμένων αποθηκευμένων αντικειμένων σε άλλους κόμβους του συστήματος. Αυτό συμβαίνει επειδή οι κόμβοι στους οποίους αποθηκεύονται και διατηρούνται τα αντικείμενα επιλέγονται βάσει του αποτελέσματος της συνάρτησης κατακερματισμού των αντικειμένων και των κόμβων. Σε περιπτώσεις όπου υπάρχουν αλλαγές στη σύνθεση των κόμβων του συστήματος, η αντιστοίχιση των αντικειμένων στους κόμβους μεταβάλλεται, γεγονός που απαιτεί την προσαρμογή της κατανομής τους. Έτσι, τα αντικείμενα που ήταν αποθηκευμένα σε έναν κόμβο ενδέχεται να πρέπει να μεταφερθούν σε κάποιο άλλο κόμβο, προκειμένου να διασφαλιστεί η σωστή λειτουργία του συστήματος αποθήκευσης δεδομένων.

Όπως έχει ήδη αναφερθεί, όταν προστίθενται ή αφαιρούνται κόμβοι από την ομάδα του συστήματος, απαιτείται ανακατανομή ή μετακίνηση ορισμένων αποθηκευμένων αντικειμένων σε άλλους κόμβους της ομάδας. Αυτό συμβαίνει επειδή οι κόμβοι στους οποίους αποθηκεύονται τα αντικείμενα επιλέγονται βάσει του αποτελέσματος της συνάρτησης κατακερματισμού, η οποία βασίζεται τόσο στα κλειδιά των αντικειμένων όσο και στις τιμές κατακερματισμού των κόμβων. Όταν αλλάζει η σύνθεση της ομάδας, δηλαδή όταν προστίθεται ή αφαιρείται ένας κόμβος στο σύστημα, η αντιστοίχιση των αντικειμένων στους κόμβους αλλάζει επίσης. Αυτή η αλλαγή οδηγεί στην ανάγκη της ανακατανομής των αποθηκευμένων αντικειμένων, ώστε κάθε αντικείμενο να αποθηκεύεται και να διατηρείται στους σωστούς υπεύθυνους κόμβους σύμφωνα με την ενημερωμένη δομή του δακτυλίου κατακερματισμού. Επομένως, τα αντικείμενα που ήταν αποθηκευμένα σε έναν συγκεκριμένο κόμβο μπορεί να χρειαστεί να μεταφερθούν σε άλλους κόμβους, προκειμένου να εξασφαλιστεί η σωστή λειτουργία του συστήματος αποθήκευσης δεδομένων.

Βασική λειτουργία της διαδικασίας ανακατανομής των δεδομένων:

Η διαδικασία ανακατανομής των δεδομένων αποτελείται από μια σειρά βημάτων ως εξής:

1. Εντοπισμός των αντικειμένων προς μετακίνηση

Κάθε κόμβος εντοπίζει τα αντικείμενα για τα οποία ο ίδιος είναι υπεύθυνος, δηλαδή εκείνα τα αντικείμενα των οποίων η τιμή κατακερματισμού τους βρίσκεται μεταξύ της δικής του τιμής και της τιμής του επόμενου γειτονικού κόμβου στον δακτύλιο κατακερματισμού του συστήματος. Τα αντικείμενα αυτά ανήκουν στον συγκεκριμένο κόμβο, ο οποίος είναι υπεύθυνος για τη αποθήκευση τους και τη διαχείρισή τους.

2. Ευρεση των υπευθύνων κόμβων για κάθε αντικείμενο:

Για κάθε αντικείμενο που έχει εντοπιστεί στο προηγούμενο βήμα, υπολογίζεται το νέο σύνολο των υπευθύνων κόμβων με βάση τον ενημερωμένο δακτύλιο κατακερματισμού. Αν το νέο σύνολο των υπευθύνων κόμβων διαφέρει από το προηγούμενο, σημαίνει ότι το αντικείμενο πρέπει να μετακινηθεί. Αυτό συμβαίνει επειδή η προσθήκη ή η αφαίρεση ενός κόμβου αλλάζει την κατανομή των τιμών κατακερματισμού στον δακτύλιο, οδηγώντας σε αλλαγές στην αντιστοίχιση των αντικειμένων στους κόμβους του συστήματος.

3. Μετακίνηση αντικειμένων:

Για κάθε αντικείμενο που χρειάζεται να μετακινηθεί, προσδιορίζονται δύο κύρια σύνολα κόμβων. Το πρώτο σύνολο περιλαμβάνει τους κόμβους που διατηρούν το συγκεκριμένο αντικείμενο, αλλά λόγω της μεταβολής στη δομή του συστήματος, δεν είναι πλέον υπεύθυνοι για αυτό και δεν πρέπει να διατηρούν αποθηκευμένο αντίγραφο του. Το συγκεκριμένο σύνολο υπάρχει αποθηκευμένο τοπικά σε κάθε κόμβο της ομάδας στην λίστα των αποθηκευμένων αντικειμένων που διατηρούνται. Το δεύτερο σύνολο περιλαμβάνει τους κόμβους που είναι πλέον υπεύθυνοι για το αντικείμενο και πρέπει να αποθηκεύσουν ένα αντίγραφο του αντικειμένου. Με βάση αυτά τα δύο σύνολα, υπολογίζεται ο κόμβος από τον οποίο θα διαγραφεί το αντικείμενο και ο νέος κόμβος στον οποίο θα αποθηκευτεί, δηλαδή πραγματοποιείται μια μετακίνηση του αντικειμένου από τον ένα κόμβο στον άλλο. Οι υπόλοιποι κόμβοι που διατηρούν αντίγραφο του αντικειμένου μπορούν να χρησιμοποιηθούν για την ανάκτηση του κατά τη διαδικασία της μετακίνησης.

Συγκεκριμένα, για τη μετακίνηση ενός αντικειμένου, ο υπεύθυνος κόμβος βρίσκει τον κόμβο από τον οποίο το αντικείμενο πρέπει να διαγραφεί και του στέλνει αίτηση διαγραφής. Σε περίπτωση που ο κόμβος από τον οποίο πρέπει να διαγραφεί το αντικείμενο έχει ήδη αφαιρεθεί από την ομάδα, η αίτηση διαγραφής δεν πραγματοποιεί καμία ενέργεια. Στη συνέχεια, στέλνεται αίτηση ανάκτησης του αντικειμένου σε άλλους κόμβους που το έχουν αποθηκευμένο. Τέλος, το αντικείμενο αποθηκεύεται στον νέο κόμβο που είναι πλέον υπεύθυνος για την αποθήκευση ενός αντιγράφου του.

4. Ενημέρωση της κατάστασης:

Κατά την διαδικασία της ανακατανομής και της μετακίνησης των αντικειμένων, οι υπεύθυνοι κόμβοι για κάθε αποθηκευμένο αντικείμενο ενημερώνεται στις τοπικές λίστες των ξεχωριστών μελών της ομάδας. Αυτό πραγματοποιείται όταν ολοκληρωθεί η μετακίνηση ενός αντικειμένου, όπου στέλνεται αίτημα για την ενημέρωση του αντικειμένου στους υπόλοιπους κόμβους του συστήματος.

ΚΕΦΑΛΑΙΟ 7 Βελτιώσεις και μελλοντικές προσθήκες

7.1 Συνεπής κατακερματισμός με τη χρήση εικονικών κόμβων

Το σύστημα αποθήκευσης που υλοποιήθηκε στην παρούσα εργασία βασίζεται στη μέθοδο του συνεπής κατακερματισμού (consistent hashing) και υποστηρίζει τη διατήρηση πολλαπλών αντιγράφων των αποθηκευμένων αντικειμένων. Αυτή η προσέγγιση εξασφαλίζει την αξιοπιστία, την διασφάλιση της υψηλής διαθεσιμότητας των αποθηκευμένων δεδομένων καθώς και την αποδοτικότητα του συστήματος, καθώς επιτρέπει την αποθήκευση και ανάκτηση δεδομένων με ελάχιστες μετακινήσεις όταν προστίθενται ή αφαιρούνται κόμβοι. Ωστόσο, υπάρχουν αρκετές βελτιώσεις και πιθανές μελλοντικές προσθήκες που θα μπορούσαν να ενισχύσουν περαιτέρω την απόδοση του συστήματος, ιδιαίτερα σε ζητήματα που αφορούν την ομοιόμορφη κατανομή των δεδομένων στους κόμβους και την ισορροπία του φορτίου. Μία από τις πιο σημαντικές βελτιώσεις που θα μπορούσαν να εφαρμοστούν είναι η χρήση εικονικών κόμβων ή αλλιώς Virtual Nodes ή VNodes.

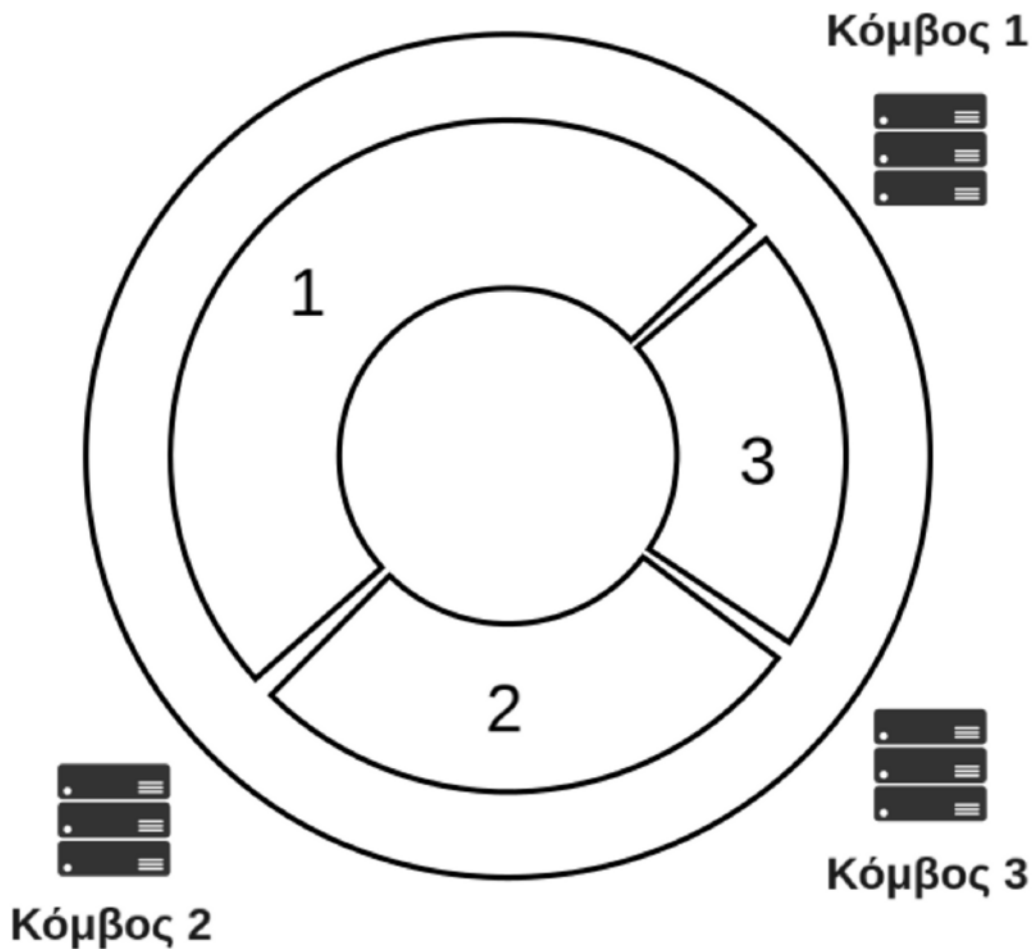
Τα Virtual Nodes (VNodes) αποτελούν μια τεχνική που χρησιμοποιείται στο συνεπή κατακερματισμό για τη βελτίωση της ομοιομορφίας στην κατανομή των δεδομένων και την εξισορρόπηση του φορτίου μεταξύ των φυσικών κόμβων του συστήματος. Σε αντίθεση με την παραδοσιακή μέθοδο κατανομής, όπου κάθε φυσικός κόμβος αντιστοιχίζεται σε μία μόνο θέση στον νοητό δακτύλιο κατακερματισμού, τα VNodes επιτρέπουν τη δημιουργία πολλαπλών εικονικών αντιγράφων για κάθε φυσικό κόμβο. Κάθε εικονικό αντίγραφο

τοποθετείται σε διαφορετικές θέσεις στον δακτύλιο, δημιουργώντας έτσι μια πιο ομοιόμορφη και ευέλικτη κατανομή των κόμβων στο χώρο κατακερματισμού.

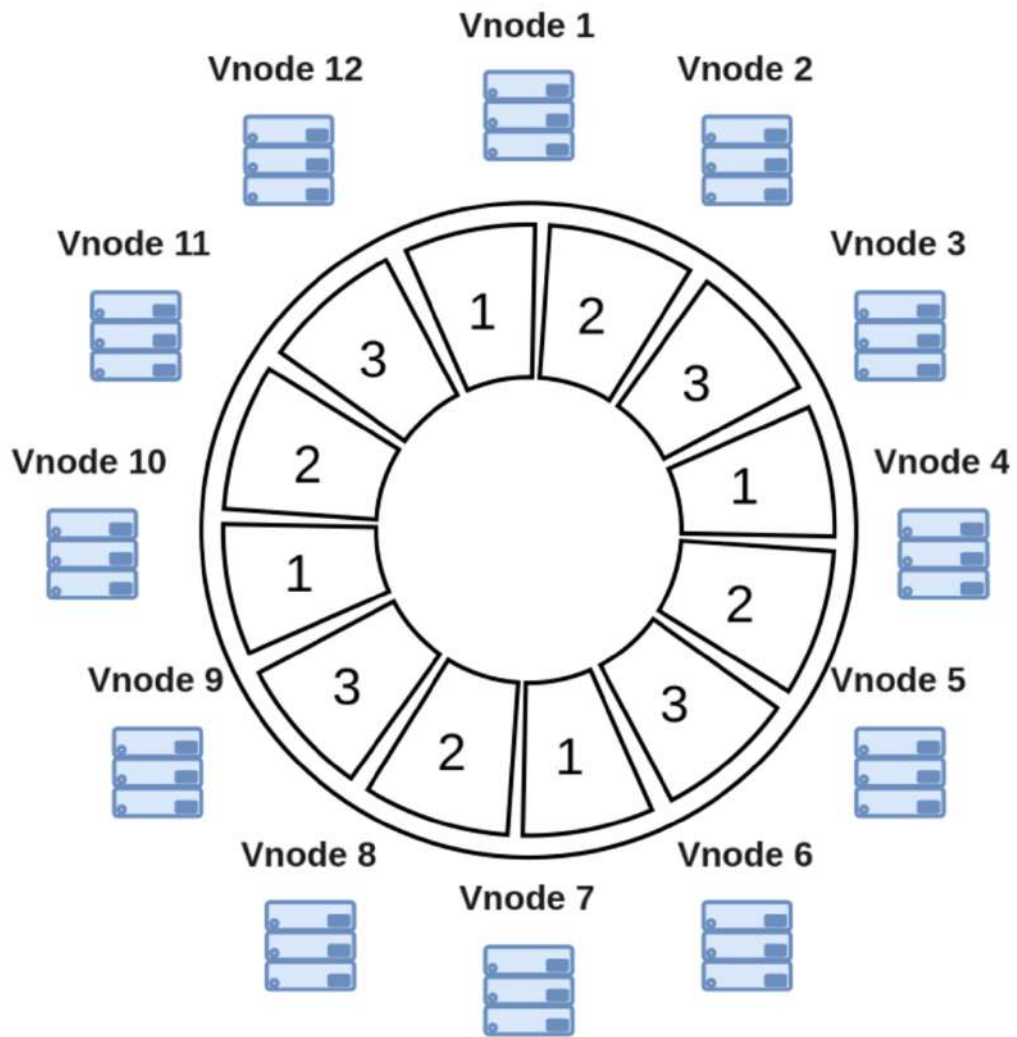
Η χρήση εικονικών κόμβων προσφέρει αρκετά πλεονεκτήματα. Πρώτον, βελτιώνει σημαντικά την ισορροπία φόρτου στους ξεχωριστούς κόμβους του συστήματος. Στο βασικό συνεπή κατακερματισμό, εάν οι κόμβοι έχουν μεγάλες διαφορές στις τιμές κατακερματισμού τους αυτό συνεπάγεται τη μη ομοιόμορφη κατανομή τους στον δακτύλιο κατακερματισμού. Συμφωνα με το γεγονός αυτό μπορεί να προκύψουν ανισορροπίες και ανομοιομορφία στην κατανομή των δεδομένων, με ορισμένους κόμβους να φορτώνονται υπερβολικά δημιουργώντας σημεία υψηλής συγκέντρωσης δεδομένων (hotspots) και άλλους να μην αξιοποιούνται τόσο πού. Με τη χρήση των εικονικών κόμβων ή VNodes, η κατανομή των δεδομένων γίνεται πιο ομοιόμορφη, καθώς κάθε φυσικός κόμβος αναλαμβάνει πολλές, μικρότερες περιοχές του δακτυλίου, μειώνοντας έτσι την πιθανότητα δημιουργίας σημείων υψηλής συγκέντρωσης.

7.2 Πώς Λειτουργούν οι εικονικοί κόμβοι

1. Δημιουργία Εικονικών Κόμβων: Κάθε φυσικός κόμβος του συστήματος δημιουργεί πολλαπλά εικονικά αντίγραφα (Virtual Nodes - VNodes), τα οποία τοποθετούνται σε διάφορες θέσεις στον δακτύλιο κατακερματισμού. Κάθε VNode αντιπροσωπεύει μια μικρή περιοχή του δακτυλίου και είναι υπεύθυνο για την αποθήκευση και τη διαχείριση των αντικειμένων που ανήκουν σε αυτήν την περιοχή. Ωστόσο, όλοι οι εικονικοί κόμβοι αναφέρονται σε έναν συγκεκριμένο φυσικό κόμβο, ο οποίος είναι υπεύθυνος για την αποθήκευση και τη διαχείριση των αντικειμένων σε όλες τις περιοχές που αντιστοιχούν στα VNodes του. Με αυτόν τον τρόπο, ο κάθε φυσικός κόμβος αναλαμβάνει την ευθύνη για πολλαπλές μικρές περιοχές του δακτυλίου, αντί για μία μόνο μεγάλη συνεχόμενη περιοχή, όπως συμβαίνει στο βασικό συνεπή κατακερματισμό.
2. Αντιστοίχιση των αντικειμένων: Όταν εισάγεται ένα νέο αντικείμενο για αποθήκευση στο σύστημα, υπολογίζεται η τιμή κατακερματισμού του αντικειμένου και αντιστοιχίζεται στον πλησιέστερο εικονικό κόμβο του δακτυλίου, όπως και στη παραδοσιακή μέθοδο του συνεπή κατακερματισμού. Ωστόσο ο φυσικός κόμβος που αντιστοιχεί στον εικονικό κόμβο που ανατέθηκε το αντικείμενο, αναλαμβάνει την αποθήκευση και τη διαχείριση του συγκεκριμένου αντικειμένου.



Εικόνα 7.2.1: Στην παραπάνω εικόνα οπτικοποιείται ο διαμοιρασμός ή αλλιώς ο διαχωρισμός των τμημάτων του δακτυλίου κατακερματισμού σε ένα σύστημα αποθήκευσης που λειτουργεί με τη κλασική μέθοδο του συνεπή κατακερματισμού. Όπως φαίνεται στο σχήμα κάθε κόμβος του συστήματος είναι υπεύθυνος για ένα συνεχόμενο τμήμα του δακτυλίου ή του χώρου κατακερματισμού. Στο συγκεκριμένο παράδειγμα ο κόμβος 3 είναι υπεύθυνος για την περιοχή του δακτυλίου μεταξύ των κόμβων 1 και 3, ο κόμβος 2 για την περιοχή του δακτυλίου μεταξύ των κόμβων 2 και 3 ενώ ο κόμβος 1 για την περιοχή του δακτυλίου μεταξύ των κόμβων 1 και 2. Όπως παρουσιάζεται ο κόμβος 1 έχει μεγαλύτερο τμήμα του δακτυλίου για το οποίο είναι υπεύθυνος σε σύγκριση με τους άλλους δύο κόμβους.



Κόμβος 1



Vnode 1



Vnode 4



Vnode 7



Vnode 10



Κόμβος 2



Vnode 2



Vnode 5



Vnode 8



Vnode 11



Κόμβος 3



Vnode 3



Vnode 6



Vnode 9



Vnode 12



Εικόνα 7.2.2: Στην παραπάνω εικόνα παρουσιάζεται ο διαμοιρασμός των τμημάτων του δακτυλίου κατακερματισμού σε ένα σύστημα αποθήκευσης που χρησιμοποιεί τη μέθοδο του συνεπή κατακερματισμού με τη χρήση εικονικών κόμβων ή αλλιώς VNodes. Σε αυτή τη περίπτωση, ο κύκλος κατακερματισμού χωρίζεται σε πολλαπλά ισομερή τμήματα και κάθε κόμβος αναλαμβάνει την ευθύνη για ένα σύνολο αυτών των τμημάτων. Αντί να αναλαμβάνει ευθύνη για ένα μόνο συνεχόμενο τμήμα του χώρου, κάθε κόμβος διαχειρίζεται περισσότερα από ένα τμήματα. Όπως φαίνεται στο σχήμα του παραδείγματος, η κατανομή των τμημάτων του δακτυλίου μεταξύ των κόμβων είναι πιο ισότιμη και παρατηρείται μεγαλύτερη ισορροπία σε σχέση με την κλασική μέθοδο.

7.3 Πλεονεκτήματα της χρήσης εικονικών κόμβων

- Καλύτερη και πιο ισορροπημένη κατανομή δεδομένων: Στο βασικό συνεπή κατακερματισμό, αν οι κόμβοι έχουν μεγάλες διαφορές στις θέσεις τους στον δακτύλιο, μπορεί να προκύψουν ανισορροπίες στην κατανομή των δεδομένων. Η μέθοδος του συνεπή κατακερματισμού με τη χρήση εικονικών κόμβων επιτυγχάνει καλύτερη, πιο ομοιόμορφη και πιο αποδοτική κατανομή δεδομένων τους επιμέρους κόμβους του συστήματος
- Μείωση των σημείων υψηλής συγκέντρωσης: Σε ένα σύστημα αποθήκευσης που λειτουργεί με τη μέθοδο του συνεπή κατακερματισμού και ειδικά σε περιπτώσεις που περιέχει μικρό αριθμό κόμβων, ορισμένοι μπορεί να συγκεντρώνουν υπερβολικά μεγάλο αριθμό αντικειμένων, δημιουργώντας σημεία υψηλής συγκέντρωσης hotspots. Το γεγονός αυτό παρουσιάζει σημεία συμφόρησης στο σύστημα και κακή αξιοποίηση των πόρων του. Με τη χρήση των εικονικών κόμβων, κάθε φυσικός κόμβος είναι υπεύθυνος για πολλές, μικρότερες περιοχές του δακτυλίου, μειώνοντας την πιθανότητα υπερφόρτωσης μεμονωμένων κόμβων.

Στην παρούσα εργασία πραγματοποιήθηκε η εξέταση των βασικών μεθόδων και τεχνικών κατανεμημένης αποθήκευσης δεδομένων σε δίκτυα υπολογιστών, καθώς και η ανάπτυξη ενός συστήματος αποθήκευσης που βασίζεται στην κατανομή των δεδομένων σε πολλαπλούς αυτόνομους κόμβους. Οι βασικοί στόχοι της εργασίας περιλαμβάνουν την συνεργατική λειτουργία πολλαπλών αυτόνομων κόμβων, την ομοιόμορφη και ισότιμη κατανομή των δεδομένων στο σύστημα, την εξισορρόπηση του φορτίου μεταξύ των κόμβων, την δυνατότητα κλιμάκωσης του συστήματος για την προσαρμογή σε αυξανόμενες ανάγκες πόρων και αποθηκευτικού χώρου, την δυνατότητα προσαρμογής του συστήματος σε δυναμικά περιβάλλοντα, όπου οι κόμβοι μπορούν να προστίθενται ή να αφαιρούνται συχνά, καθώς και την διασφάλιση της αξιοπιστίας του συστήματος και την ανοχή σε βλάβες κόμβων, με στόχο την επίτευξη υψηλής διαθεσιμότητας των αποθηκευμένων δεδομένων.

Για την επίτευξη αυτών των στόχων, παρουσιάστηκαν και αναλύθηκαν διάφορες μέθοδοι κατανεμημένης αποθήκευσης, όπως η κυκλική κατανομή (Round Robin), η στατικής κατανομής (Static Partitioning), το υπόλοιπο κατακερματισμού (Hash Modulo) καθώς και η Αντιγραφή δεδομένων (Replication) σε πολλαπλούς κόμβους. Ωστόσο, η μέθοδος που επιλέχθηκε για την υλοποίηση του συστήματος ήταν η μέθοδος του συνεπή κατακερματισμού (Consistent Hashing), η οποία προσφέρει σημαντικά πλεονεκτήματα σε σχέση με τις άλλες μεθόδους. Η επιλογή αυτή βασίστηκε στην ικανότητα του συνεπή κατακερματισμού να εξασφαλίζει ομοιόμορφη κατανομή των δεδομένων, εξισορρόπηση του φορτίου και ευελιξία για προσαρμογή σε δυναμικά περιβάλλοντα.

Μέσω της υλοποίησης του συστήματος με βάση τον συνεπή κατακερματισμό, επιτεύχθηκε μια ομοιόμορφη και ισορροπημένη κατανομή των δεδομένων στους επιμέρους κόμβους, καθώς και η δυνατότητα προσαρμογής του συστήματος σε δυναμικά περιβάλλοντα, όπου οι κόμβοι μπορούν να προστίθενται ή να αφαιρούνται στο σύστημα. Επιπλέον, η συγκεκριμένη μέθοδος προσαρμόστηκε και τροποποιήθηκε για την διατήρηση πολλαπλών αντιγράφων των αποθηκευμένων αντικειμένων, με στόχο την επίτευξη υψηλής διαθεσιμότητας των δεδομένων και την ανοχή του συστήματος σε βλάβες. Αυτή η προσέγγιση εξασφάλισε ότι το σύστημα παραμένει λειτουργικό και διαθέσιμο, ακόμα και σε περίπτωση αποτυχίας μεμονωμένων κόμβων, καθώς τα δεδομένα διατηρούνται σε πολλαπλά αντίγραφα σε διαφορετικούς κόμβους.

Η ανάπτυξη του συστήματος αποθήκευσης παρουσίασε αρκετές προκλήσεις και δυσκολίες στην υλοποίηση, οι οποίες σχετίζονταν κυρίως με την κατανεμημένη φύση του συστήματος και την ανάγκη διατήρησης μιας κοινής κατάστασης και συνέπειας μεταξύ των ξεχωριστών κόμβων. Μία από τις κυριότερες προκλήσεις ήταν η διασφάλιση ότι όλοι οι κόμβοι έχουν πρόσβαση σε κοινά δεδομένα που είναι συνεπή σε όλους τους κόμβους, όπως η λίστα των κόμβων του συστήματος και η λίστα με τα αποθηκευμένα αντικείμενα. Αυτό χρειάστηκε την υλοποίηση μηχανισμών που να εξασφαλίζουν ότι οι πληροφορίες αυτές παραμένουν ενημερωμένες και συνεπείς σε όλους τους κόμβους του συστήματος.

Επιπλέον, ένα σημαντικό ζητούμενο είναι η επίτευξη της ισοτιμίας στους κόμβους του συστήματος. Δηλαδή κάθε κόμβος πρέπει να είναι σε θέση να εκτελεί όλες τις απαραίτητες λειτουργίες και ενέργειες, όπως για παράδειγμα η προσθήκη ή αφαίρεση κόμβων και αντικειμένων, χωρίς να υπάρχει κάποιος κεντρικός κόμβος ή διαχειριστής που να ελέγχει τη λειτουργία του συστήματος. Αυτή η απαίτηση αύξησε την πολυπλοκότητα του συστήματος, καθώς κάθε κόμβος έπρεπε να είναι αυτόνομος και να μπορεί να συνεργάζεται με τους υπόλοιπους κόμβους του συστήματος και τη διατήρηση της συνέπειας.

Μία ακόμη σημαντική πρόκληση αποτέλεσε η ανακατανομή και μετακίνηση των αποθηκευμένων δεδομένων μεταξύ των κόμβων σε περίπτωση αλλαγής της σύνθεσης του συστήματος, όπως η προσθήκη ή η αφαίρεση κόμβων. Η δυσκολία αυτή προέκυψε λόγω της ανάγκης οι κόμβοι να λειτουργούν συνεργατικά και κατανεμημένα, ώστε ο καθένας να αναλαμβάνει την ανακατανομή ενός τμήματος των δεδομένων χωρίς να υπάρχουν επικαλύψεις ή συγκρούσεις μεταξύ τους. Με άλλα λόγια, ήταν απαραίτητη η ανάπτυξη ενός μηχανισμού όπου κάθε κόμβος είναι υπεύθυνος για την ανακατανομή ενός συγκεκριμένου τμήματος του συνόλου των αποθηκευμένων δεδομένων. Έτσι, επιτεύχθηκε η δυνατότητα της παράλληλης και κατανεμημένης εκτέλεση της ανακατανομής των δεδομένων από όλους τους κόμβους του συστήματος.

Αυτή η δυσκολία αυξήθηκε σημαντικά μετά την παραμετροποίηση του συστήματος για τη διατήρηση πολλαπλών αντιγράφων των αντικειμένων. Αυτό οφείλεται στο γεγονός ότι οι μετακινήσεις των αντικειμένων δεν γίνονταν μεταξύ δύο μόνο κόμβων, αλλά μεταξύ δύο συνόλων κόμβων, καθώς κάθε αντικείμενο διατηρείται σε πολλαπλά αντίγραφα στο σύστημα. Επιπλέον, η ανακατανομή των αντικειμένων έπρεπε να γίνεται με τρόπο που να ελαχιστοποιεί τον αριθμό των μεταφορών δεδομένων μεταξύ κόμβων, ώστε να διασφαλιστεί η γρήγορη και αποτελεσματική ολοκλήρωση της διαδικασίας. Αυτή η απαίτηση αύξησε την πολυπλοκότητα του συστήματος, καθώς έπρεπε να εξασφαλιστεί ότι η μετακίνηση των δεδομένων γίνεται με τον πιο βέλτιστο τρόπο, χωρίς να δημιουργούνται περιττές μεταφορές ή συγκρούσεις μεταξύ των κόμβων για τις μετακινήσεις των αντικειμένων.

Ωστόσο, η χρήση της μεθόδου του δακτυλίου κατακερματισμού διευκόλυνε σημαντικά την αντιμετώπιση αυτών των δυσκολιών. Κάθε κόμβος μπορούσε εύκολα να επιλέξει τα δεδομένα για τα οποία είναι υπεύθυνος και να υπολογίσει πού χρειάζεται να μετακινηθούν. Αυτό είναι εφικτό χάρη στη δομή του δακτυλίου κατακερματισμού όπου προσδιορίζεται με ακρίβεια ο υπεύθυνος κόμβος κάθε αποθηκευμένου αντικειμένου. Επιπλέον, οι ξεχωριστοί κόμβοι έχουν ξεχωριστά σύνολα αντικειμένων, καθώς είναι υπεύθυνοι για διαφορετικά τμήματα του δακτυλίου. Αυτό το γεγονός απέτρεψε συγκρούσεις μεταξύ των κόμβων, καθώς δεν υπάρχουν επικαλύψεις στα σύνολα αντικειμένων για τα οποία είναι υπεύθυνοι ξεχωριστοί κόμβοι. Με αυτόν τον τρόπο, η διαδικασία της ανακατανομής έγινε πιο απλή, αποτελεσματική και χωρίς να παρουσιάζονται συγκρούσεις μεταξύ ξεχωριστών κόμβων και επιτρέποντας την κατανεμημένη εκτέλεση της λειτουργίας ανακατανομής.

Σε ζητήματα που αφορούν μελλοντικές βελτιώσεις και προσθήκες, υπάρχουν αρκετές πιθανές βελτιώσεις και επεκτάσεις που θα μπορούσαν να ενισχύσουν περαιτέρω την απόδοση και την αξιοπιστία του συστήματος αποθήκευσης. Μία από τις πιο σημαντικές βελτιώσεις θα ήταν η χρήση εικονικών κόμβων (Virtual Nodes), οι οποίοι θα βελτίωναν την ομοιόμορφη κατανομή των δεδομένων και θα μείωναν την πιθανότητα δημιουργίας περιοχών υψηλής συγκέντρωσης δεδομένων που επιτυγχάνει πιο καλή κατανομή των δεδομένων.

Συνολικά, το σύστημα που αναπτύχθηκε για την εργασία αποτελεί μια αποτελεσματική λύση για την κατανομή και αποθήκευση δεδομένων σε κατανεμημένα περιβάλλοντα. Η επιλογή του συνεπή κατακερματισμού και η υποστήριξη πολλαπλών αντιγράφων καθιστούν το σύστημα αποδοτικό, αξιόπιστο, και με δυνατότητα κλιμάκωσης και προσαρμογής του σε δυναμικά περιβάλλοντα. Για την επίτευξη καλύτερης κατανομής δεδομένων σε πολλαπλούς κόμβους η χρήση της μεθόδου των εικονικών κόμβων (Virtual Nodes) θα μπορούσε να αποτελέσει μια σημαντική βελτίωση, βελτιώνοντας περαιτέρω την ισορροπία φόρτου και την αποδοτικότητα του συστήματος αποθήκευσης.

- Consistent Hash Algorithm in Distributed Monitoring System. (2020). *Distributed Processing System*, 1(2). <https://doi.org/10.38007/DPS.2020.010204>
- Consistent Hashing Explained*. (2022, December 18). System Design. <https://systemdesign.one/consistent-hashing-explained/>
- hashlib—Secure hashes and message digests*. (n.d.). Python Documentation. Retrieved February 26, 2025, from <https://docs.python.org/3/library/hashlib.html>
- Hwang, S.-Y., Lee, K., & Chin, Y. (2006). *Data replication in a distributed system: A performance study* (Vol. 1134, pp. 708–717). <https://doi.org/10.1007/BFb0034724>
- Key–value database. (2024). In *Wikipedia*. https://en.wikipedia.org/w/index.php?title=Key%E2%80%93value_database&oldid=1258737249
- Messaoud, A. A., Mokhtar, S. B., & Simonet-Boulogne, A. (2025). TEE-based Key-Value Stores: A Survey. *The VLDB Journal*, 34(1), 10. <https://doi.org/10.1007/s00778-024-00877-6>
- Roughgarden, T., & Valiant, G. (n.d.). *CS168: The Modern Algorithmic Toolbox Lecture #1: Introduction and Consistent Hashing*.
- Round-robin item allocation. (2024). In *Wikipedia*. https://en.wikipedia.org/w/index.php?title=Round-robin_item_allocation&oldid=1239142204
- Tanenbaum, A. S., & Steen, M. van. (2007). *Distributed systems: Principles and paradigms* (Second edition). Pearson, Prentice Hall.

9780134032801.pdf. (n.d.). Retrieved February 26, 2025, from

https://ptgmedia.pearsoncmg.com/images/9780134032801/samplepages/9780134032801.pdf?fbclid=IwZXh0bgNhZW0CMTEAAR1mZSx8dpqnfytnJCnhQJNiLd8E-QQhmeE_so3uyLAOqdonnAlZ03LgDw4_aem_sCK8nGnzZHF00UUqSSg6CA

Welcome to Flask—Flask Documentation (3.1.x). (n.d.). Retrieved February 26, 2025, from

https://flask.palletsprojects.com/en/stable/?fbclid=IwZXh0bgNhZW0CMTEAAR3-mSLBkAVhxmoyAkdvQDJdMpaDvFPa0IP2N88GDNnbMUQp3j73IG_kHLY_aem_Msrs44LOERUzUfEJbiH_VA

Welcome to Python.org. (2025, February 18). Python.Org. <https://www.python.org/>

Yuan, D., Peng, X., Liu, T., & Cui, Z. (2017). *A Randomly Expandable Method for Data Layout of RAID Storage Systems* (arXiv:1709.04708). arXiv. <https://doi.org/10.48550/arXiv.1709.04708>

Συνάρτηση κατακερματισμού. (2024). In *Βικιπαίδεια*.

https://el.wikipedia.org/w/index.php?title=%CE%A3%CF%85%CE%BD%CE%AC%CF%81%CF%84%CE%B7%CF%83%CE%B7_%CE%BA%CE%B1%CF%84%CE%B1%CE%BA%CE%B5%CF%81%CE%BC%CE%B1%CF%84%CE%B9%CF%83%CE%BC%CE%BF%CF%8D&oldid=10408018