COST-AWARE BIG DATA PROCESSING ACROSS GEO-DISTRIBUTED DATACENTERS

A PROJECT REPORT

Submitted by

SUKRITI TIWARI [Reg No: RA1511008010407]

ANIRUDH JAISWAL [Reg No: RA1511008010439]

YASH JAIN [Reg No: RA1511008010463]

Under the Guidance of

Ms. G. Geetha

(Assistant Professor, Department of Information Technology)

In partial fulfillment of the Requirements for the Degree

of

BACHELOR OF TECHNOLOGY
IN
INFORMATION TECHNOLOGY

SRM
INSTITUTE OF SCIENCE & TECHNOLOGY
Deemed to be University u/s 3 of UGC Act, 1956

DEPARTMENT OF INFORMATION TECHNOLOGY

FACULTY OF ENGINEERING AND TECHNOLOGY

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

MAY 2019

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY
KATTANKULATHUR-603203

BONAFIDE CERTIFICATE

Certified that this project report titled **"COST-AWARE BIG DATA PROCESSING ACROSS GEO-DISTRIBUTED DATACENTERS"** is the bonafide work of **"SUKRITI TIWARI [RA1511008010407], ANIRUDH JAISWAL [RA1511008010439], and YASH JAIN [RA1511008010463],** who carried out the project work under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or award was conferred on an earlier occasion for this or any other candidate.

Ms. G. GEETHA	Dr. G. VADIVU
GUIDE	**HEAD OF THE DEPARTMENT**
Assistant Professor	Dept. of Information Technology
Dept. of Information Technology	

Signature of Internal Examiner

Signature of External Examiner

ACKNOWLEDGEMENT

The achievement and the ultimate result of our venture required direction and help from various sources and we feel extremely lucky to have this up and down the finish of our task. Whatever we have done is to a great extent because of such direction and help and we would not neglect to express gratitude toward them.We express our sincere thanks to the Head of the Department, Department of Information Technology, Dr.G.Vadivu, for all the help and infrastructure provided to us to complete this project successfully and her valuable guidance.

We owe our significant appreciation to our undertaking guide Ms.G.Geetha, who took a distinct fascination for our task work and guided every one of us along, till the culmination of our venture work by giving all the essential data to building up a decent framework. We are grateful for and sufficiently lucky to get consistent consolation, backing and direction from all the Teaching staff of the Department of Information Technology which helped us in effectively finishing our significant undertaking work. Additionally, we might want to stretch out our earnest respects to all the non-showing staff of the Department of Information Technology for their convenient help.

SUKRITI TIWARI (RA1511008010407)

ANIRUDH JAISWAL (RA1511008010439)

YASH JAIN (RA1511008010463)

ABSTRACT

With administration globalization, associations are continually creating vast volumes of information that should be broken down crosswise over geo-scattered areas. Customarily focal methodology is that exchanging all information to a solitary group is wasteful or ineffectual because of confinements, for example, wide - zone transmission capacity shortage and information handling low idleness necessities. Colossal data planning transversely over geo-scattered server cultivates starting late continues grabbing acclaim.

Supervising MapReduce passed on figuring across over geo-scattered server ranches, regardless, indicates different particular troubles: How to administer information to geo-appropriated server farm choices to decrease correspondence costs? How to decide the provisioning system for VM (Virtual Machine) offering superior and ease?What's more, what criteria should be used to pick a server ranch as the last enormous data examination work reducer? These difficulties are tended to through adjusting the expense of transmission capacity, cost of capacity, cost of registering, cost of relocation, and cost of idleness between the two periods of MapReduce crosswise over datacenters.

For data improvement, resource provisioning and reducer decision, we figure this staggering cost headway issue into a joint stochastic entire number nonlinear streamlining issue at the same time limiting the five cost factors. The MapReduce and clustering structure is coordinated into our examination and the further plan is an effective online calculation equipped for limiting the long - term normal working expense. Hypothetical examination demonstrates that our online calculation can furnish a practically ideal arrangement with a demonstrated hole and can guarantee that information preparing can be finished inside predefined time limits.

TABLE OF CONTENTS

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS AND ABBREVIATIONS

D: set of datacenters (DC)

R: set of data locations

K: set of VM types

$a_r(t)$: amount of the data generated from data source (DS) r at time t

HadUP: Hadoop with Update Processing

DS: Data Service Provider

RFID: Radio Frequency Identification

CHAPTER 1

INTRODUCTION

1.1 GENERAL

Be that as it may, because of properties, for example, extensive volume, high multifaceted nature and dispersive nature of Big Data combined with Wide zone transmission capacity shortage (e.g. trans-oceanic link); the processing of data with centralized solutions is inefficient and/or impossible.

This has driven strong industry companies to deploy cloud and hybrid cloud multi-data center. These cloud advancements give an amazing and savvy answer for location the undeniably rapid of huge information produced from geo-conveyed sources. For most common organizations (e.g., SKA), renting resources from the public cloud is cost-effective, taking into account the recompenses of cloud computing such as tractability and business model pay-as-you-go.

1.2 PURPOSE

We're trying to come up with a centralized solution by introducing more parameters in order to flexibly change the way big data is handled. By including an increased number of parameters, we can have greater control of the decisions taken while steering data according to usage metrics. Various scenarios are incorporated in order to obtain analytical graphs that help determine changes in future decision making practices.

In terms of cost optimality and worst-case delay, we formally analyze our algorithm's performance. We demonstrate that the algorithm estimates the optimal solution within verifiedfrontiers and ensures that processing can be concluded within predefined delays.We lead broad tests with genuine world datasets to assess the execution of our online calculation.

1.3 SCOPE

Data processing is a widely used feedback technique for most companies these days and our algorithm is mainly focused on helping them achieve optimal results. Cost escalations can occur at every point of the data analysis process and we aim to reduce those through further analyzing decisions.

MapReduce is a distributed indoctrination model for parallel large-scale data set processing that has demonstrated its exceptional efficiency in many existing applications. Because the original MapReduce model is not optimized for data center deployment, a widely used approach is to aggregate distributed data to a single data center for centralized processing.Be that as it may, due to the heterogeneous and restricted client cloud interface data transfer capacity. We're sitting tight for such unified accumulation experiences critical deferrals.

Note that the transmission capacity of the between datacenter connect is typically committed to generally high transfer speed lines, moving the information to numerous parallel guide activity datacenters and afterward collecting the halfway information to a solitary server farm to decrease task utilizing between datacenter connection may lessen dormancy.

Also, unique sorts of expenses (for example moving information or leasing VM) can likewise be advanced thinking about the heterogeneity of connection speed, information age dynamism and asset cost. In this way, appropriating and preparing multi-source information into multi-server farms utilizing circulated MapReduce is a thoughtful approach to manage scattered vast volume information.Up to this point, the most imperative inquiries to be settled include:

1) How to upgrade the position of extensive scale datasets for handling from various areas onto the geo-appropriated server farm cloud, and

2) How many resources, such as computer resources, should be provided to ensure performance and availability while minimizing costs? The fluctuation and multiple generated data sources in combination with the cloud resource's dynamic utility - driven pricing model make it a very challenging issue.

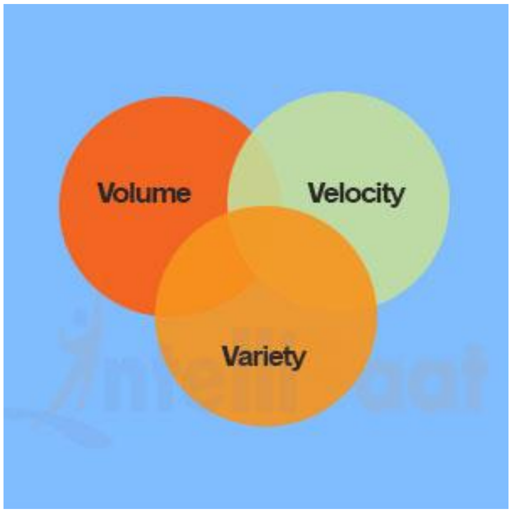
1.4 WHAT IS BIG DATA?

Big Data is a characterized term for substantial or complex informational indexes that are lacking for customary information handling applications. Big Data basically consists of zing analysis, data capture, data creation, search, sharing, storage, transfer, visualization, and data privacy query. Big Data is a gathering of expansive informational collections which can't be prepared sufficiently utilizing conventional handling strategies. Big Data isn't simply information that has turned into a total subject, including diverse instruments, procedures and structures.

Big Data term depicts the proportion of data in the ordinary business condition, both sorted out and unstructured.It is important that the data that matters is used by what organizations with these. Big data helps analyze in-depth concepts for better organizational decision-making and strategic development.

- In request to benefit from Big Data, it is important to require a framework that oversees and forms immense volumes of organized and unstructured information continuously and can ensure information protection and security. There are numerous advances accessible available from different sellers to approach Big Data, including Amazon, IBM, Microsoft, and so forth.

While in this period the expression "big data" is new as it is the demonstration of the social affair and putting away immense measures of data for inevitable investigation is old. The idea appeared in the mid-2000s when Doug Laney, a modern expert, characterized enormous information as the accompanying three classes:



Big data works on data produced and applied by different devices. Below are some of the fields involved in Big Data's umbrella.

Black Box Data: Black box Data: It is built-in flight craft that stores a lot of data, including a discussion between team individuals and some other correspondences (alert messages or any order passed) by the technical grounds duty staff.

Power Grid Data: Power lattice information chiefly contains the base station data devoured by a particular hub.

What is Big Data's importance?

Big data's importance is how you use the data you own. Data can be collected from any source and analyzed to solve that enables us to make smart decisions in terms of

1) cost reductions

2) time reductions,

3) new product development and optimized offers, and

4) Smart decision making

Combining big data with high-powered analytics, you can have a big impact on your business strategy for example:

- Finding the main driver of disappointments, issues and weaknesses continuously activities.
- Given the client's propensity for purchasing products, creating coupons at the purpose of offer.
- Complete hazard portfolios are determined in minutes.
- Detect fraudulent behaviors before they affect your organization and risk them.

1.5 NEED OF BIG DATA ANALYSIS

The" big data" phenomenon is now present in every sector and function of the global economy. Contemporary shared settings are frequently connected with huge, consistently expanding quantities of different information types that shift regarding significance, subjectivity and significance. Knowledge extracted can range from individual opinions to widely accepted practices. The present organizations face difficulties in information the board as well as in breaking down huge information, requiring new ways to deal with additional bits of knowledge from an exceedingly itemized, contextualized and rich substance. Collaborative sensory making occurs very often in such settings, orchestrated or otherwise, before actions or decisions are made.

However, our understanding of how these tools can interact with users to foster and exploit a synergy between the intelligence of humans and machines is quite often behind the technologies. Often the term " data analytics" is used to cover any decision-making driven by data. Real interest in big data, legitimately focused on, can result in major logical advances as well as establish the framework for the up and coming age of advances in science, drug and business.To help basic leadership, data analysts pick enlightening measurements that can be figured with the important calculations or devices from accessible information and report the outcomes in a way that the chiefs can fathom and follow upon.

Big Data analytics is a work process refining terabytes of low-esteem information (e.g., each tweet) down to a solitary piece of high-esteem information sometimes (e.g., should Company X secure Company Y?). To construct frameworks and propelled calculations or administrations for big data examination, advancements, for example, information mining, AI and the semantic web are being abused. Most administrations and calculations are built in an innovation-driven way with little client contribution to drive arrangements advancement.

1.6 PROBLEM STATEMENT

The most frequently asked questions include:

1) How to transfer large-scale data sets from various locations into the geo distributed datacenter cloud and

2) How many resources such as computing resources should be provisioned in such a geo distributed cloud to guarantee performance and availability while minimizing the cost.

The heterogeneity and multiple sources of generated data combined with the dynamic utility driven resource pricing model make it a very challenging problem.

The inter-dependency between multiple stages of distributed computation, such as the interplay between the Map phase and the Reduce phase of Map Reduce programs, further escalates the complexity of the data movement; resource provisioning and final reduce selection problems in geo-distributed datacenters.

1.7 RESEARCH OBJECTIVE

Heretofore, the most essential inquiries to be settled include:

1) How to enhance the position of extensive scale datasets for preparing from various areas onto the geo-distributed data center cloud, and

2) How many resources, such as computer resources, should be provided to ensure performance and availability while minimizing costs? The fluctuation and multiple generated data sources in combination with the cloud resource's dynamic utility-driven pricing model make it a very challenging issue.

3)Comprehension and Targeting Customers: Big information enables an association to comprehend its clients better, and causes it restricted down the intended interest group, in this way improving its promoting effort.

4) Taking Strategic Decisions: With big data, businesses can take data-backed decisions. No compelling reason to mishandle in obscurity, when the tremendous volume of information gives for all intents and purposes all of the data about your business, advertise, clients, industry, and rivalry.

5) Cost Optimization: Costs can be better enhanced when you have the information to know which components are depleting costs yet not returning high esteem. For example, the human services part can use huge information to discover the reason for the cost climb of social insurance offices.

6) Improving Customer Experiences: Take the case of retail. Big data can help retailers & wholesalers show customer reviews about the product’s quality & delivery time, thus improving customer experiences in the retail buying process.

7) We address these difficulties by adjusting transfer speed cost, stockpiling cost, registering cost, relocation cost, and dormancy cost, between the two Map Reduce stages crosswise over datacenters.

8) We detail this convoluted cost streamlining issue for information development, asset provisioning and reducer determination into a joint stochastic whole number nonlinear enhancement issue by limiting the three cost factors at the same time.

9) We coordinate the MapReduce and Clustering system into our investigation and plan a proficient online calculation that can limit the long haul time-found the middle value of activity cost.

CHAPTER 2

LITERATURE SURVEY

2.1 DATA MINING WITH BIG DATA

[1] It doesn't take long for data mining application results to become outdated over time as new data and updates are regularly arriving. Incremental processing is a lucrative way of refreshing mining results. It uses states that were previously saved to avoid scratching re - computing costs. We propose i2MapReduce in this paper, a new incremental processing extension for MapReduce.As compared to the high developing state of Incoop's work, i2MapReduce conducts key-value pair - level gradual handling rather than task level re - computing. It not only upholds one-step computing but also far more technologically advanced iterative computing and also integrates a set of new techniques to reduce overhead I / O access to preserved fine-grain computing states.Research results on Amazon EC2 indicate compelling refinements in the performance of i2MapReduce re - calculation contrasted to MapReduce both simple and iterative.

[2] An important characteristic of today's big data acquisition is that the same calculation is often replicated over time on sets of data like web and social network data. While it is viable to repeat complete computation of the enormous datasets with parallel programming frameworks like Hadoop, it is obviously unproductive and drains resources. HadUP, a changed Hadoop engineering customized to established MapReduce calculations for huge - scale gradual handling, is exhibited here. Several strategies were proposed which use task-level memoization to produce a similar goal. Task - level memoization, even so, detects the change in databases at a crude stage, which often affects such approaches. Instead, HadUP uses deduplication - based screenshot differential algorithm (D - SD) to locate and calculate data set changes at a fine-grained level and patch them.

[3] Big Data has become a hot spot in contemporary times as the advent of the internet, cloud services, cellular network and the Internet of Things is growing rapidly. Big Data Extraction is embroiled in our routine existence, such as portable devices, RFID and remote sensor systems, with the goal of handling the immersive trillion user data worldwide. Simultaneously, in an integrated system, real-time processing is keenly needed. Several technologies are initiated in this document associated with real-time big data processing, including the underlying technology called Storm. A complete Storm-based system is designed in conjunction with RabbitMQ, NoSQL and JSP. A simulation system is created and shows tolerable performance in different attributes using spec sheets and Ganglia to ensure practical pertinence and high efficiency. It is demonstrated that Storm-based large data real-time sequencing can be widely used in different computing environments.

2.2 LARGE-SCALE INCREMENTAL PROCESSING

[4] Big Data is about large, nuanced, growing data sets with several, and autonomous sources. With its massive expansion, Big Data is gaining popularity in all branches of science and engineering, including physiological, biological and biomedical sciences, with the rapid enlargement of networking, data inventory and capture capacity. This paper poses a HACE data mining theorem that embodies the characteristics of the Big Data Revolution and recommends the monitoring of Big Data model. This data-driven model entails aggregating information sources in a demand-driven manner, mining and analysis, modelling user interest, and aspects of security and confidentiality. We are analyzing the challenging issues in the data-driven paradigm and also in the Big Data revolution.

[5] Cloud computing allows groups of academic subversives, business associate groups, etc. to meet in an ad - hoc sense. This paper focuses on the group-based data transfer problem in such

settings. Each contributor source site within such a unit has a vast dataset that can range from gigabytes to terabytes in size. This data must be transmitted to one single sink site (like AWS, Google datacenters etc.) to minimize both the group's total dollar expenses and the total offload latency of the collective data set. The first paper to examine the problem of organizing a group-based, deadline-oriented data transmission, this technology stores in a lot of potential, especially in a circumstance where data can be sent via both: (1) the internet and (2) shipping solid state drives (e.g. external or hot-plug drives or SSDs) via companies such as FedEx, UPS, USPS etc. The problem is first formalized by proving its NP-Hardness. Crisp calculations are then proposed and used to fabricate a booking framework called Pandora (People and Networks Moving Data Around). Pandora utilizes new ideas of time-extended systems and delta-time-extended systems, joining them with number programming procedures and advancements for both delivery and web edges.Our test assessment utilizing genuine information from FedEx and from PlanetLab demonstrates the Pandora organizer figures out how to fulfill due dates and diminish costs altogether.

[6] The developing force and cooling necessities of high-thickness processing frameworks present noteworthy difficulties for the structure and activity of PCs and their offices. The rising working costs for datacenters request the execution of vitality effective advancements and the best power the board arrangements. This instructional exercise tends to control the board and cooling arrangements from the individual PC framework level to the datacenter. The group of onlookers will find out about the crucial idea of the issues, ways to deal with creating arrangements, accessible business arrangements, and momentum look into bearings.

2.3 CLUSTERING TECHNIQUES

[7] Map Reduce has demonstrated surprisingly compelling for a wide assortment of information serious applications, yet it was intended to keep running on huge single-site homogeneous

bunches. Analysts have started to investigate the degree to which the first Map Reduce suspicions can be loose, including skewed outstanding tasks at hand, iterative applications, and heterogeneous registering situations. This paper proceeds with this investigation by applying Map Reduce crosswise over geo-disseminated information over geo-circulated calculation assets. Utilizing Hadoop, we demonstrate that system and hub heterogeneity and the absence of information region lead to poor execution in light of the fact that the cooperation of Map Reduce stages ends up articulated within the sight of heterogeneous system conduct. To address these issues, we adopt a two-dimensional strategy: We initially build up a model-driven streamlining that fills in as a prophet, giving abnormal state experiences. We at that point apply these bits of knowledge to configuration cross-stage advancement strategies that we execute and exhibit in a true Map Reduce usage.

[8] Productively breaking down Big Data is a noteworthy issue in our present period. Instances of investigation assignments incorporate distinguishing proof or discovery of worldwide climate designs, financial changes, social wonders, or plagues. The distributed computing worldview alongside programming instruments, for example, usage of the well-known MapReduce structure offers a reaction to the issue by conveying calculations among huge arrangements of hubs. In numerous situations, input information is, in any case, geologically conveyed (geo appropriated) crosswise over server farms, and clearly moving all information to a solitary server farm before handling it tends to be restrictively costly. Previously mentioned apparatuses are intended to work inside a solitary group or server farm and perform ineffectively or not in the slightest degree when sent crosswise over server farms. This paper manages executing groupings of MapReduce occupations on geo-circulated informational collections.

[9] Cloud computing, quickly rising as another calculation worldview, gives nimble and versatile asset access in an utility-like style, particularly for the handling of enormous information. A significant open issue here is to productively move the information, from various geological

areas after some time, into a cloud for compelling handling. The accepted methodology of hard drive shipping isn't adaptable or secure. This work examines the opportune, cost-limiting transfer of gigantic, powerfully produced, geo-scattered information into the cloud, for preparing to utilize a MapReduce-like structure.

[10] Focusing at a cloud including dissimilar server farms, we model an expense limiting information movement issue, and propose two online calculations: an online languid relocation (OLM) calculation and a randomized fixed skyline control (RFHC) calculation, for improving at some random time the decision of the server farm for information collection and preparing, just as the courses for transmitting information there. Cautious correlations among these on the web and disconnected calculations in sensible settings are led through broad analyses, which exhibit the near disconnected ideal execution of the online calculations

CHAPTER 3

PROPOSED METHODOLOGY

With regards to MapReduce running over numerous server farms and VMs of different sorts and dynamic costs, we propose a structure that can manage the issues of data advancement, resource provisioning and reducer assurance.We figure the intricate cost enhancement issue as a joint stochastic whole number nonlinear streamlining issue and comprehend it utilizing MapReduce and Clustering advancement structure by changing the first issue into three free subproblems (information development, asset provisioning and choice decrease) that can be illuminated with some straightforward arrangements. We are planning a proficient and disseminated online. Our algorithm calculation that can limit long - term, time-arrived at the midpoint of activity costs. We are planning an effective and conveyed online. Our algorithm calculation that can limit long - term, time-found the middle value of activity costs. Exploratory outcomes exhibit its cost-viability, framework dependability, and choice - making — making prevalence over existing agent methodologies, for example, the mix of information allotment systems (vicinity - mindful, load - balance - mindful) and asset arrangement procedures (e.g., stable methodology, heuristic technique).

CHAPTER 4

MODULE DESCRIPTION

THE EXECUTIVES OF NUMEROUS DATACENTERS

Dealing with different conveyed server farms has pulled in organizations like Facebook, Google, HP, and Cisco. Facebook built up a Prism venture to help geo - disseminated Hadoop information stockpiling by adding a sensible deliberation layer to the Hadoop bunch. Concentrating on adaptation to internal failure and burden adjusting, Google has circulated its Spanner database framework, which enables information to be naturally moved crosswise over datacenters. Additionally, HP and Cisco attempted endeavors to deal with their geo-disseminated server farms by advancing the system of entombing - datacenters on the information interface layer. Notwithstanding, their transportation reliance, intricacy and absence of versatility limit current down to earth strategies. Moreover, these strategies centre essentially around giving a superior nature of administration for progressively worldwide client requests, however not information figuring.

COST

The DSP will likely limit the general expense of the framework by streamlining the measure of information assigned to every datum focus, the number of assets required, and the reasonable server farm for task decrease. Specifically, this paper thinks about the accompanying cost parts: cost of data transmission, cost of capacity, cost of idleness, cost of registering and cost of relocation. The transmission capacity cost is normally shifted crosswise over various VPN joins since they frequently have a place with various suppliers of web administrations.

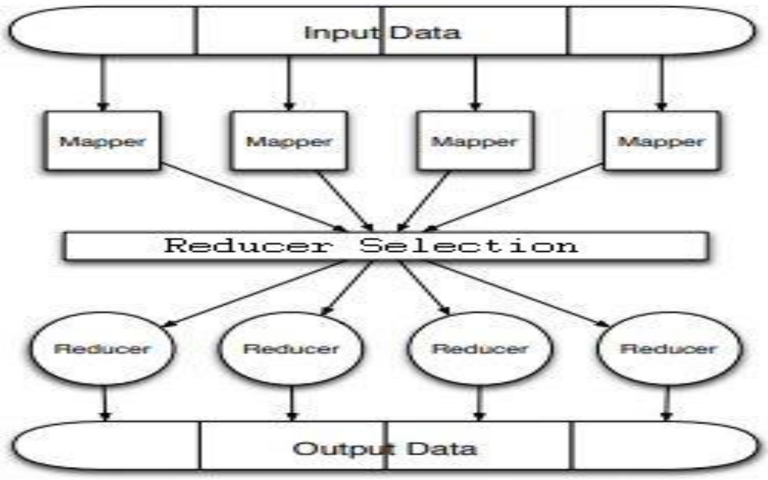
ONLINE ALGORITHM DESIGN

An outstanding component of enhancing MapReduce and Clustering is that it needn't bother with the future remaining task at hand data. By avariciously limiting the float in addition to punishment at each space-time, it can successfully take care of the issue of long haul streamlining with an answer that can turn out to be discretionarily near the ideal. Next, we initially change the P1 issue into an issue of limiting the MapReduce and Clustering float in addition to punishment term and after that plan the comparing calculation on the web.

INFORMATION ALLOCATION

Cautious perception of the connection between different factors on the R.H.S uncovers the assignment of information. Moreover, since the information produced at every datum source is autonomous; the incorporated enhancement can be freely executed and appropriated at every datum source.

The three complex issues of information assignment, asset provisioning, and opening t reducer determination have been tackled autonomously and proficiently up until this point. The basic methodologies encourage true calculation sending on the web. Utilizing the line refreshing technique alongside schedule openings, we can plan our algorithm online calculation to take care of the long haul issues.



LINEAR, LINEARBIG

This module is implemented when either the sourceor destination of two or more data transmissions are same. Our idea is to promote bulk sharing from various locations. In order to optimally minimize costs pertaining to this, various factors such as the type of data being sent, the bandwidth speed, the distance between the source and destination and many more have to be taken into consideration. Depending on the size of the area, we've modified code accordingly.

JOIN, JOINBIG

As long as the distance to be covered is the same, no matter what source or destination, this algorithm is used. The main aim of all these modules is to incorporate greater number of scenarios into the situation. The latitude and longitude key value pairs are fed in and after keying in the other parameters, the resultant graph comes out.

OPTIMAL

The processes are stored in a queue as they arrive. In the first cycle, 50% of the data is fed in. After that is handled and an intermediate graph is obtained, a quarter of the remaining data is processed. Finally the remaining 25% is treated and all results are averaged. This is the closest model to real time usage and has the potential to make it big in the industry.

CUMULATIVE

All data is sent to the system at once in this approach and the mean value is obtained. This model is prone to error as outliers tend to disrupt. However this remains the most commonly used technique. Trends do show a decrease in costs over time as evident from the output graphs attached. The slope is, however, not very steep.

QUEUE

Each clock cycle witnesses one half of data being fed into the system and the other half of processing and graph rendering. The two results are averaged in the end. The error percentage in this model is less than cumulative but more than optimal. In real time humongous quantities of data are uploaded at insane speeds onto the server and ideally these analysis graphs are to change every second. However, ours is a prototype of the original system and hence has static graphs.

CHAPTER 5

SYSTEM ANALYSIS

HARDWARE REQUIREMENTS:

- System : Pentium Dual Core (Minimum).
- Hard Disk : 256 GB (Minimum).
- Monitor : 15’’ LED
- Input Devices : Keyboard, Mouse
- Ram : 1GB (Minimum)

SOFTWARE REQUIREMENTS:

- Operating system: Windows 7/UBUNTU.
- Coding Language : Java 1.7,Hadoop 0.8.1
- IDE : Eclipse
- Database : MYSQL

5.1 EXISTING SYSTEM

• They proposed an asymptotic planning component for some, registering errands for enormous information handling stages. The basic component of these works is thinking about a static situation where the information is pre-put away in the cloud and the measure of information is fixed.

•Zhang et al. proposed an online calculation to relocate progressively produced information from different areas to the mists and concentrated how to limit the transmission capacity cost of exchanging information for deferral tolerant preparing with numerous Internet Service Providers (ISPs).

•Zhang et al. contemplated how to proficiently plan and perform investigation over information that is geologically dispersed over different datacenters and structured framework level improvements including work confinement, information arrangement and information pre-bringing for improving the execution of Hadoop administration provisioning in a geo-conveyed cloud.

•As of late, MapReduce and Clustering streamlining system was connected to a distributed computing setting to manage work confirmation and asset portion issue.

•Yao et al. expands it from the single time scale to two-time-scale for accomplishing power cost decrease in topographically conveyed datacenters.

5.2 ISSUES IN EXISTING SYSTEM

A distributed model of programming, MapReduce is for parallel processing of large-scale data sets, which has demonstrated its exceptional efficiency in many existing applications. Since the MapReduce algorithm has not been optimized for data center deployment, a widely used approach is to aggregate this distributed data, spread over geographically varied terrain, to a single data center for centralized processing. However, this gives rise to a number of other problems, the heterogeneous and limited bandwidth of the user cloud link to name a few. Waiting for or even switching to such centralized aggregation of data makes the process suffer from significant delays. Apart from the physical hardware support, software upgrades also need to be made. Note that the inter-datacenter link is usually dedicated to relatively high bandwidth lines, obviously to allow high speed data transfers. Moving the data in parallel to multiple mapped operation datacenters and subsequently aggregating the intermediately obtained data to one single data center can reduce latency.

- It is wasteful as well as infeasible to process the information with incorporated arrangements.
- Since unique Map Reduce display isn't improved for arrangement crosswise over server farms, totaling appropriated information to a solitary server farm for concentrated handling is a generally utilized methodology. Be that as it may, sitting tight for such brought together total experiences altogether delays due to the heterogeneous and restricted data transfer capacity of client cloud connect.
- Moving the information to various datacenters for guide activity in parallel and after that totaling the halfway information to a solitary server farm to decrease task utilizing the between datacenter interface can possibly lessen the inactivity.

5.3 FEASIBILITY ANALYSIS

In this phase, the project's feasibility is analyzed and business proposal is presented with a very general project plan and some estimates of costs.The achievability investigation of the proposed framework will be done amid the framework examination. This is to ensure that there is no burden on the company with the proposed system. Some comprehension of the real framework prerequisites is fundamental for practicality investigation.

Three key considerations involved in the feasibility analysis are

5.3.1 ECONOMICAL FEASIBILITY

This study is conducted to check the organization's economic impact of the system. There is a restricted measure of reserve that the organization can fill the framework's innovative work. The costs must be legitimized. As a result, the developed system was also achieved within the budget, as most of the technologies used are available freely. Only the custom products had to be bought**.**

5.3.2 TECHNICAL FEASIBILITY

This investigation is led to check the specialized attainability, for example, the framework's specialized prerequisites. Any created framework must not have an intense interest in the specialized assets accessible. This will result in levels of popularity on the specialized assets usable.The framework created must have a humble prerequisite, as the usage of this framework requires just insignificant or invalid changes.

5.3.3 SOCIAL FEASIBILITY

The study aspect is to check the user's acceptance level of the system. This includes the user training process for efficient use of the system. The system must not threaten the user, but must accept it as a necessity. The level of user acceptance depends solely on the methods used to educate and familiarize the user with the system. His confidence level must be raised so that he can also make some constructive criticism that is welcomed as he is the system's ultimate user.

CHAPTER 6

SYSTEM DESIGN&ARCHITECTURE

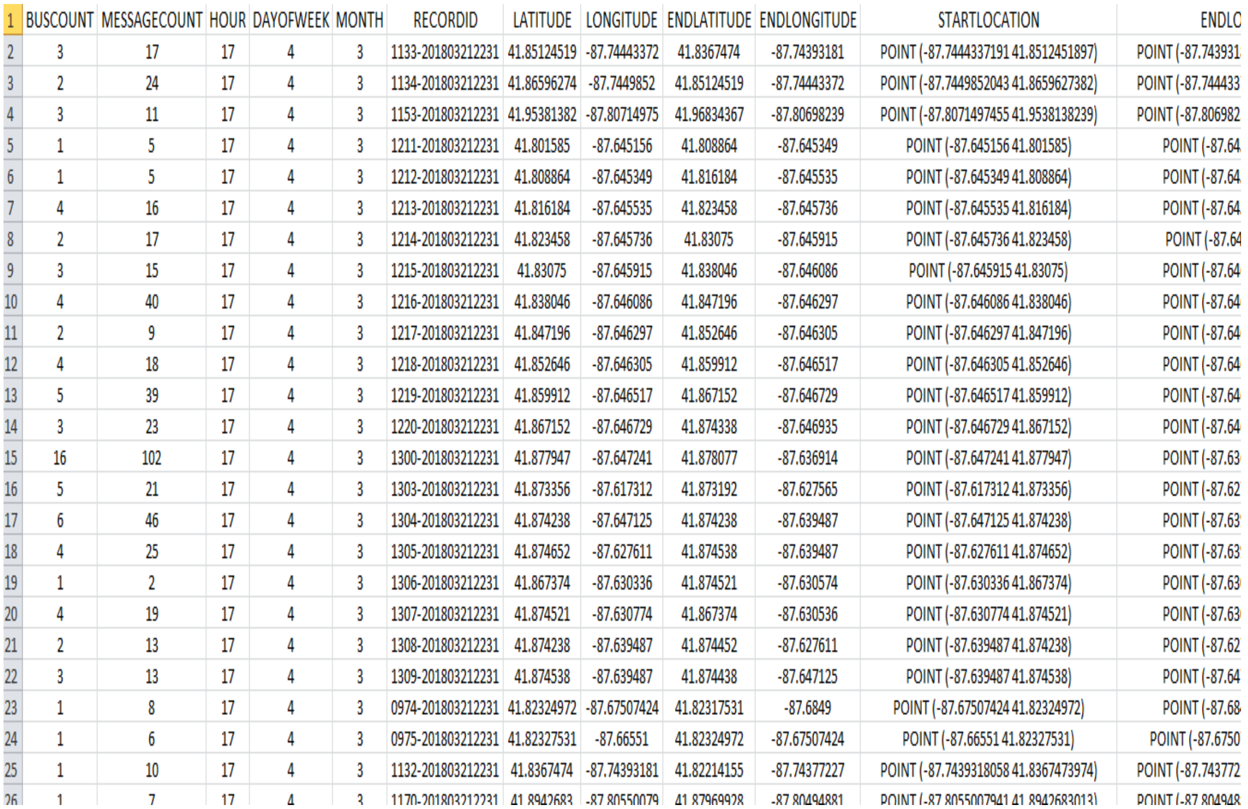
6.1 INPUT DESIGN

The info configuration is the connection between the data framework and the client. It includes the creating determination and methodology for information arrangement and those means are important to put exchange information into a usable structure for planning can be practised by analyzing the PC to scrutinize data from a made or printed file or it can occur by having people entering the information straightforwardly into the framework. The plan of info centres around controlling the measure of information required, controlling the mistakes, maintaining a strategic distance from postponement, evading extra methods and keeping the technique fundamental. The information is planned in such a way in this way, that it furnishes security and convenience with holding protection. Info Design considered the going with things:

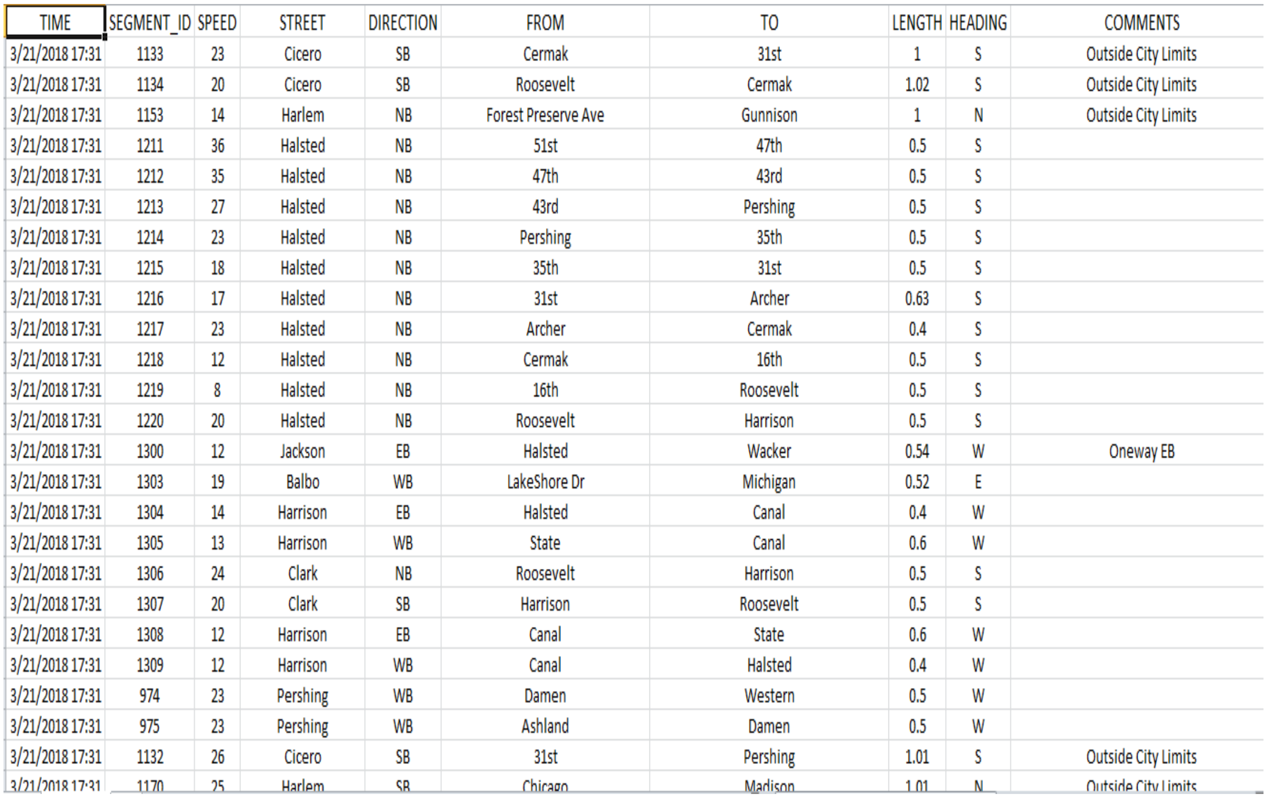
1) What data should be given as Input?

2) How the data should be organized or coded?The discourse needs to be decided to direct the working staff in giving information.

3) Methods for preparing input validations and steps to follow when error occurs.



1	BUSCOUNT	MESSAGECOUNT	HOUR	DAYOFWEEK	MONTH	RECORDID	LATITUDE	LONGITUDE	ENDLATITUDE	ENDLONGITUDE	STARTLOCATION	ENDLC
2	3	17	17	4	3	1133-201803212231	41.85124519	-87.74443372	41.8367474	-87.74393181	POINT (-87.7444337191 41.8512451897)	POINT (-87.743931
3	2	24	17	4	3	1134-201803212231	41.86596274	-87.7449852	41.85124519	-87.74443372	POINT (-87.7449852043 41.8659627382)	POINT (-87.744433
4	3	11	17	4	3	1153-201803212231	41.95381382	-87.80714975	41.96834367	-87.80698239	POINT (-87.8071497455 41.9538138239)	POINT (-87.806982
5	1	5	17	4	3	1211-201803212231	41.801585	-87.645156	41.808864	-87.645349	POINT (-87.645156 41.801585)	POINT (-87.64
6	1	5	17	4	3	1212-201803212231	41.808864	-87.645349	41.816184	-87.645535	POINT (-87.645349 41.808864)	POINT (-87.64
7	4	16	17	4	3	1213-201803212231	41.816184	-87.645535	41.823458	-87.645736	POINT (-87.645535 41.816184)	POINT (-87.64
8	2	17	17	4	3	1214-201803212231	41.823458	-87.645736	41.83075	-87.645915	POINT (-87.645736 41.823458)	POINT (-87.64
9	3	15	17	4	3	1215-201803212231	41.83075	-87.645915	41.838046	-87.646086	POINT (-87.645915 41.83075)	POINT (-87.64
10	4	40	17	4	3	1216-201803212231	41.838046	-87.646086	41.847196	-87.646297	POINT (-87.646086 41.838046)	POINT (-87.64
11	2	9	17	4	3	1217-201803212231	41.847196	-87.646297	41.852646	-87.646305	POINT (-87.646297 41.847196)	POINT (-87.64
12	4	18	17	4	3	1218-201803212231	41.852646	-87.646305	41.859912	-87.646517	POINT (-87.646305 41.852646)	POINT (-87.64
13	5	39	17	4	3	1219-201803212231	41.859912	-87.646517	41.867152	-87.646729	POINT (-87.646517 41.859912)	POINT (-87.64
14	3	23	17	4	3	1220-201803212231	41.867152	-87.646729	41.874338	-87.646935	POINT (-87.646729 41.867152)	POINT (-87.64
15	16	102	17	4	3	1300-201803212231	41.877947	-87.647241	41.878077	-87.636914	POINT (-87.647241 41.877947)	POINT (-87.63
16	5	21	17	4	3	1303-201803212231	41.873356	-87.617312	41.873192	-87.627565	POINT (-87.617312 41.873356)	POINT (-87.62
17	6	46	17	4	3	1304-201803212231	41.874238	-87.647125	41.874238	-87.639487	POINT (-87.647125 41.874238)	POINT (-87.63
18	4	25	17	4	3	1305-201803212231	41.874652	-87.627611	41.874538	-87.639487	POINT (-87.627611 41.874652)	POINT (-87.63
19	1	2	17	4	3	1306-201803212231	41.867374	-87.630336	41.874521	-87.630574	POINT (-87.630336 41.867374)	POINT (-87.63
20	4	19	17	4	3	1307-201803212231	41.874521	-87.630774	41.867374	-87.630536	POINT (-87.630774 41.874521)	POINT (-87.63
21	2	13	17	4	3	1308-201803212231	41.874238	-87.639487	41.874452	-87.627611	POINT (-87.639487 41.874238)	POINT (-87.62
22	3	13	17	4	3	1309-201803212231	41.874538	-87.639487	41.874438	-87.647125	POINT (-87.639487 41.874538)	POINT (-87.64
23	1	8	17	4	3	0974-201803212231	41.82324972	-87.67507424	41.82317531	-87.6849	POINT (-87.67507424 41.82324972)	POINT (-87.68
24	1	6	17	4	3	0975-201803212231	41.82327531	-87.66551	41.82324972	-87.67507424	POINT (-87.66551 41.82327531)	POINT (-87.6750
25	1	10	17	4	3	1132-201803212231	41.8367474	-87.74393181	41.82214155	-87.74377227	POINT (-87.7439318058 41.8367473974)	POINT (-87.743772
26	1	7	17	4	3	1170-201803212231	41.8942683	-87.80550079	41.87969928	-87.80494881	POINT (-87.8055007941 41.8942683013)	POINT (-87.804948



TIME	SEGMENT_ID	SPEED	STREET	DIRECTION	FROM	TO	LENGTH	HEADING	COMMENTS
3/21/2018 17:31	1133	23	Cicero	SB	Cermak	31st	1	S	Outside City Limits
3/21/2018 17:31	1134	20	Cicero	SB	Roosevelt	Cermak	1.02	S	Outside City Limits
3/21/2018 17:31	1153	14	Harlem	NB	Forest Preserve Ave	Gunnison	1	N	Outside City Limits
3/21/2018 17:31	1211	36	Halsted	NB	51st	47th	0.5	S	
3/21/2018 17:31	1212	35	Halsted	NB	47th	43rd	0.5	S	
3/21/2018 17:31	1213	27	Halsted	NB	43rd	Pershing	0.5	S	
3/21/2018 17:31	1214	23	Halsted	NB	Pershing	35th	0.5	S	
3/21/2018 17:31	1215	18	Halsted	NB	35th	31st	0.5	S	
3/21/2018 17:31	1216	17	Halsted	NB	31st	Archer	0.63	S	
3/21/2018 17:31	1217	23	Halsted	NB	Archer	Cermak	0.4	S	
3/21/2018 17:31	1218	12	Halsted	NB	Cermak	16th	0.5	S	
3/21/2018 17:31	1219	8	Halsted	NB	16th	Roosevelt	0.5	S	
3/21/2018 17:31	1220	20	Halsted	NB	Roosevelt	Harrison	0.5	S	
3/21/2018 17:31	1300	12	Jackson	EB	Halsted	Wacker	0.54	W	Oneway EB
3/21/2018 17:31	1303	19	Balbo	WB	LakeShore Dr	Michigan	0.52	E	
3/21/2018 17:31	1304	14	Harrison	EB	Halsted	Canal	0.4	W	
3/21/2018 17:31	1305	13	Harrison	WB	State	Canal	0.6	W	
3/21/2018 17:31	1306	24	Clark	NB	Roosevelt	Harrison	0.5	S	
3/21/2018 17:31	1307	20	Clark	SB	Harrison	Roosevelt	0.5	S	
3/21/2018 17:31	1308	12	Harrison	EB	Canal	State	0.6	W	
3/21/2018 17:31	1309	12	Harrison	WB	Canal	Halsted	0.4	W	
3/21/2018 17:31	974	23	Pershing	WB	Damen	Western	0.5	W	
3/21/2018 17:31	975	23	Pershing	WB	Ashland	Damen	0.5	W	
3/21/2018 17:31	1132	26	Cicero	SB	31st	Pershing	1.01	S	Outside City Limits
3/21/2018 17:31	1170	25	Harlem	SB	Chicago	Madison	1.01	N	Outside City Limits

6.2 OBJECTIVES

1. Input Design is the path toward changing over a customer arranged delineation of the commitment to a PC based structure.This structure is imperative to maintain a strategic distance from blunders in the information input procedure and demonstrate the right course to the administration for getting the right data from the mechanized framework.

2. It is accomplished by making easy to understand screens for the information passage to deal with a substantial volume of information. The objective of structuring input is to make information passage simpler and to be free from blunders. The information passage screen is structured so that every one of the information controls can be performed. It additionally gives record seeing offices.

3. At the point when the information is entered it will check for its legitimacy. Information can be entered with the assistance of screens. Appropriate messages are provided as when needed so that the userwill not be in maize of instant. Along these lines, the target of information configuration is to make an info format that is anything but difficult to pursue.

6.3 OUTPUT DESIGN

In output design, it is resolved how the data is to be uprooted for quick need and furthermore the printed version yield. It is the most imperative and direct source data to the client. Effective and smart yield configuration improves the framework's relationship to help client basic leadership.

1. Planning PC yield ought to continue in a composed, very much considered way; the correct yield must be created while guaranteeing that each yield component is structured with the goal that individuals will discover the framework can utilize effectively and adequately. At the point when investigation structure PC yield, they ought to identify the particular yield that is expected to meet the prerequisites.

2. Select strategies for introducing the data.

3. Make a record, report, or different configurations that contain data created by the framework.

The yield type of data framework ought to achieve at least one of the accompanying goals.

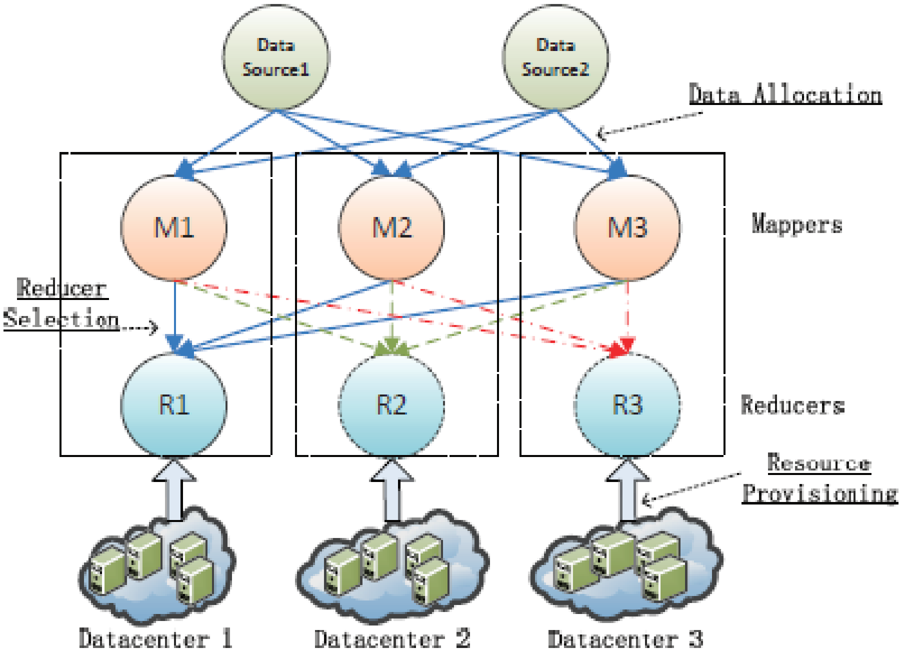
1) Convey information about past activities, current status or projections

2) Future

3) Signal basic events, openings, issues, or exhortations.

4)Trigger an activity.

5)Confirm an activity.



6.3 Architecture Diagram

Without loss of generality, we consider such a system scenario where a DSP (Data Service Provider) manages multiple data sources and transfers all the data into cloud for processing using MapReduce. The DSP may either deploy its private datacenters (e.g., Google deploys tens of datacenters over the world) or rent the resource from public clouds (e.g., SKA may rent the resource from public cloud such as Amazon EC2). Specially, for the DSP that have its private cloud, datasources overlaps datacenters since generated data are collected and stored in its own datacenters. System architecture is presented in Fig. 1: Data sources from multiple geographical data locations continuously produce massive data. Data analysis applications are deployed in the cloud and the data sources is connected to datacenters located in multiple places. In this model, data are moved to the datacenters once they are generated and are processed in a incremental style in which only the newly arrived data are computed and the intermediate data from past can be reused. Specifically, both mappers and reducers are running on every datacenter. As the *GEO* execution path mentioned above is considered in this paper, there are two corresponding phases for the data moving procedure. At the first phase, data can be moved to any datacenter for Map operation. At the second phase, the intermediate data of Mappers must be moved into a single datacenter with consideration of data correlations. As shown in Fig. 1, the bold line is an example of execution path, which shows that the raw data from data source 1 and data source 2 are moved to multiple datacenters for Map operation and then the output data of Mappers are aggregated into the Reducer in datacenter 1 for Reduce operation.

Formally, let D be the set of geographically distributed data- centerswithsizeofD=|D|(indexedbyd (1≤d≤D) andK be the set of VM types with size K = |K|, each of which has a specific capacity v_k with configurations such as CPU and memory. All types of VMs can be provisioned in each datacenter. Data are dynamically and continuous generated from R = |R| different datasource locations (indexed by r,1 ≤ r ≤ R), denoted as a set R. Data from any location can be moved to any datacenter for Map operation and then aggregate the intermediate data into a single datacenter. To be realistic, we assume that the bandwidth $B_{rd}$from data location r to datacenter d is limited. Also note that inter-datacenter links (e.g., trans-oceanic links) are expensive to lay down, so the costs of using these links are considered as a first-order entity when migrating the intermediate data among datacenters. In addition, the data generation in each location is independent and the prices of the resource (e.g., VM) in each datacenter are varied in both spatial and temporal domain.

CHAPTER 7

SYSTEM TESTING

The purpose of testing is to discover goofs. Testing is the route toward endeavouring to locate every conceivable defect or deficiency in a work thing. It gives a way to deal with check the value of parts, sub-social affairs, and assemblages or possibly a finished thing.It is the way toward taking a shot at programming with the target of guaranteeing that the Software structure fulfils its necessities and client needs and does not tumble in an unacceptable way.

TYPES OF TESTS:

UNIT TESTING

Unit testing incorporates the structure of analyses that favour that the inside program justification is working suitably and that program inputs produce considerable yields. Every decision branch and inward code stream should be endorsed. It is the trying of individual programming units of the application. It is done after the fulfilment of an individual unit before joining. This is essentially attempting, that relies upon the data of its advancement and is meddling. Unit tests perform essential tests at the part level and test a particular business procedure, application, and additionally framework design. Unit tests ensure that each exceptional method for a business technique performs exactly to the chronicled conclusions and contains clearly portrayed data sources and foreseen results.

Unit testing is commonly coordinated as a noteworthy part of a solidified code and unit trial of the item lifecycle, notwithstanding the way that it isn't phenomenal for coding and unit testing to be driven as two specific stages

Test strategy and approach:

Field testing will be performed manually and functional tests will be written in detail.

Test objectives:

1) All field sections must work legitimately.

2) Pages must be enacted from the distinguished connection.

3) The section screen, messages and reactions must not be postponed.

Features to be tested:

1) Confirm that the sections are of the right configuration.

2) No copy passages ought to be permitted.

3) All connections should take the client to the right page.

INTEGRATION TESTING

Integration tests are intended to test coordinated programming segments to decide whether they really keep running as one program. Testing is event driven and is progressively stressed over the essential consequence of screens or fields. Integration tests display that notwithstanding the way that the portions were autonomously satisfaction, as showed up by successfully unit testing, the blend of parts is correct and dependable. Integration testing is unequivocally away for revealing the issues that rise up out of the mix of parts.

The errand of the incorporation test is to watch that parts or programming applications, for example parts in a product framework or - one stage up - programming applications at the organization level - communicate without mistake.

Test outcomes: All the experiments referenced above passed effectively. No irregularities have been experienced

ACCEPTANCE TESTING

User Acceptance Testing is a basic obligation of any undertaking and requires noteworthy support by the end client. It additionally guarantees that the framework meets practical prerequisites.

Test outcomes: All the experiments referenced above passed effectively. No deformities experienced.

FUNCTIONAL TESTING

Functional tests give exact shows that limits attempted are available as dictated by the business and specific necessities, system documentation, and customer manuals. Functional testing is focused on the accompanying things:

Valid Input: Latitudes and Longitudes of locations, source and destination

Invalid Input: corrupted data, unrecognized data type

Functions: map out distances and select shortest one, scenario based decision making

Output: graphs on time taken and data traffic

Frameworks/Procedures: interfacing structures or procedure must be gathered..

Prerequisites, key capacities, or exceptional experiments are what association and planning of practical tests centered around. In addition, precise consideration identifying with perceiving Business process streams; data fields, predefined structures, and dynamic techniques must be considered for testing.

SYSTEM TESTING

System testing ensures that the entire facilitated programming system meets necessities. It tests a setup to ensure known and obvious results. An instance of framework testing is the structure arranged system fuse test. System testing relies upon method depictions and streams, focusing on pre-driven methodology associations and compromise centres.

WHITE BOX TESTING

White Box Testing is a trying in which the product analyzer knows about the internal functions, structure and language of the product, or if nothing else its motivation. It is the reason. It is utilized to test territories that can't become from a discovery level.

BLACK BOX TESTING

Black box Testing will attempt the item with no data about the inside exercises, structure or language of the module being attempted. Black box tests, as most unique sorts of tests, must be made out of a total source record, for instance, detail or necessities chronicle, for instance, specific or requirements report. It is an attempting in which the item under test is managed, as a revelation .you can't "see" into it. The test gives data sources and reacts to yields without thinking about how the product functions.

TEST CASE ID	TEST CASE NAME	TEST CASE DESCRIPTION	TEST STEPS			TEST STATUS (P/F)
			STEPS	EXPECTED RESULT	ACTUAL RESULT	
1	Out of city bounds	In case a destination location lies outside the periphery of the city (or a set distance)	Input source and destination latitude and longitude	Error message shown saying out of bounds distance	Error message displayed	P
2	Invalid distance input	Distance 0 or negative	Check if source and destination are same	Identified classes of valid input must be accepted	Identified classes of valid input are accepted	P
3	Linear/ Linear Big	At least one transfer location is same	Algorithm used based on description	All the experiments referenced above must pass effectively. No imperfections should be experienced.	All the experiments referenced above passed effectively. No imperfections experienced.	P

4	Join / Join Big	Distance to be covered by data same	Algorithm used based on description	All the experiments referenced above must pass effectively. No deformities should be experienced.	All the experiments referenced above passed effectively. No deformities experienced.	P

CHAPTER 8

CONCLUSION

Big Data handling crosswise over geologically appropriated server farms are turning into an alluring and financially savvy procedure for some, huge information organizations and associations with fast and a high volume of huge information produced from topographically scattered sources. A methodological system for compelling information development, asset provisioning and reducer choice are created in this paper with the target of cost minimization. We balance five sorts of expenses between the two MapReduce stages crosswise over datacenters: cost of transmission capacity, cost of capacity, cost of registering, cost of relocation, and cost of dormancy. This mind-boggling cost enhancement issue is detailed by limiting the five cost factors at the same time into a joint stochastic number nonlinear streamlining issue. By utilizing the MapReduce and Clustering system, we change the first issue into three autonomous sub-issues that can be understood by planning a productive online calculation to limit the long haul normal expense of activity.

We lead a hypothetical examination to exhibit the cost-ideal and most pessimistic scenario postpone the adequacy of our calculation. We direct a test assessment utilizing genuine follow dataset to approve our calculation's hypothetical result and prevalence by looking at it over the existing run of the mill approaches and disconnected techniques.

CHAPTER 9

FUTURE ENHANCEMENTS

The Future approach can be extended with minor extensions to adapt to these cases. For example, if we change the original model by adding a data flow of self-circulation at the allocation stage and designing a coordinator to transfer the reduced result to mappers, iterative jobs can be supported, taking into account the model's data replication factor. Data replication is well known as a high-availability and high-fault tolerance effective solution.

Given that our goal is cost minimization, introducing data replication will add additional cost of replicating data across datacenters. Thus, this factor is not considered in our current cost minimization algorithm and we left it to be one of our ongoing research efforts. 3) In addition, we will concentrate on deploying the proposed algorithm in the real systems such as Amazon EC2 to further validate its effectiveness.

REFERENCES

[1] A. Vulimiri, C. Curino, B. Godfrey, T. Jungblut, J. Padhye, and G. Varghese, “Global analytics in the face of bandwidth and regulatory constraints,” in *Proceedings of the USENIX NSDI’15*, 2015.

[2] J. Dean and S. Ghemawat, “Mapreduce: Simplified data processing on large clusters,” *Communications of the ACM*, vol. 51, no. 1, pp. 107–113, 2008.

[3] M. Zaharia, M. Chowdhury, M. J. Franklin, S. Shenker, and I. Stoica, “Spark: Cluster computing with working sets,” in *Proceedings of the USENIX HotCloud’10*, 2010.

[4] E. E. Schadt, M. D. Linderman, J. Sorenson, L. Lee, and G. P.Nolan, “Computational solutions to large-scale data management and analysis,” *Nature Reviews Genetics*, vol. 11, no. 9, pp. 647–657, 2010.

[5] M. Cardosa, C. Wang, A. Nangia*et al.*, “Exploring mapreduce efficiency with highly-distributed data,” in *Proceedings of the second international workshop on MapReduce and its applications*, 2011.

[6] L. Zhang, C.Wu, Z. Li, C. Guo, M. Chen, and F. C. M. Lau, “Moving big data to the cloud: An online cost-minimizing approach,” *IEEE Journal on Selected Areas in Communications*, vol. 31, pp. 2710–2721, 2013.

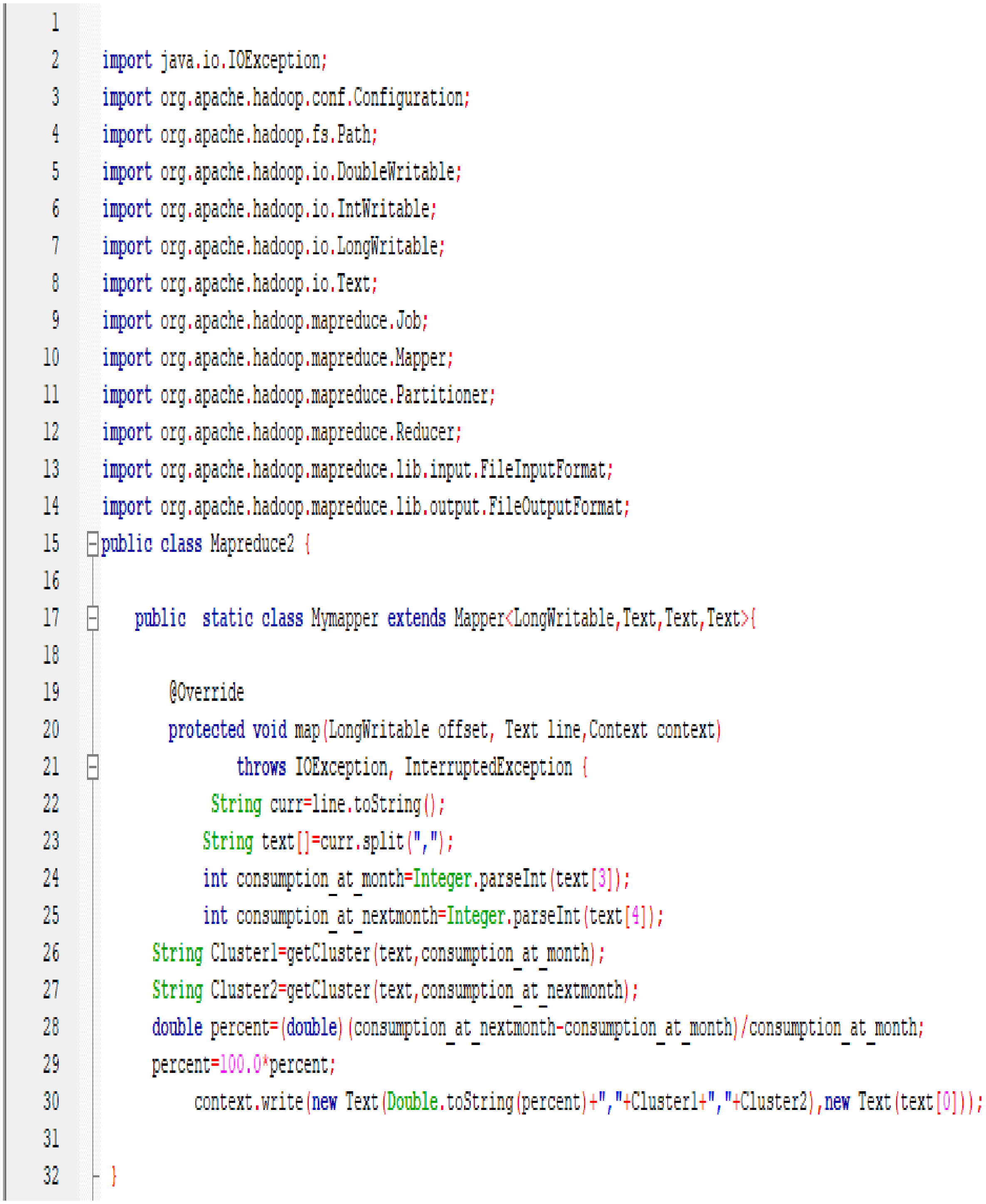
[7] W. Yang, X. Liu, L. Zhang, and L. T. Yang, “Big data real-time processing based on storm,” in *Proceedings of the IEEE TrustCom’13*, 2013.

[8] Y. Zhang, S. Chen, Q. Wang, and G. Yu, “i2mapreduce: Incremental mapreduce for mining evolving big data,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 27, pp. 1906–1919, 2015.

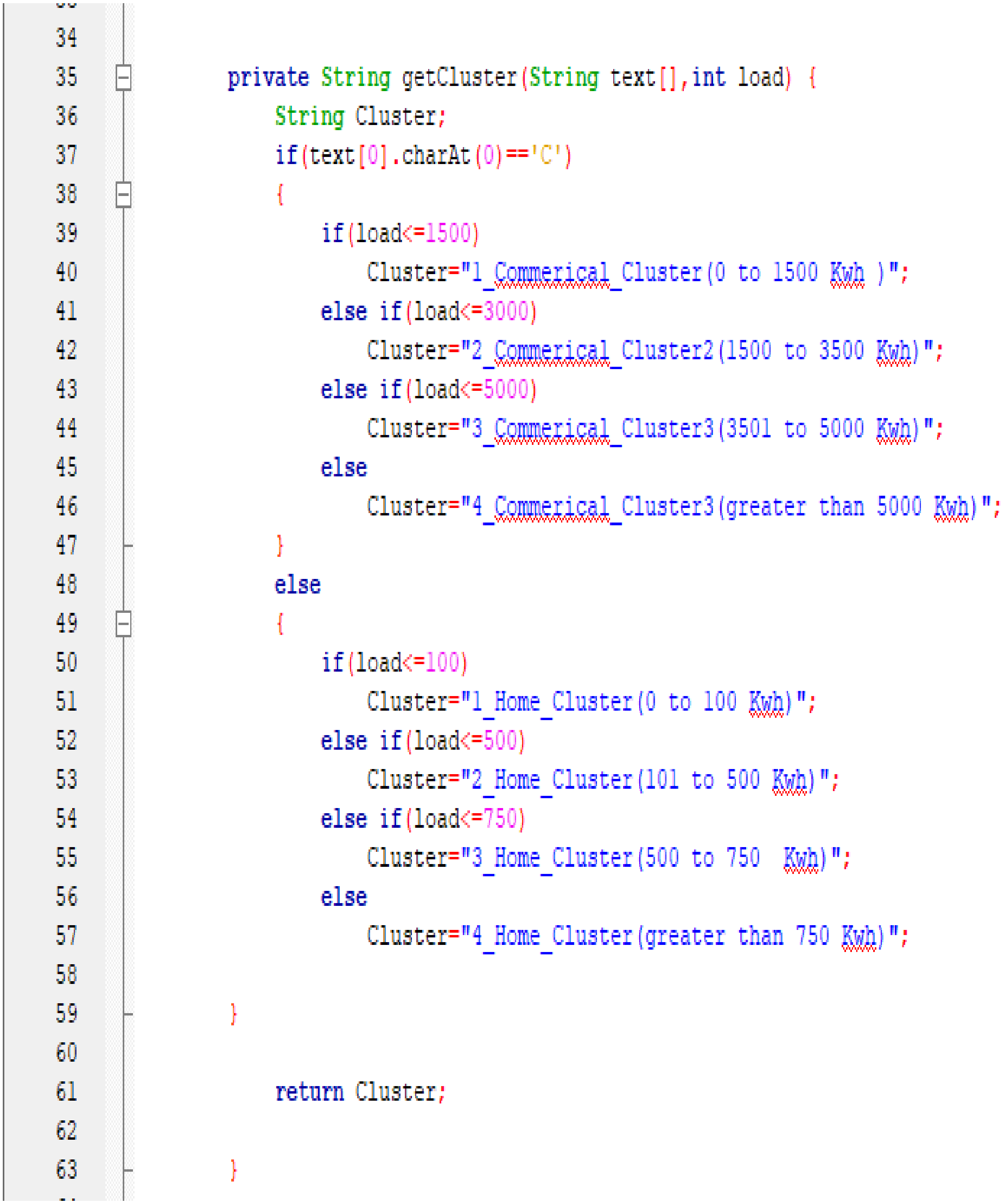
[9] D. Lee, J. S. Kim, and S. Maeng, “Large-scale incremental processing with mapreduce,” *Future Generation Computer Systems,* vol. 36, no. 7, pp. 66–79, 2014.

[10] “Square kilometre array,” http://www.skatelescope.org/.

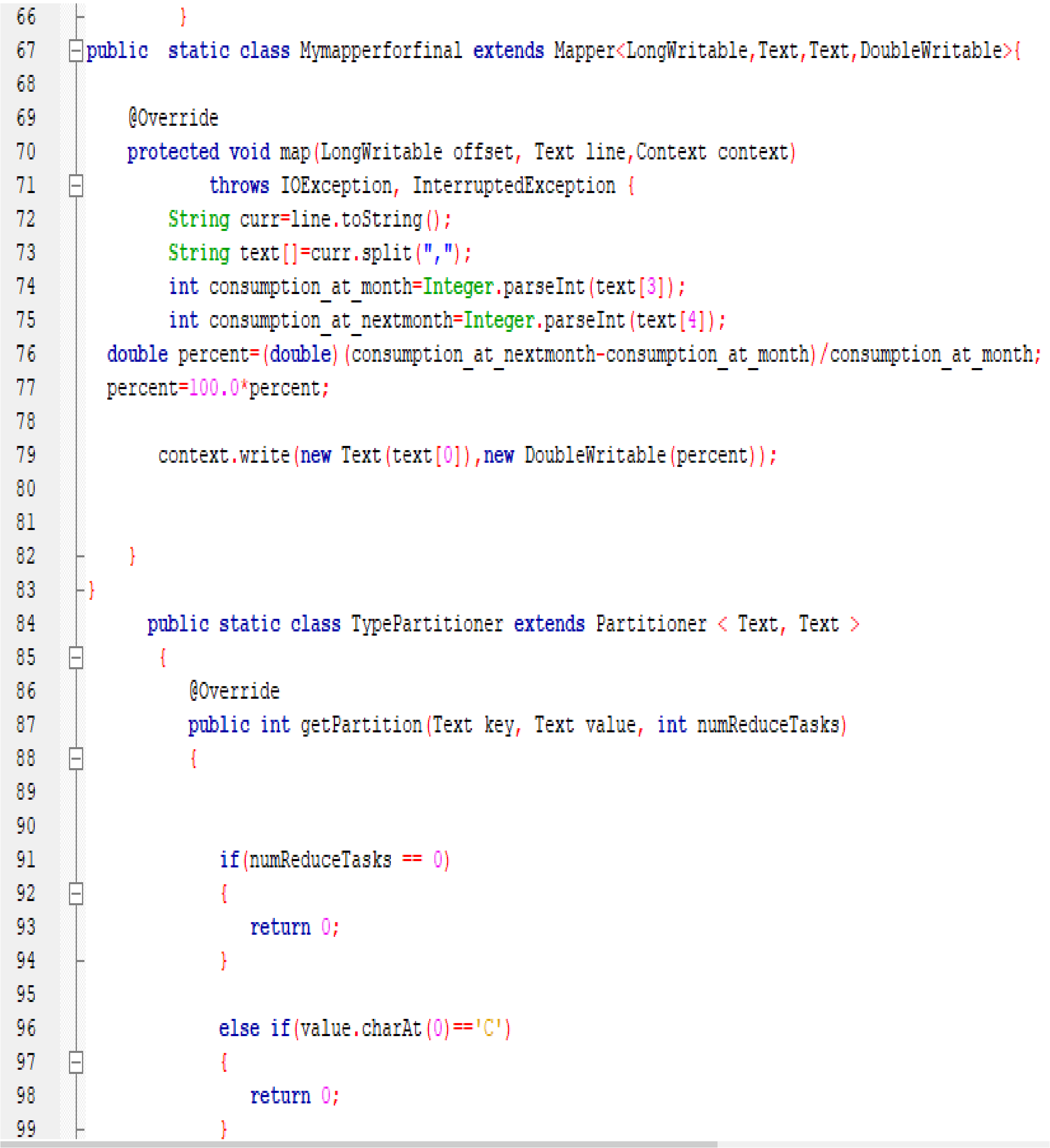
APPENDIX



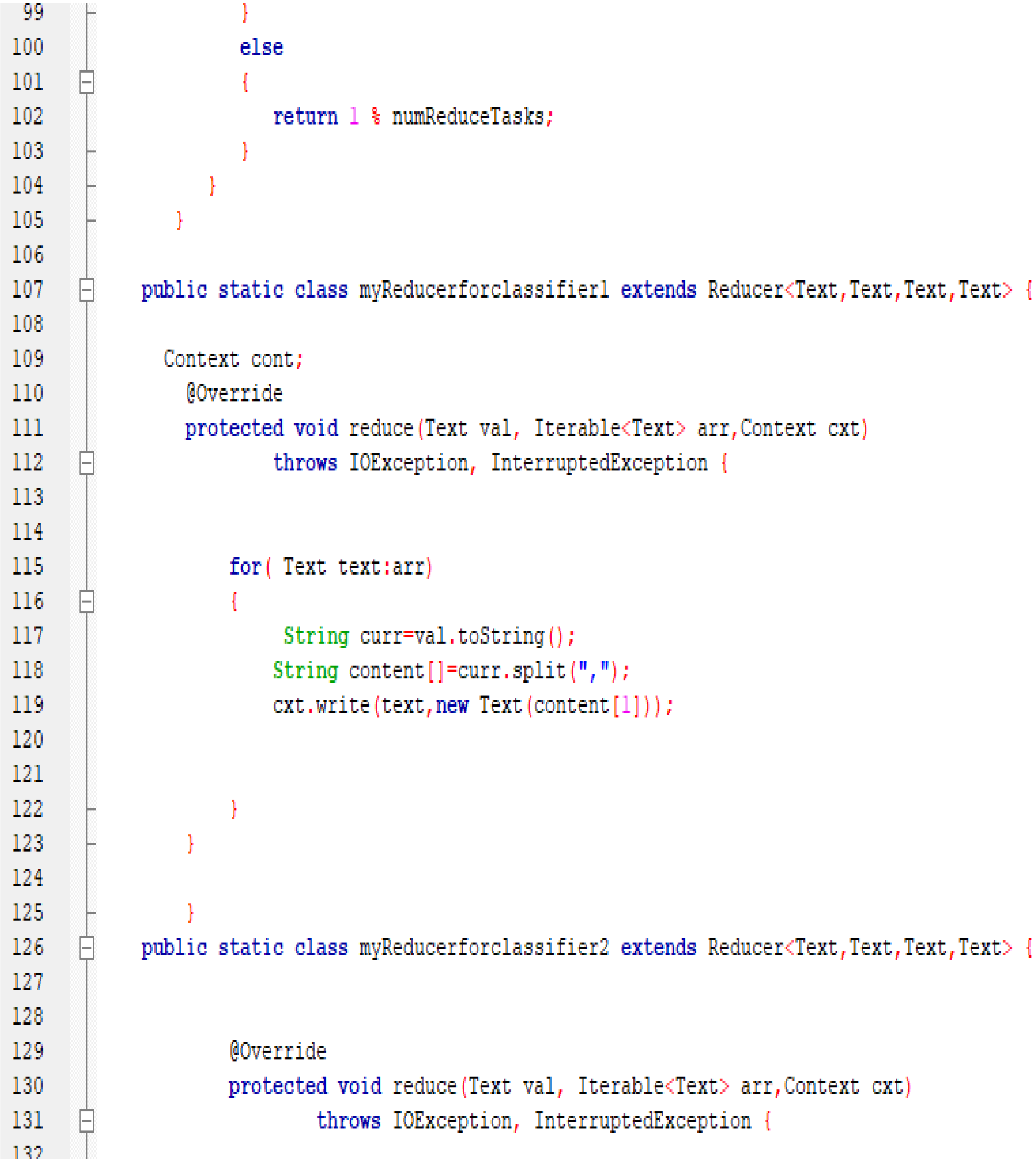
```java

import java.io.IOException;
import org.apache.hadoop.conf.Configuration;
import org.apache.hadoop.fs.Path;
import org.apache.hadoop.io.DoubleWritable;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Job;
import org.apache.hadoop.mapreduce.Mapper;
import org.apache.hadoop.mapreduce.Partitioner;
import org.apache.hadoop.mapreduce.Reducer;
import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;
public class Mapreduce2 {

    public  static class Mymapper extends Mapper<LongWritable,Text,Text,Text>{

        @Override
        protected void map(LongWritable offset, Text line,Context context)
                throws IOException, InterruptedException {
             String curr=line.toString();
            String text[]=curr.split(",");
            int consumption_at_month=Integer.parseInt(text[3]);
            int consumption_at_nextmonth=Integer.parseInt(text[4]);
      String Cluster1=getCluster(text,consumption_at_month);
      String Cluster2=getCluster(text,consumption_at_nextmonth);
      double percent=(double)(consumption_at_nextmonth-consumption_at_month)/consumption_at_month;
      percent=100.0*percent;
           context.write(new Text(Double.toString(percent)+","+Cluster1+","+Cluster2),new Text(text[0]));

 }
```



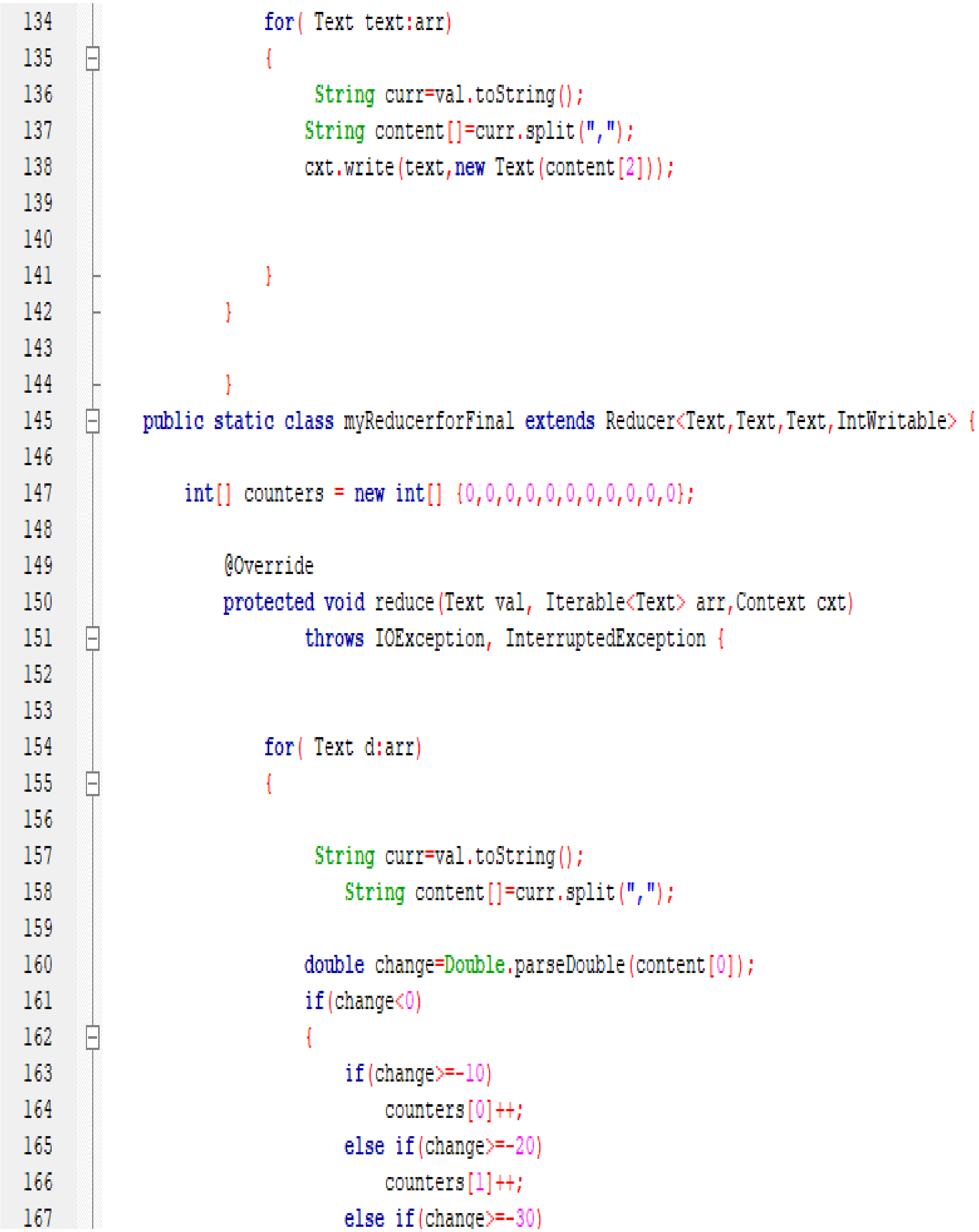
```
private String getCluster(String text[],int load) {
    String Cluster;
    if(text[0].charAt(0)=='C')
    {
        if(load<=1500)
            Cluster="1_Commerical_Cluster(0 to 1500 Kwh )";
        else if(load<=3000)
            Cluster="2_Commerical_Cluster2(1500 to 3500 Kwh)";
        else if(load<=5000)
            Cluster="3_Commerical_Cluster3(3501 to 5000 Kwh)";
        else
            Cluster="4_Commerical_Cluster3(greater than 5000 Kwh)";
    }
    else
    {
        if(load<=100)
            Cluster="1_Home_Cluster(0 to 100 Kwh)";
        else if(load<=500)
            Cluster="2_Home_Cluster(101 to 500 Kwh)";
        else if(load<=750)
            Cluster="3_Home_Cluster(500 to 750  Kwh)";
        else
            Cluster="4_Home_Cluster(greater than 750 Kwh)";

}


    return Cluster;


}
```



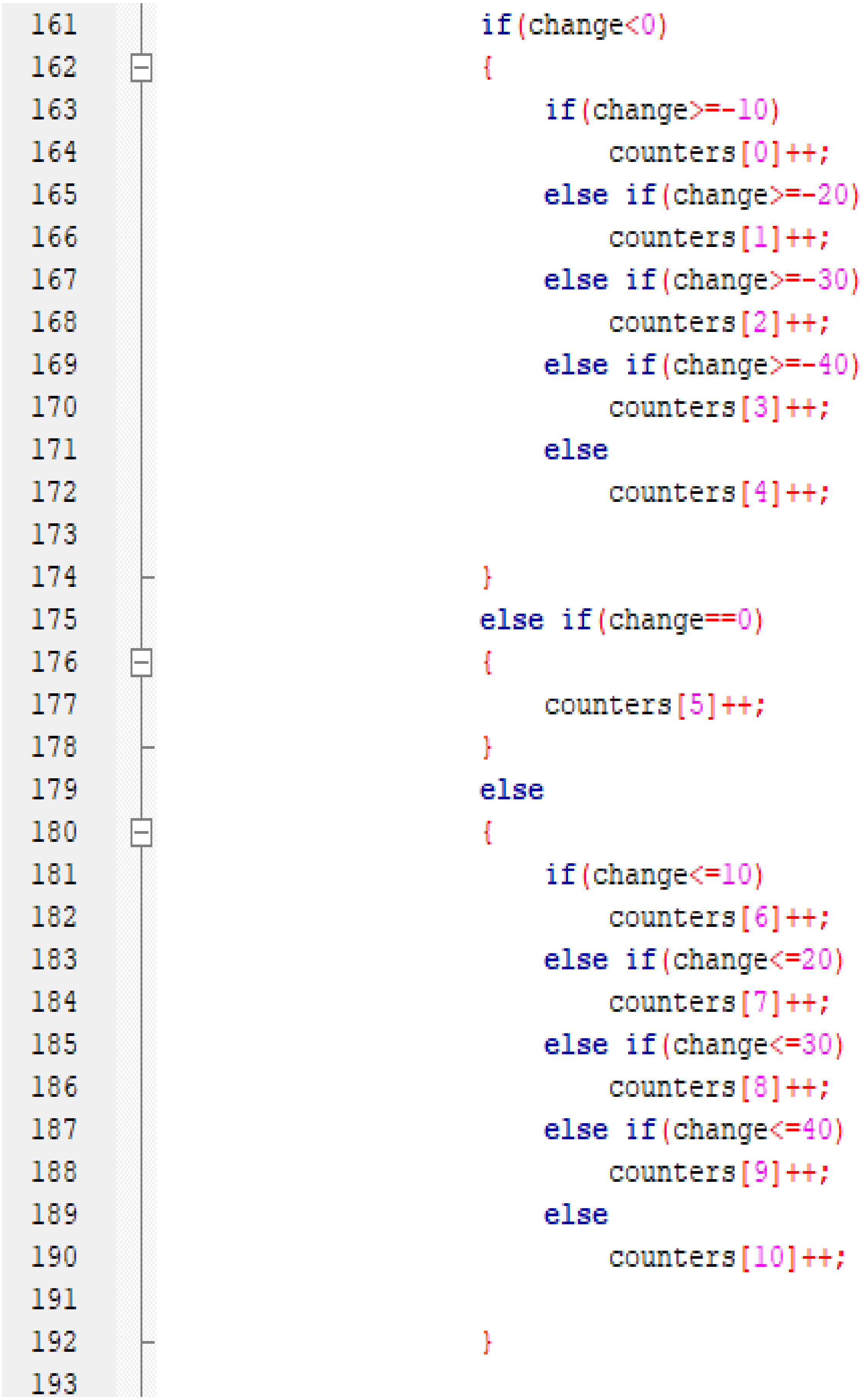
```
         }
public  static class Mymapperforfinal extends Mapper<LongWritable,Text,Text,DoubleWritable>{

    @Override
    protected void map(LongWritable offset, Text line,Context context)
            throws IOException, InterruptedException {
        String curr=line.toString();
        String text[]=curr.split(",");
        int consumption_at_month=Integer.parseInt(text[3]);
        int consumption_at_nextmonth=Integer.parseInt(text[4]);
  double percent=(double)(consumption_at_nextmonth-consumption_at_month)/consumption_at_month;
  percent=100.0*percent;

       context.write(new Text(text[0]),new DoubleWritable(percent));


    }
}
      public static class TypePartitioner extends Partitioner < Text, Text >
       {
          @Override
          public int getPartition(Text key, Text value, int numReduceTasks)
          {


             if(numReduceTasks == 0)
             {
                return 0;
             }

             else if(value.charAt(0)=='C')
             {
                return 0;
             }
```



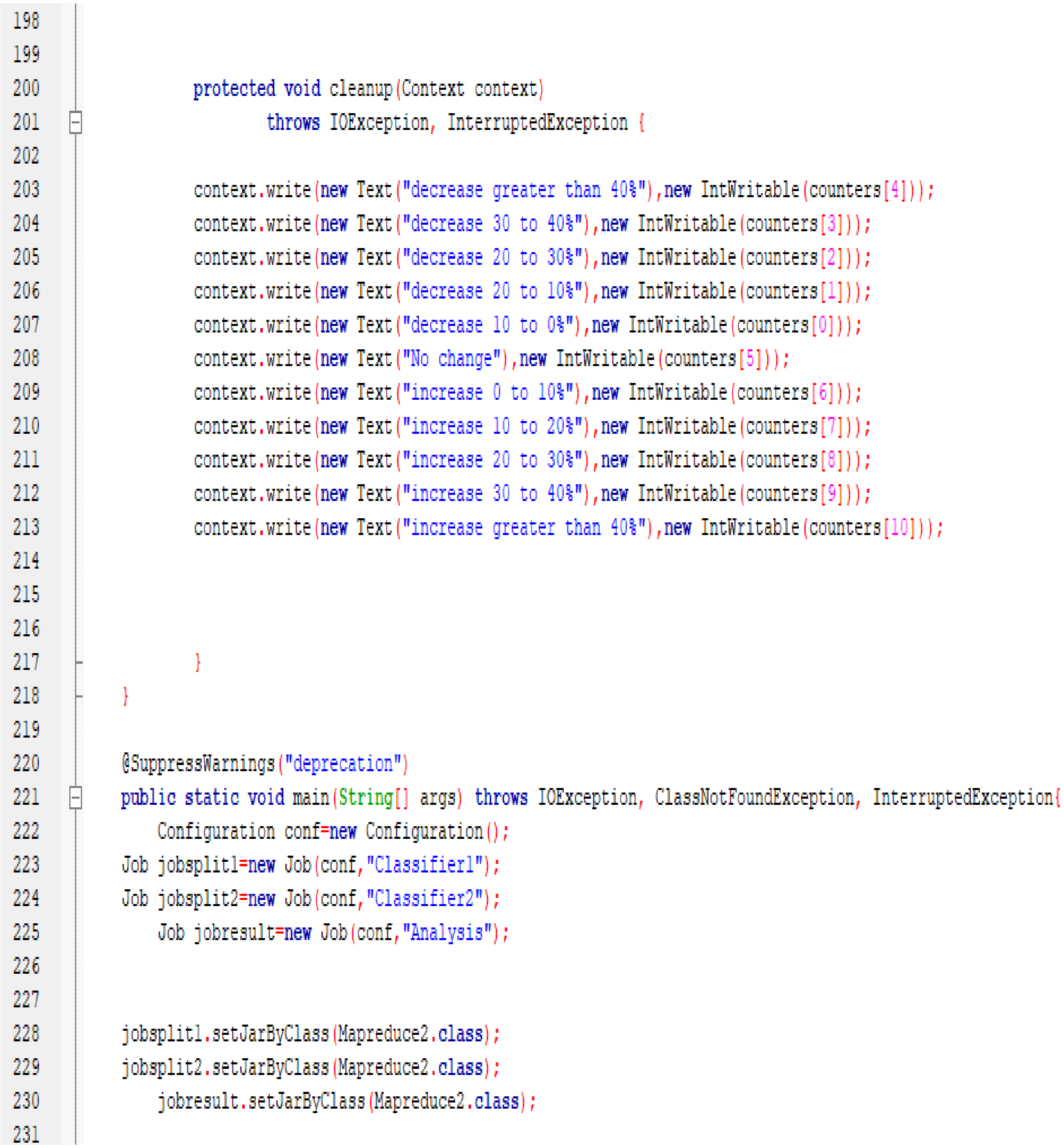
```
         }
         else
         {
            return 1 % numReduceTasks;
         }
      }
   }

public static class myReducerforclassifier1 extends Reducer<Text,Text,Text,Text> {

  Context cont;
    @Override
    protected void reduce(Text val, Iterable<Text> arr,Context cxt)
            throws IOException, InterruptedException {


        for( Text text:arr)
        {
             String curr=val.toString();
            String content[]=curr.split(",");
            cxt.write(text,new Text(content[1]));


        }
    }

    }
public static class myReducerforclassifier2 extends Reducer<Text,Text,Text,Text> {


        @Override
        protected void reduce(Text val, Iterable<Text> arr,Context cxt)
                throws IOException, InterruptedException {
```



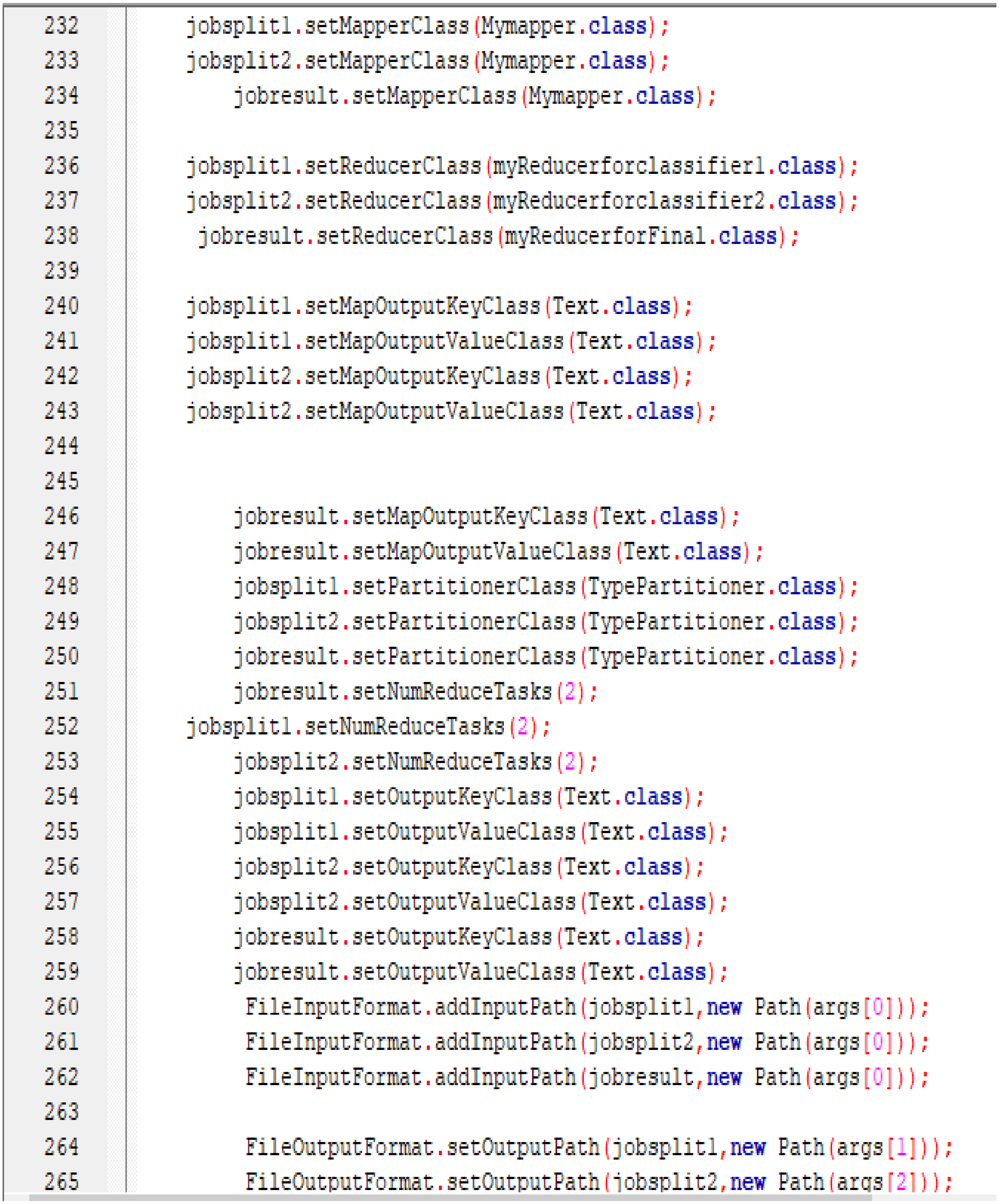
```
            for( Text text:arr)
            {
                 String curr=val.toString();
                String content[]=curr.split(",");
                cxt.write(text,new Text(content[2]));


            }
        }

        }
public static class myReducerforFinal extends Reducer<Text,Text,Text,IntWritable> {

    int[] counters = new int[] {0,0,0,0,0,0,0,0,0,0,0};

        @Override
        protected void reduce(Text val, Iterable<Text> arr,Context cxt)
                throws IOException, InterruptedException {


            for( Text d:arr)
            {

                 String curr=val.toString();
                    String content[]=curr.split(",");

                double change=Double.parseDouble(content[0]);
                if(change<0)
                {
                    if(change>=-10)
                        counters[0]++;
                    else if(change>=-20)
                        counters[1]++;
                    else if(change>=-30)
```



```
if(change<0)
{
    if(change>=-10)
        counters[0]++;
    else if(change>=-20)
        counters[1]++;
    else if(change>=-30)
        counters[2]++;
    else if(change>=-40)
        counters[3]++;
    else
        counters[4]++;

}
else if(change==0)
{
    counters[5]++;
}
else
{
    if(change<=10)
        counters[6]++;
    else if(change<=20)
        counters[7]++;
    else if(change<=30)
        counters[8]++;
    else if(change<=40)
        counters[9]++;
    else
        counters[10]++;

}
```

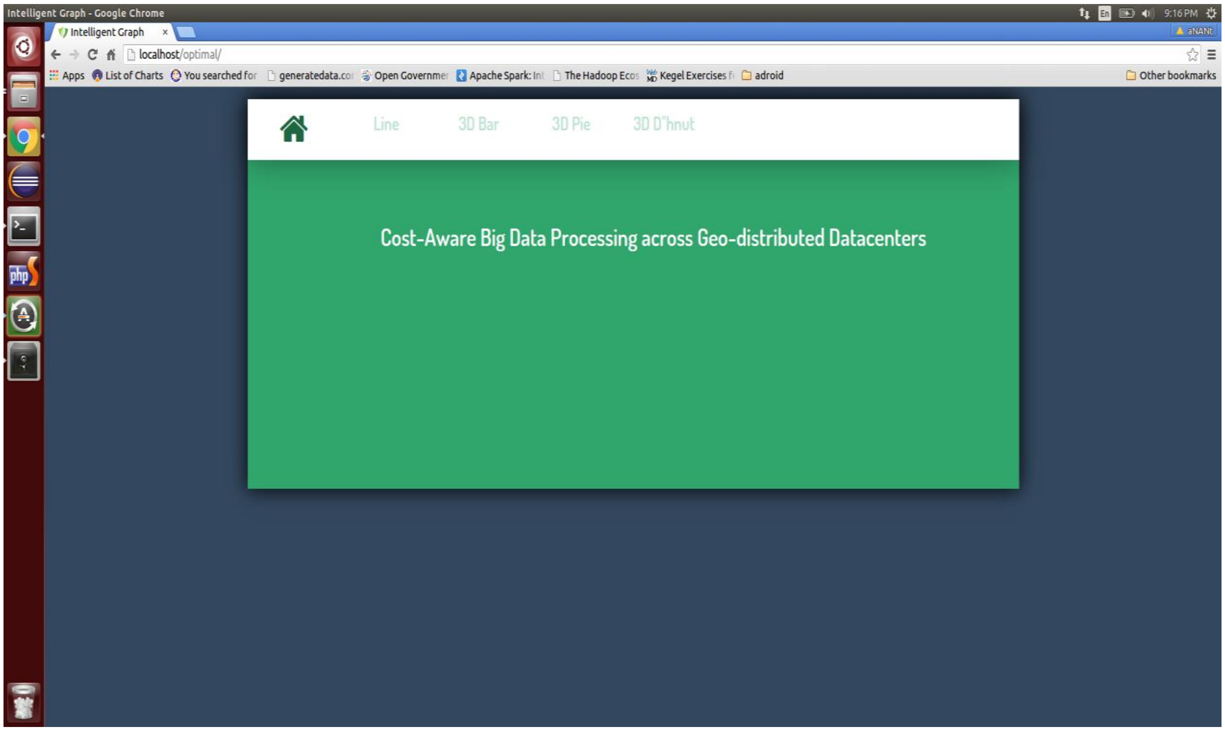


```
        protected void cleanup(Context context)
                throws IOException, InterruptedException {

        context.write(new Text("decrease greater than 40%"),new IntWritable(counters[4]));
        context.write(new Text("decrease 30 to 40%"),new IntWritable(counters[3]));
        context.write(new Text("decrease 20 to 30%"),new IntWritable(counters[2]));
        context.write(new Text("decrease 20 to 10%"),new IntWritable(counters[1]));
        context.write(new Text("decrease 10 to 0%"),new IntWritable(counters[0]));
        context.write(new Text("No change"),new IntWritable(counters[5]));
        context.write(new Text("increase 0 to 10%"),new IntWritable(counters[6]));
        context.write(new Text("increase 10 to 20%"),new IntWritable(counters[7]));
        context.write(new Text("increase 20 to 30%"),new IntWritable(counters[8]));
        context.write(new Text("increase 30 to 40%"),new IntWritable(counters[9]));
        context.write(new Text("increase greater than 40%"),new IntWritable(counters[10]));



        }
}

@SuppressWarnings("deprecation")
public static void main(String[] args) throws IOException, ClassNotFoundException, InterruptedException{
    Configuration conf=new Configuration();
Job jobsplit1=new Job(conf,"Classifier1");
Job jobsplit2=new Job(conf,"Classifier2");
    Job jobresult=new Job(conf,"Analysis");


jobsplit1.setJarByClass(Mapreduce2.class);
jobsplit2.setJarByClass(Mapreduce2.class);
    jobresult.setJarByClass(Mapreduce2.class);
```

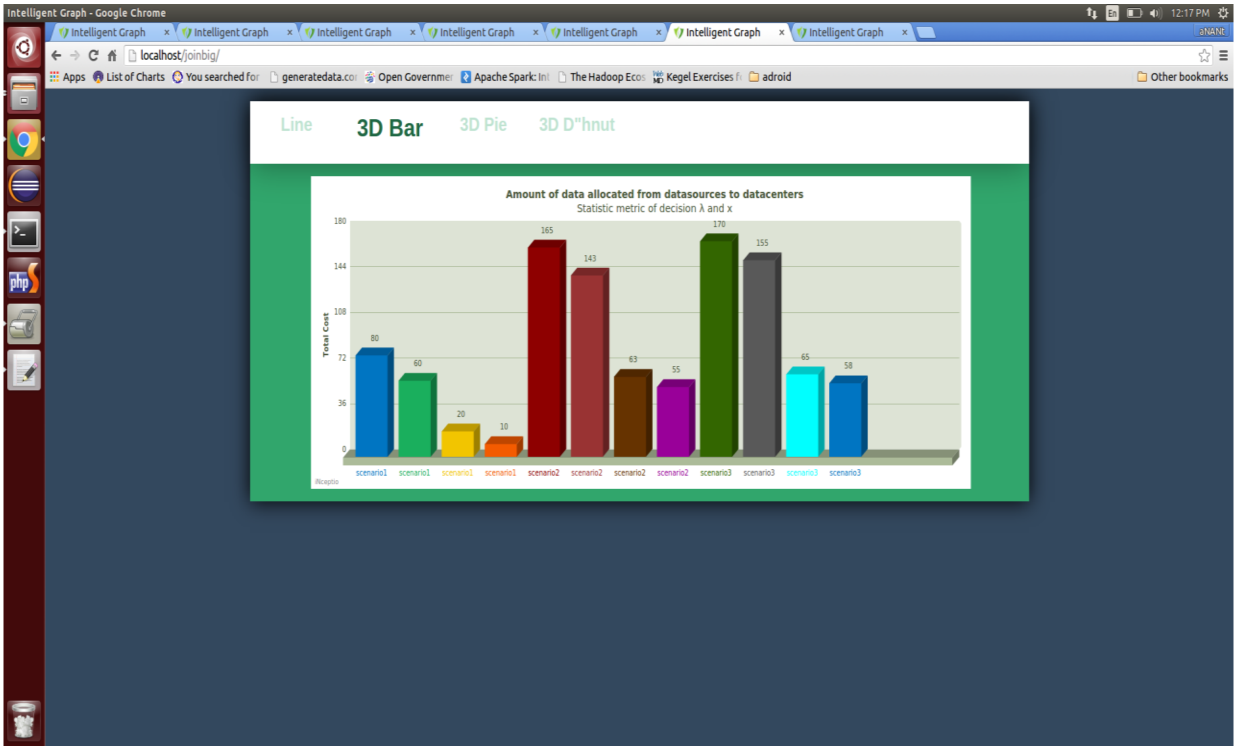


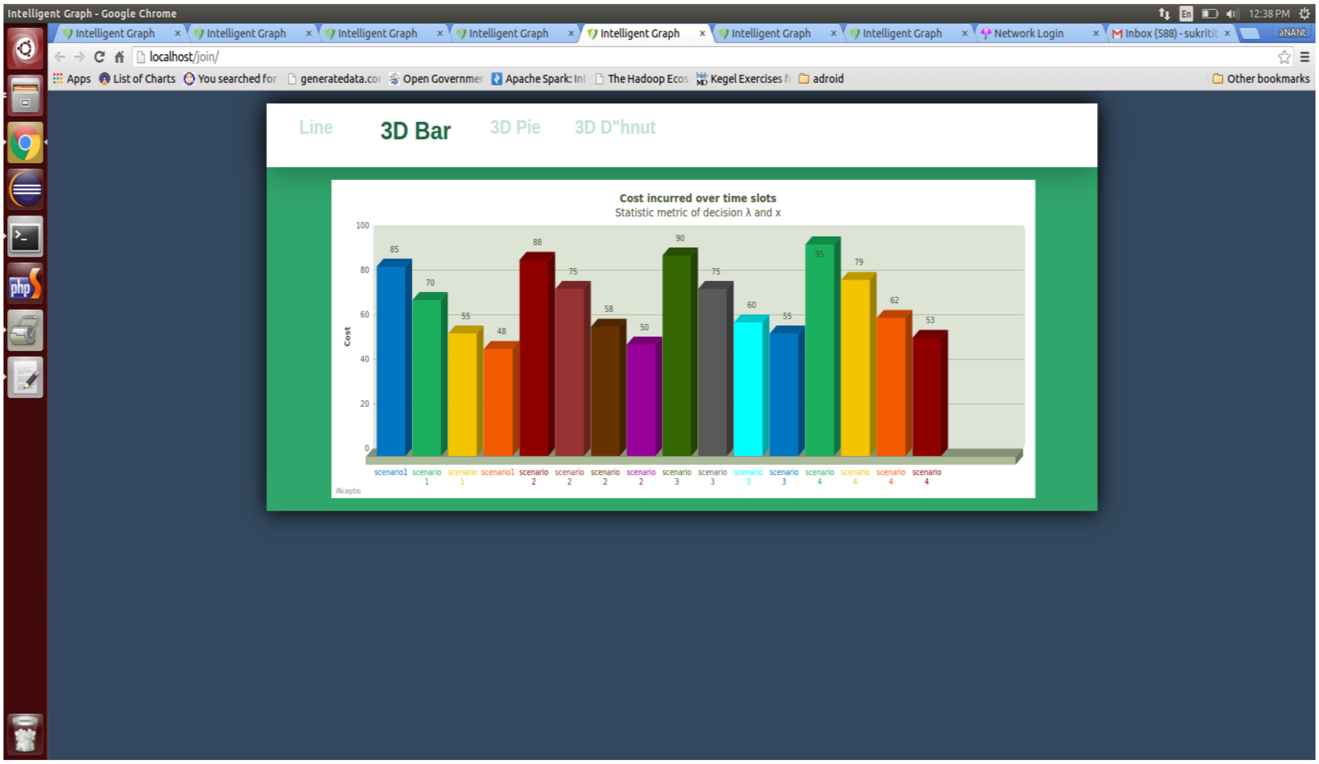
```
        jobsplit1.setMapperClass(Mymapper.class);
        jobsplit2.setMapperClass(Mymapper.class);
            jobresult.setMapperClass(Mymapper.class);

        jobsplit1.setReducerClass(myReducerforclassifier1.class);
        jobsplit2.setReducerClass(myReducerforclassifier2.class);
         jobresult.setReducerClass(myReducerforFinal.class);

        jobsplit1.setMapOutputKeyClass(Text.class);
        jobsplit1.setMapOutputValueClass(Text.class);
        jobsplit2.setMapOutputKeyClass(Text.class);
        jobsplit2.setMapOutputValueClass(Text.class);


            jobresult.setMapOutputKeyClass(Text.class);
            jobresult.setMapOutputValueClass(Text.class);
            jobsplit1.setPartitionerClass(TypePartitioner.class);
            jobsplit2.setPartitionerClass(TypePartitioner.class);
            jobresult.setPartitionerClass(TypePartitioner.class);
            jobresult.setNumReduceTasks(2);
        jobsplit1.setNumReduceTasks(2);
            jobsplit2.setNumReduceTasks(2);
            jobsplit1.setOutputKeyClass(Text.class);
            jobsplit1.setOutputValueClass(Text.class);
            jobsplit2.setOutputKeyClass(Text.class);
            jobsplit2.setOutputValueClass(Text.class);
            jobresult.setOutputKeyClass(Text.class);
            jobresult.setOutputValueClass(Text.class);
             FileInputFormat.addInputPath(jobsplit1,new Path(args[0]));
             FileInputFormat.addInputPath(jobsplit2,new Path(args[0]));
             FileInputFormat.addInputPath(jobresult,new Path(args[0]));

             FileOutputFormat.setOutputPath(jobsplit1,new Path(args[1]));
             FileOutputFormat.setOutputPath(jobsplit2,new Path(args[2]));
```

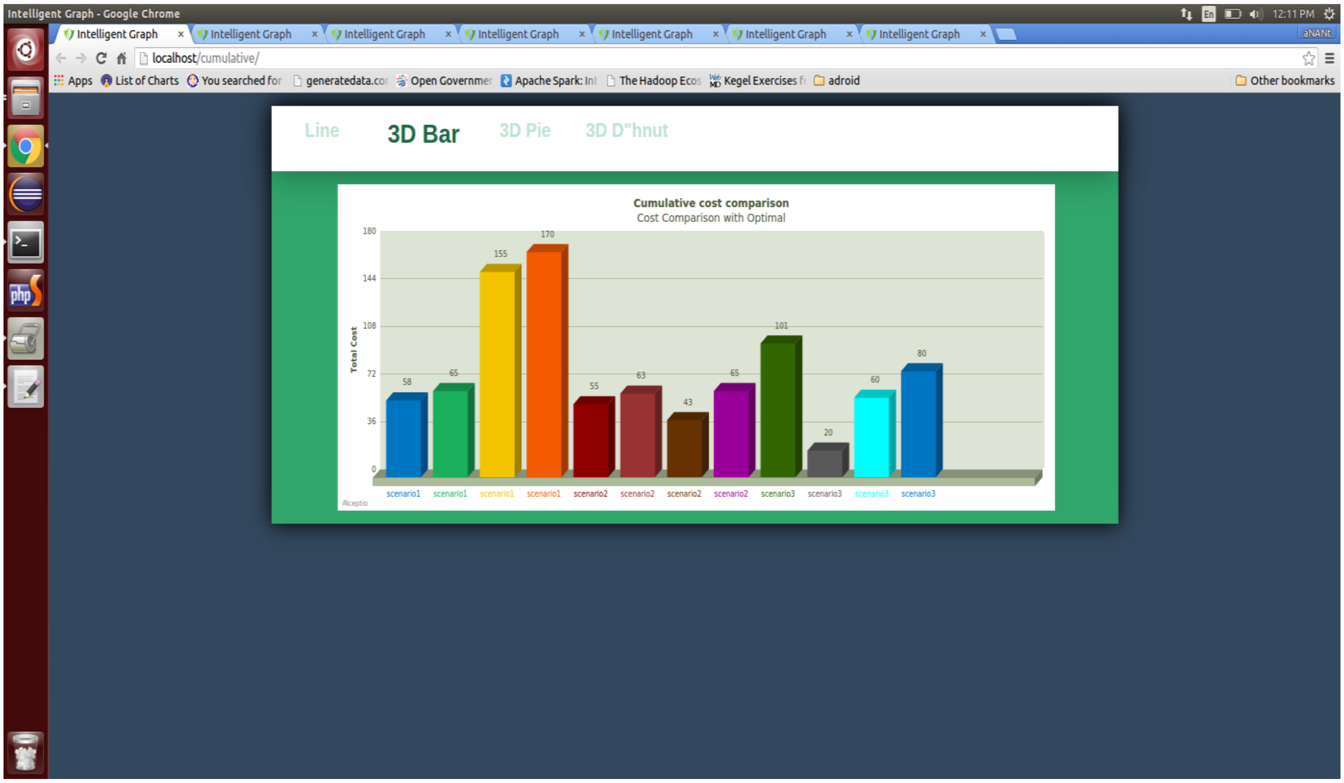
OUTPUT

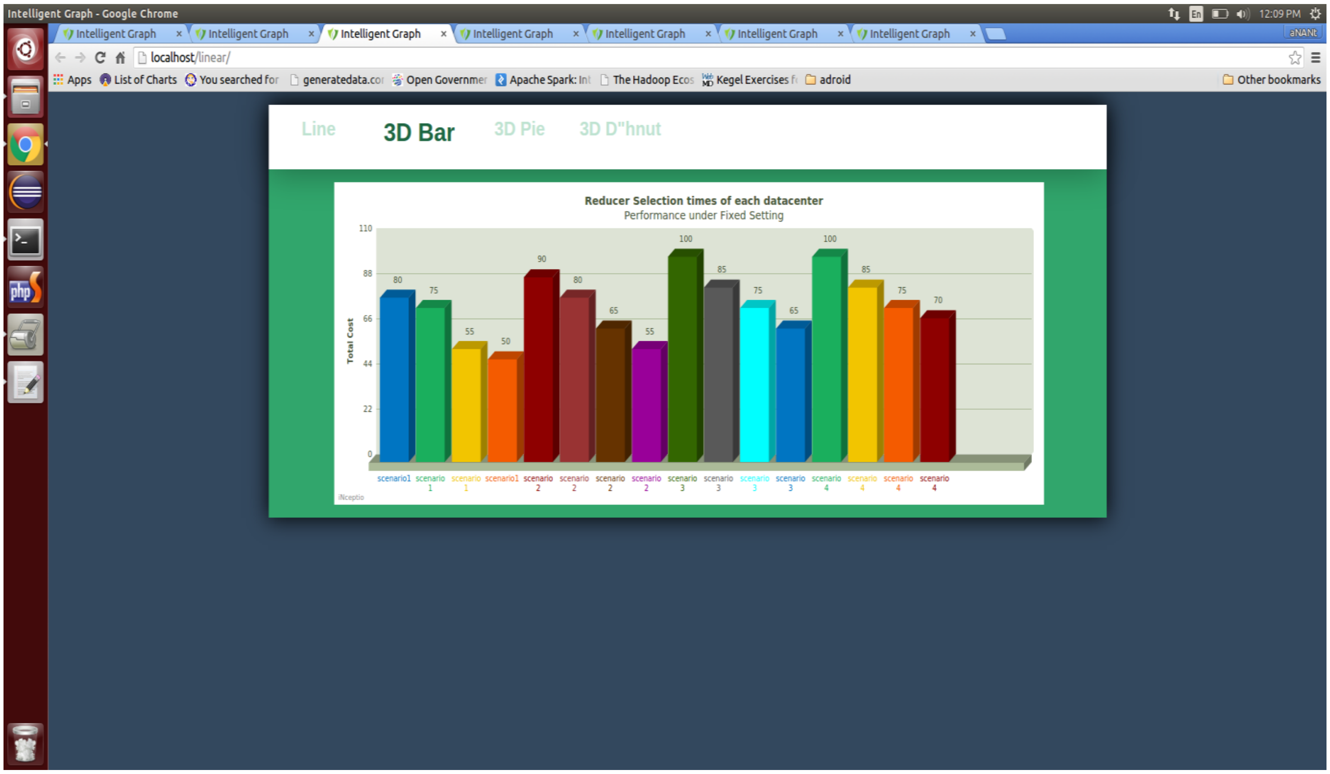


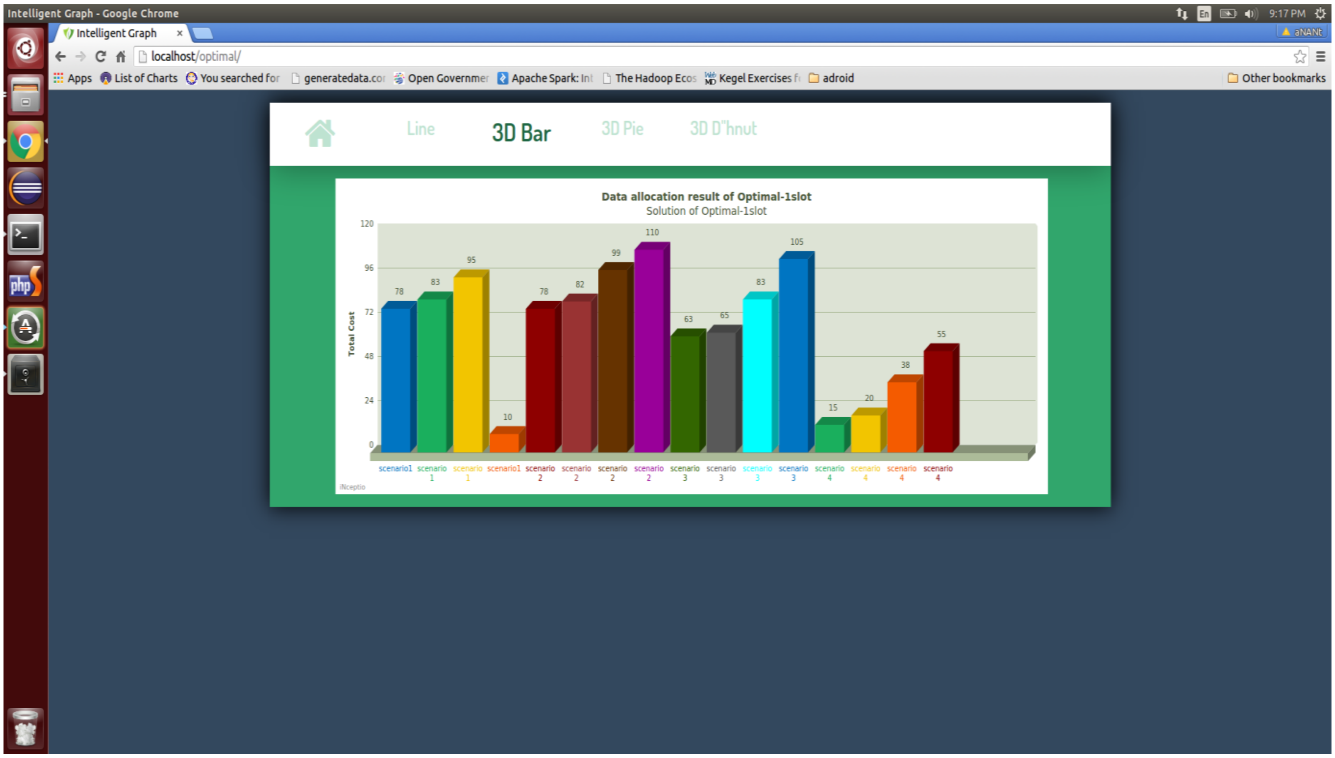
Cost-Aware Big Data Processing across Geo-distributed Datacenters

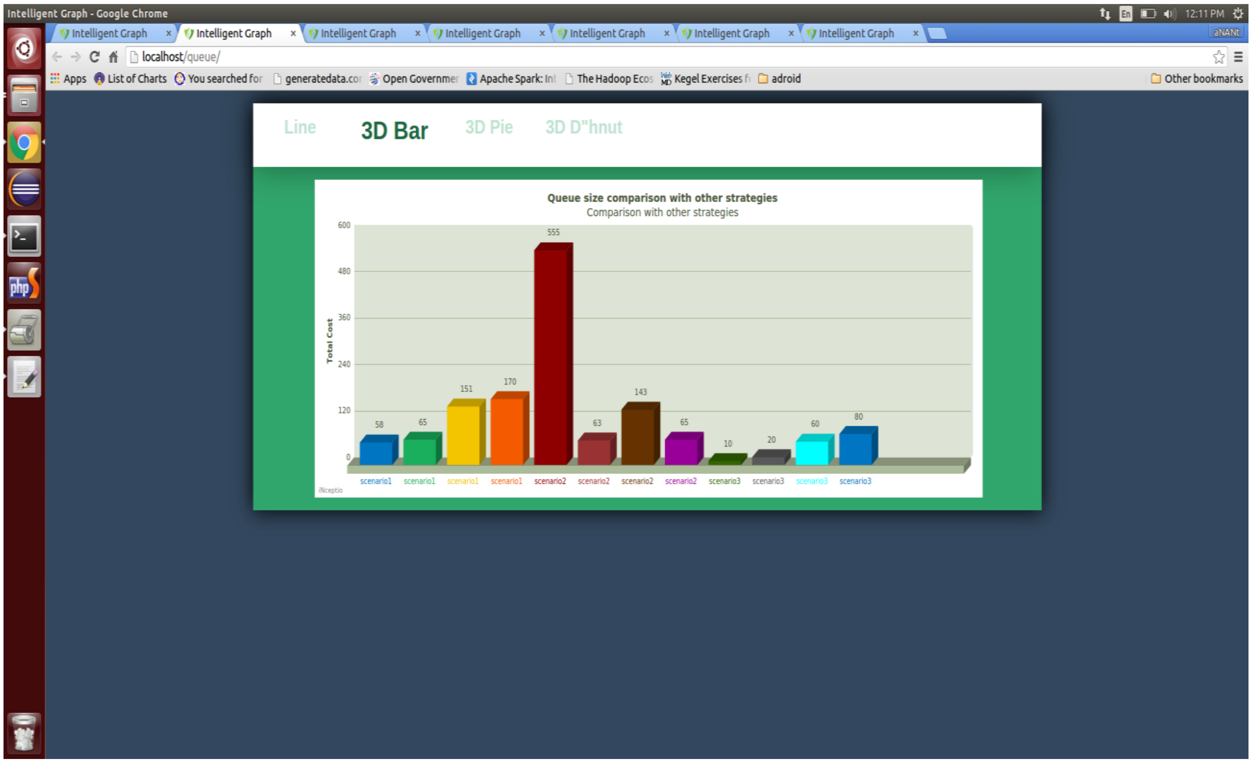


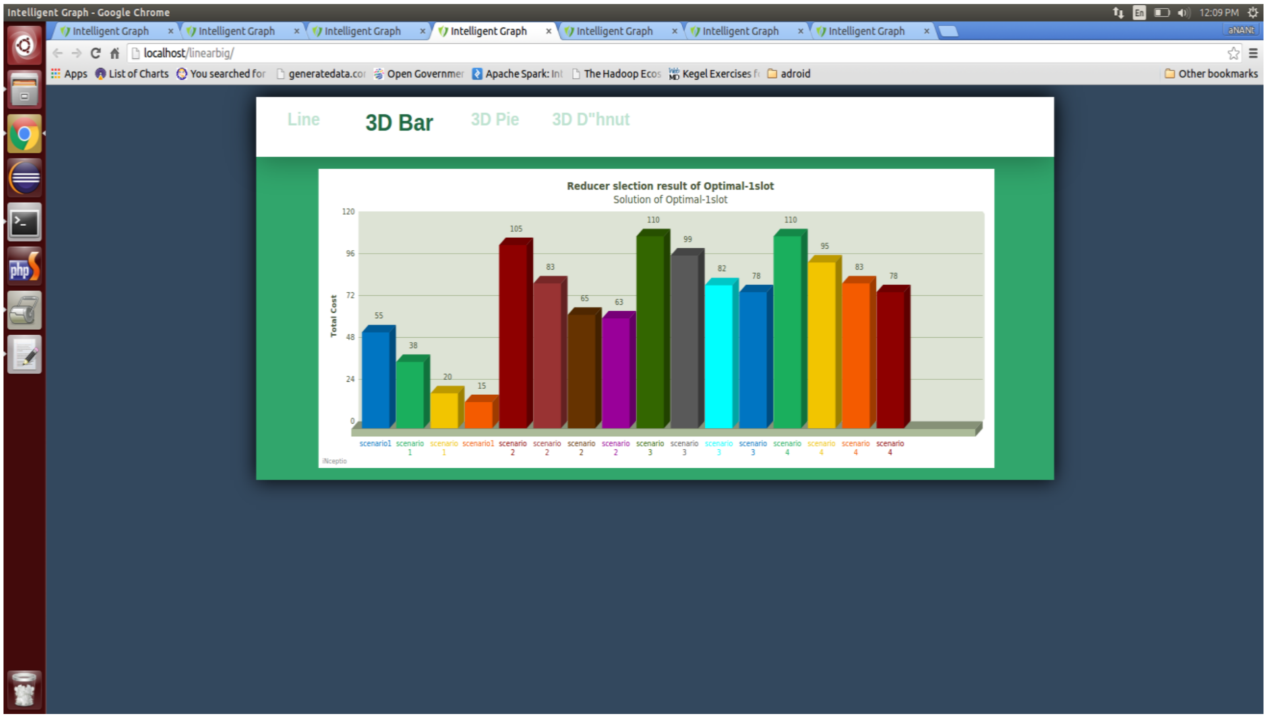

Intelligent Graph - Google Chrome
Intelligent Graph
Network Login
Inbox (588) - sukriti
localhost/join/
Apps
List of Charts
You searched for
generatedata.co
Open Governme
Apache Spark: Int
The Hadoop Ecos
Kegel Exercises f
adroid
Other bookmarks
12:38 PM
Line
3D Bar
3D Pie
3D D"hnut
Cost incurred over time slots
Statistic metric of decision λ and x
Cost
100
80
60
40
20
0
85
70
55
48
88
75
58
50
90
75
60
55
95
79
62
53
scenario1
scenario 1
scenario 1
scenario1
scenario 2
scenario 2
scenario 2
scenario 2
scenario 3
scenario 3
scenario 3
scenario 3
scenario 4
scenario 4
scenario 4
scenario 4
iNceptio


Intelligent Graph - Google Chrome
Intelligent Graph
localhost/cumulative/
Apps
List of Charts
You searched for
generatedata.co
Open Governme
Apache Spark: Int
The Hadoop Ecos
Kegel Exercises f
adroid
Other bookmarks
12:11 PM
Line
3D Bar
3D Pie
3D D"hnut
Cumulative cost comparison
Cost Comparison with Optimal
Total Cost
180
144
108
72
36
0
58
65
155
170
55
63
43
65
101
20
60
80
scenario1
scenario1
scenario1
scenario1
scenario2
scenario2
scenario2
scenario2
scenario3
scenario3
scenario3
scenario3
iNceptio


Intelligent Graph - Google Chrome
12:09 PM
Intelligent Graph
localhost/linear/
Apps
List of Charts
You searched for
generatedata.co
Open Governmer
Apache Spark: Int
The Hadoop Ecos
Kegel Exercises f
adroid
Other bookmarks
Line
3D Bar
3D Pie
3D D"hnut
Reducer Selection times of each datacenter
Performance under Fixed Setting
Total Cost
110
88
66
44
22
0
80
75
55
50
90
80
65
55
100
85
75
65
100
85
75
70
scenario1
scenario 1
scenario 1
scenario1
scenario 2
scenario 2
scenario 2
scenario 2
scenario 3
scenario 3
scenario 3
scenario 3
scenario 4
scenario 4
scenario 4
scenario 4
iNceptio


Intelligent Graph - Google Chrome
9:17 PM
Intelligent Graph
localhost/optimal/
Apps
List of Charts
You searched for
generatedata.co
Open Governmer
Apache Spark: Int
The Hadoop Ecos
Kegel Exercises f
adroid
Other bookmarks
Line
3D Bar
3D Pie
3D D"hnut
Data allocation result of Optimal-1slot
Solution of Optimal-1slot
Total Cost
120
96
72
48
24
0
78
83
95
10
78
82
99
110
63
65
83
105
15
20
38
55
scenario1
scenario 1
scenario 1
scenario1
scenario 2
scenario 2
scenario 2
scenario 2
scenario 3
scenario 3
scenario 3
scenario 3
scenario 4
scenario 4
scenario 4
scenario 4
iNceptio


Intelligent Graph - Google Chrome
12:11 PM
Intelligent Graph
localhost/queue/
Apps
List of Charts
You searched for
generatedata.com
Open Governmen
Apache Spark: Int
The Hadoop Ecos
Kegel Exercises f
adroid
Other bookmarks
Line
3D Bar
3D Pie
3D D"hnut
Queue size comparison with other strategies
Comparison with other strategies
Total Cost
600
480
360
240
120
0
58
65
151
170
555
63
143
65
10
20
60
80
scenario1
scenario1
scenario1
scenario1
scenario2
scenario2
scenario2
scenario2
scenario3
scenario3
scenario3
scenario3
iNceptio


Intelligent Graph - Google Chrome
12:09 PM
Intelligent Graph
localhost/linearbig/
Apps
List of Charts
You searched for
generatedata.com
Open Governmen
Apache Spark: Int
The Hadoop Ecos
Kegel Exercises f
adroid
Other bookmarks
Line
3D Bar
3D Pie
3D D"hnut
Reducer slection result of Optimal-1slot
Solution of Optimal-1slot
Total Cost
120
96
72
48
24
0
55
38
20
15
105
83
65
63
110
99
82
78
110
95
83
78
scenario1
scenario 1
scenario 1
scenario1
scenario 2
scenario 2
scenario 2
scenario 2
scenario 3
scenario 3
scenario 3
scenario 3
scenario 4
scenario 4
scenario 4
scenario 4
iNceptio

PAPER PUBLICATION STATUS

Publication process not yet started