KARPAGAM ACADEMY OF HIGHER EDUCATION

(Deemed to be University)

(Established Under Section 3 of UGC Act 1956)

Coimbatore – 641 021.

(For the Candidates admitted from 2016 onwards)

DEPARTMENT OF COMPUTER SCIENCE, CA & IT

SUBJECT: BIG DATA ANALYTICS
SEMESTER: V
SUB.CODE:16CAP505D **CLASS: III MCA**

SCOPE:

This scope of this course to explain the students the fundamentals of big data analytics and the methodologies used in storing, manipulating, and analyzing big data.

OBJECTIVES:

To impart to students the skills required to design scalable systems that can accept, store, and analyze large volumes of unstructured data.

UNIT-I

Fundamentals of Big Data - The Evolution of Data Management Understanding the Waves of Managing Data- Defining Big Data - Big Data Management Architecture- The Big Data Journey - Big Data Types-Defining Structured Data-Defining Unstructured Data-Putting Big Data Together.

UNIT-II

Big Data Stack- Basics of Virtualization - The importance of virtualization to big data -Server virtualization - Application virtualization - Network virtualization -Processor and memory virtualization - Data and storage virtualization-Abstraction and Virtualization-Implementing Virtualization to Work with Big Data.

UNIT-III

Hadoop - Hadoop Distributed File System - Hadoop MapReduce- The Hadoop foundation and Ecosystem.

UNIT-IV

Big Data Analytics-Text Analytics and Big Data-Customized Approaches for Analysis of Big Data

UNIT-V

Integrating Data Sources-Real-Time Data Streams and Complex Event Processing-Operationalizing Big Data.

SUGGESTED READINGS

1. Judith Hurwitz, Alan Nugent, Fern Halper, Marcia Kaufman (2013), Big Data For Dummies, Wiley India,New Delhi.
2. Paul Zikopoulos, Dirk deRoos, Krishnan Parasuraman, Thomas Deutsch, James Giles, David Corrigan (2012), Harness the Power of Big Data The IBM Big Data Platform, Tata McGraw Hill Publications, New Delhi.
3. Michael Minelli, Michele Chambers, Ambiga Dhiraj (2013). Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today's Businesses, Wiley Publications, New Delhi.
4. Zikopoulos, Paul, Chris Eaton (2011), Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data, Tata McGraw Hill Publications, New Delhi.

WEB SITES

1. www.oracle.com/BigData
2. www.planet-data.eu/sites/default/files/Big_Data_Tutorial_part4.pdf
3. www.ibm.com/developerworks/data
4. www.solacesystems.com
5. en.wikipedia.org/wiki/Big_data
6. www.sap.com/solution/big-data.html

5.

KARPAGAM ACADEMY OF HIGHER EDUCATION

(Deemed to be University)

(Established Under Section 3 of UGC Act 1956)

Coimbatore – 641 021.

(For the Candidates admitted from 2016 onwards)

DEPARTMENT OF COMPUTER SCIENCE, CA & IT

SUBJECT: BIG DATA ANALYTICS
SEMESTER: V
SUB.CODE:16CAP505D **CLASS: III MCA**

LECTURE PLAN
DEPARTMENT OF COMPUTER APPLICATIONS

S.No	Lecture Duration Period	Topics to be Covered	Support Material/Page Nos
		UNIT-I	T1:9, W1
1	1	Fundamentals of Big Data	T1:10, W1
2	1	The Evolution of Data Management	T1:11-12
3	1	Understanding the waves of Managing Data	T1:15
4	1	Defining Big Data	T1: 16
5	1	Big Data Management Architecture	T1:23-24
6	1	The Big Data Journey	T1:25-26
7	1	Big Data Types	T1:26,W1
8	1	Defining Structured Data	T1:9-30, J1
9	1	Defining Unstructured Data	T1:33, W1,J1
10	1	Putting Big Data Together	T1:9, W1
11	1	Recapitulation and Discussion of important Questions	
	Total No of Hours Planned For Unit 1=11		
		UNIT-II	
1	1	Big Data Stack	T1:48

2	1	Basics of Virtualization	T1:61- 62 , J2
3	1	The importance of Virtualization to big data	T1:63-64
4	1	Server Virtualization	T1:64
5	1	Application Virtualization	T1: 63
6	1	Network Virtualization	T1: 64
7	1	Processor and memory Virtualization	W2
8	1	Data and storage Virtualization	T1:67,J2
9	1	Abstraction and Virtualization	T1:69
10	1	Implementing Virtualization to work with Big Data	W2
11	1	Recapitulation and Discussion of important Questions	
	Total No of Hours Planned For Unit II=11		
		UNIT-III	
1	2	Hadoop	T1:111 , J3
2	2	Hadoop Distributed File System	T1:112 – 113 , J3
3	2	Hadoop Map Reduce	T1:116 , W1
4	1	The Hadoop Foundation and Ecosystem	T1: 121, W1
5	1	Recapitulation and Discussion of important Questions	
	Total No of Hours Planned For Unit III=8		
		UNIT-IV	
1	2	Big Data Analytics	T1: 141 – 142 ,J4
2	2	Text Analytics and Big Data	T1: 153 – 155
3	2	Customized Approaches for Analysis of Big Data	T1 : 167 – 169, J4
4	1	Big to Small in the goal	T1: 169 , W1
5	1	Recapitulation and Discussion of important Questions	
	Total No of Hours Planned For Unit IV=8		
		UNIT-V	

1	1	Integrating Data Sources	T1: 181 ,J5
2	2	Real time Data streams processing	T1:193 - 194
3	1	Complex Event Processing	T1: 194
4	2	Operationalizing Big Data	T1: 201 – 202,W1
5	1	Recapitulation and discussion of Important Questions	
6	1	Discussion of previous year ESE Question Paper	
7	1	Discussion of previous year ESE Question Paper	
8	1	Discussion of previous year ESE Question Paper	
	Total No of Hours Planned for unit V=10		
Total Planned Hours	**48**		

SUGGESTED READINGS

T1: Judith Hurwitz, Alan Nugent, Fern Halper, Marcia Kaufman, 2013 Big Data For Dummies, Wiley India, New Delhi.

WEBSITES

W1: en.wikipedia.org/wiki/Big.data

W2: www.oracle.com/Big data

JOURNALS

J1: "Big Data: Study in Structured and Unstructured Data", Motashim Rasool and Wasim Khan ,IJTIR, Vol 14, April 2015.

J2: "An Importance of using Virtualization Technology in Cloud Computing", Rakesh Kumar, Shilpi charu, IJCT , Vol 2, No 2, Feb 2015.

J3: "A Review paper on Big Data and Hadoop", Harshawardhan S.Bhosale, Prof. Devendra Godekar, IJSRP, Vol 4, Issue 10, Oct 2014.

J4: "Big Data Analytics: A Literature Review paper", Nada Elgendy and Ahmed Elargal, Springer International , Publishing,Switzerland,2014.

J5: "A Review on Big Data Integration", B.Arputhamary, L.Arockiam, IJCA ,2014.

UNIT – I

SYLLABUS

Fundamentals of Big Data – Evolution of Data Management – Understanding the waves of Managing Data – Defining Big Data – Big Data Management Architecture – The Big Data Journey – Big Data Types – Defining Structured Data – Defining Unstructured Data – Putting Big Data together

FUNDAMENTALS OF BIG DATA

- Managing and analyzing data have always offered the greatest benefits and the greatest challenges for organizations of all sizes and across all industries. Businesses have long struggled with finding a pragmatic approach to capturing information about their customers, products, and services. When a company only had a handful of customers who all bought the same product in the same way, things were pretty straightforward and simple. But over time, companies and the markets they participate in have grown more complicated.
- To survive or gain a competitive advantage with customers, these companies added more product lines and diversified how they deliver their product. Data struggles are not limited to business. Research and development (R&D) organizations, for example, have struggled to get enough com- putting power to run sophisticated models or to process images and other sources of scientific data.
- Indeed, we are dealing with a lot of complexity when it comes to data. Some data is structured and stored in a traditional relational database, while other data, including documents, customer service records, and even pictures and videos, is unstructured. Companies also have to consider new sources of data generated by machines such as sensors. Other new information sources are human generated, such as data from social media and the click-stream data generated from website interactions. In addition, the availability and adoption of newer, more powerful mobile devices, coupled with ubiquitous access to global networks will drive the creation of new sources for data.

- Although each data source can be independently managed and searched, the challenge today is how companies can make sense of the intersection of all these different types of data. When you are dealing with so much information in so many different forms, it is impossible to think about data management in traditional ways. Although we have always had a lot of data, the difference today is that significantly more of it exists, and it varies in type and timeliness. Organizations are also finding more ways to make use of this information than ever before. Therefore, you have to think about managing data differently. That is the opportunity and challenge of big data. In this chapter, we provide you a context for what the evolution of the movement to big data is all about and what it means to your organization.

THE EVOLUTION OF DATA MANAGEMENT

- Big data is defined as any kind of data source that has at least three shared characteristics:
  - Extremely large *Volumes* of data
  - Extremely high *Velocity* of data
  - Extremely wide *Variety* of data
- Big data is important because it enables organizations to gather, store, manage, and manipulate vast amounts data at the right speed, at the right time, to gain the right insights. But before we delve into the details of big data, it is important to look at the evolution of data management and how it has led to big data. Big data is not a stand-alone technology; rather, it is a combination of the last 50 years of technology evolution.

UNDERSTANDING THE WAVES OF MANAGING DATA

- Each data management wave is born out of the necessity to try and solve a specific type of data management problem. Each of these waves or phases evolved because of cause and effect. When a new technology solution came to market, it required the discovery of new approaches. When the relational database came to market, it needed a set of tools to allow managers to study the relationship between data elements. When companies started storing unstructured data, analysts needed new capabilities such as natural language–

based analysis tools to gain insights that would be useful to business. If you were a search engine company leader, you began to realize that you had access to immense amounts of data that could be monetized.

- The data management waves over the past five decades have culminated in where we are today: the initiation of the big data era. So, to understand big data, you have to understand the underpinning of these previous waves. You also need to understand that as we move from one wave to another, we don't throw away the tools and technology and practices that we have been using to address a different set of problems.
- **Wave 1: Creating manageable data structures**
  - As computing moved into the commercial market in the late 1960s, data was stored in flat files that imposed no structure. When companies needed to get to a level of detailed understanding about customers, they had to apply brute-force methods, including very detailed programming models to create some value. Later in the 1970s, things changed with the invention of the relational data model and the relational database management system (RDBMS) that imposed structure and a method for improving performance. Most importantly, the relational model added a level of abstraction (the structured query language [SQL], report generators, and data management tools) so that it was easier for programmers to satisfy the growing business demands to extract value from data.
  - The relational model offered an ecosystem of tools from a large number of emerging software companies. It filled a growing need to help companies better organize their data and be able to compare transactions from one geography to another. In addition, it helped business managers who wanted to be able to examine information such as inventory and compare it to customer order information for decision-making purposes. But a problem emerged from this exploding demand for answers: Storing this growing volume of data was expensive and accessing it was slow. Making matters worse, lots of data duplication existed, and the actual business value of that data was hard to measure.
  - At this stage, an urgent need existed to find a new set of technologies to

support the relational model. The Entity-Relationship (ER) model emerged, which added additional abstraction to increase the usability of the data. In this model, each item was defined independently of its use. Therefore, developers could create new relationships between data sources without complex programming. It was a huge advance at the time, and it enabled developers to push the boundaries of the technology and create more complex models requiring complex techniques for joining entities together. The market for relational databases exploded and remains vibrant today. It is especially important for transactional data management of highly structured data.

- When the volume of data that organizations needed to manage grew out of control, the data warehouse provided a solution. The data warehouse enabled the IT organization to select a subset of the data being stored so that it would be easier for the business to try to gain insights. The data ware- house was intended to help companies deal with increasingly large amounts of structured data that they needed to be able to analyze by reducing the volume of the data to something smaller and more focused on a particular area of the business. It filled the need to separate operational decision support processing and decision support — for performance reasons. In addition, warehouses often store data from prior years for understanding organizational performance, identifying trends, and helping to expose pat- terns of behavior. It also provided an integrated source of information from across various data sources that could be used for analysis. Data warehouses were commercialized in the 1990s, and today, both content management systems and data warehouses are able to take advantage of improvements in scalability of hardware, virtualization technologies, and the ability to create integrated hardware and software systems, also known as appliances.
- Sometimes these data warehouses themselves were too complex and large and didn’t offer the speed and agility that the business required. The answer was a

further refinement of the data being managed through data marts. These data marts were focused on specific business issues and were much more streamlined and supported the business need for speedy queries than the more massive data warehouses. Like any wave of data management, the warehouse has evolved to support emerging technologies such as integrated systems and data appliances.

- Data warehouses and data marts solved many problems for companies needing a consistent way to manage massive transactional data. But when it came to managing huge volumes of unstructured or semi-structured data, the warehouse was not able to evolve enough to meet changing demands. To com- plicate matters, data warehouses are typically fed in batch intervals, usually weekly or daily. This is fine for planning, financial reporting, and traditional marketing campaigns, but is too slow for increasingly real-time business and consumer environments.
- How would companies be able to transform their traditional data management approaches to handle the expanding volume of unstructured data elements? The solution did not emerge overnight. As companies began to store unstructured data, vendors began to add capabilities such as BLOBs (binary large objects). In essence, an unstructured data element would be stored in a relational database as one contiguous chunk of data. This object could be labeled (that is, a customer inquiry) but you couldn't see what was inside that object. Clearly, this wasn't going to solve changing customer or business needs.
- Enter the object database management system (ODBMS). The object data- base stored the BLOB as an addressable set of pieces so that we could see what was in there. Unlike the BLOB, which was an independent unit appended to a traditional relational database, the object database provided a unified approach for dealing with unstructured data? Object databases include a programming language and a structure for the data elements so that it is easier to manipulate various data objects without programming and complex joins. The object databases introduced a new level of innovation that helped lead to the second wave of data management.

- **Wave 2: Web and content management**
  - It's no secret that most data available in the world today is unstructured. Paradoxically, companies have focused their investments in the systems with structured data that were most closely associated with revenue: line- of-business transactional systems. Enterprise Content Management systems evolved in the 1980s to provide businesses with the capability to better manage unstructured data, mostly documents. In the 1990s with the rise of the web, organizations wanted to move beyond documents and store and manage web content, images, audio, and video.
  - The market evolved from a set of disconnected solutions to a more unified model that brought together these elements into a platform that incorporated business process management, version control, information recognition, text management, and collaboration. This new generation of systems added meta- data (information about the organization and characteristics of the information). These solutions remain incredibly important for companies needing to manage all this data in a logical manner. But at same time, a new generation of requirements has begun to emerge that drive us to the next wave. These new requirements have been driven, in large part, by a convergence of factors including the web, virtualization, and cloud computing. In this new wave, organizations are beginning to understand that they need to manage a new generation of data sources with an unprecedented amount and variety of data that needs to be processed at an unheard-of speed.
- **Wave 3: Managing big data**
  - Is big data really new or is it an evolution in the data management journey? The answer is yes it is actually both. As with other waves in data manage-ment, big data is built on top of the evolution of data management practices over the past five decades. What is new is that for the first time, the cost f computing cycles and storage has reached a tipping point. Why is this important? Only a few years ago, organizations typically would compromise

by storing snapshots or subsets of important information because the cost of storage and processing limitations prohibited them from storing everything they wanted to analyze.

- In many situations, this compromise worked fine. For example, a manufacturing company might have collected machine data every two minutes to deter- mine the health of systems. However, there could be situations where the snapshot would not contain information about a new type of defect and that might go unnoticed for months.
- With big data, it is now possible to virtualize data so that it can be stored efficiently and, utilizing cloud-based storage, more cost-effectively as well. In addition, improvements in network speed and reliability have removed other physical limitations of being able to manage massive amounts of data at an acceptable pace. Add to this the impact of changes in the price and sophistication of computer memory. With all these technology transitions, it is now possible to imagine ways that companies can leverage data that would have been inconceivable only five years ago.
- But no technology transition happens in isolation; it happens when an important need exists that can be met by the availability and maturation of technology. Many of the technologies at the heart of big data, such as virtualization, parallel processing, distributed file systems, and in-memory databases, have been around for decades. Advanced analytics have also been around for decades, although they have not always been practical. Other technologies such as Hadoop and Map Reduce have been on the scene for only a few years. This combination of technology advances can now address significant business problems. Businesses want to be able to gain insights and actionable results from many different kinds of data at the right speed no matter how much data is involved.
- If companies can analyze petabytes of data (equivalent to 20 million four-drawer file cabinets filled with text files or 13.3 years of HDTV content) with acceptable performance to discern patterns and anomalies,

businesses can begin to make sense of data in new ways. The move to big data is not just about businesses. Science, research, and government activities have also helped to drive it forward. Just think about analyzing the human genome or dealing with all the astronomical data collected at observatories to advance our understanding of the world around us. Consider the amount of data the government collects in its antiterrorist activities as well, and you get the idea that big data is not just about business.

- Different approaches to handling data exist based on whether it is data in motion or data at rest. Here's a quick example of each. Data in motion would be used if a company is able to analyze the quality of its products during the manufacturing process to avoid costly errors. Data at rest would be used by a business analyst to better understand customers' current buying patterns based on all aspects of the customer relationship, including sales, social media data, and customer service interactions.
- Keep in mind that we are still at an early stage of leveraging huge volumes of data to gain a 360-degree view of the business and anticipate shifts and changes in customer expectations. The technologies required to get the answers the business needs are still isolated from each other. To get to the desired end state, the technologies from all three waves will have to come together. As you will see as you read this book, big data is not simply about one tool or one technology. It is about how all these technologies come together to give the right insights, at the right time, based on the right data — whether it is generated by people, machines, or the web

DEFINING BIG DATA

- Big data is not a single technology but a combination of old and new technologies that helps companies gain actionable insight. Therefore, big data is the capability to manage a huge volume of disparate data, at the right speed, and within the right time frame to allow real-time analysis and reaction. Big data is typically broken down by three characteristics:

- **Volume:** How much data
- **Velocity:** How fast that data is processed
- **Variety:** The various types of data

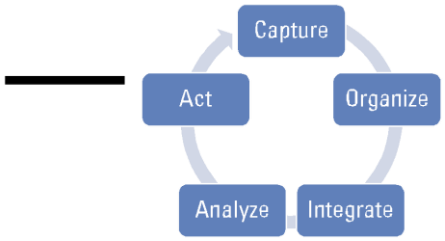
- Although it's convenient to simplify big data into the three Vs, it can be misleading and overly simplistic. For example, you may be managing a relatively small amount of very disparate, complex data or you may be processing a huge volume of very simple data. That simple data may be all structured or all unstructured. Even more important is the fourth V: veracity. How accurate is that data in predicting business value? Do the results of a big data analysis actually make sense?
- It is critical that you don't underestimate the task at hand. Data must be able to be verified based on both accuracy and context. An innovative business may want to be able to analyze massive amounts of data in real time to quickly assess the value of that customer and the potential to provide additional offers to that customer. It is necessary to identify the right amount and types of data that can be analyzed to impact business outcomes. Big data incorporates all data, including structured data and unstructured data from e-mail, social media, text streams, and more. This kind of data management requires that companies leverage both their structured and unstructured data.

BUILDING A SUCCESSFUL BIG DATA MANAGEMENT ARCHITECTURE

- We have moved from an era where an organization could implement a data- base to meet a specific project need and be done. But as data has become the fuel of growth and innovation, it is more important than ever to have an underlying architecture to support growing requirements.
- **Beginning with capture, organize, integrate, analyze, and act**

- Before we delve into the architecture, it is important to take into account the functional requirements for big data. Figure 1-1 illustrates that data must first be captured, and then organized and integrated. After this phase is successfully implemented, data can be analyzed based on the problem being addressed. Finally, management takes action based on the outcome of that analysis. For example, Amazon.com might recommend a book based on a

past purchase or a customer might receive a coupon for a discount for a future purchase of a related product to one that was just purchased.



- Although this sounds straightforward, certain nuances of these functions are complicated. Validation is a particularly important issue. If your organization is combining data sources, it is critical that you have the ability to validate that these sources make sense when combined. Also, certain data sources may contain sensitive information, so you must implement sufficient levels of security and governance.

THE BIG DATA JOURNEY

- Companies have always had to deal with lots of data in lots of forms. The change that big data brings is what you can do with that information. If you have the right technology in place, you can use big data to anticipate and solve business problems and react to opportunities. With big data, you can analyze data patterns to change everything, from the way you manage cities, prevent failures, conduct experiments, manage traffic, improve customer satisfaction, or enhance product quality, just to name a few examples. The emerging technologies and tools that are the heart of this book can help you understand and unleash the tremendous power of big data, changing the world as we know it

BIG DATA TYPES

- The two main types of data that make up big data — structured and unstructured

DEFINING STRUCTURED DATA

- The term structured data generally refers to data that has a defined length and format. Examples of structured data include numbers, dates, and groups of words and

numbers called strings (for example, a customer's name, address, and so on). Most experts agree that this kind of data accounts for about 20 percent of the data that is out there. Structured data is the data that you're probably used to dealing with. It's usually stored in a database. You can query it using a language like structured query language (SQL), which we discuss later in the "Defining Unstructured Data" section.

- Your company may already be collecting structured data from "traditional" sources. These might include your customer relationship management (CRM) data, operational enterprise resource planning (ERP) data, and financial data. Often these data elements are integrated in a data warehouse for analysis.

Exploring sources of big structured data

- Although this might seem like business as usual, in reality, structured data is taking on a new role in the world of big data. The evolution of technology provides newer sources of structured data being produced — often in real time and in large volumes. The sources of data are divided into two categories:
- **Computer- or machine-generated:** Machine-generated data generally refers to data that is created by a machine without human intervention.
- **Human-generated:** This is data that humans, in interaction with computers, supply.
- Some experts argue that a third category exists that is a hybrid between machine and human. Here though, we're concerned with the first two categories.
- Machine-generated structured data can include the following:
- **Sensor data:** Examples include radio frequency ID (RFID) tags, smart meters, medical devices, and Global Positioning System (GPS) data. For example, RFID is rapidly becoming a popular technology. It uses tiny computer chips to track items at a distance. An example of this is tracking containers of produce from one location to another. When information is transmitted from the receiver, it can go into a server and then be analyzed. Companies are interested in this for supply chain management and inventory control. Another example of sensor data is smart phones that contain sensors like GPS that can be used to understand customer behavior in new ways.
- **Web log data:** When servers, applications, networks, and so on operate, they capture

all kinds of data about their activity. This can amount to huge volumes of data that can be useful, for example, to deal with service-level agreements or to predict security breaches.

- **Point-of-sale data:** When the cashier swipes the bar code of any product that you are purchasing, all that data associated with the product is generated. Just think of all the products across all the people who purchase them and you can understand how big this data set can be.
- **Financial data:** Lots of financial systems are now programmatic; they are operated based on predefined rules that automate processes. Stock- trading data is a good example of this. It contains structured data such as the company symbol and dollar value. Some of this data is machine generated, and some is human generated.
- Examples of structured human-generated data might include the following:
- **Input data:** This is any piece of data that a human might input into a computer, such as name, age, income, non-free-form survey responses, and so on. This data can be useful to understand basic customer behavior.
- **Click-stream data:** Data is generated every time you click a link on a website. This data can be analyzed to determine customer behavior and buying patterns.
- **Gaming-related data:** Every move you make in a game can be recorded. This can be useful in understanding how end users move through a gaming portfolio.

DEFINING UNSTRUCTURED DATA

- Unstructured data is data that does not follow a specified format. If 20 percent of the data available to enterprises is structured data, the other 80 percent is unstructured. Unstructured data is really most of the data that you will encounter. Until recently, however, the technology didn't really support doing much with it except storing it or analyzing it manually.

Exploring Sources of Unstructured Data

- Unstructured data is everywhere. In fact, most individuals and organizations conduct their lives around unstructured data. Just as with

structured data, unstructured data is either machine generated or human generated.

- Here are some examples of machine-generated unstructured data:
- **Satellite images:** This includes weather data or the data that the government captures in its satellite surveillance imagery. Just think about Google Earth, and you get the picture (pun intended).
- **Scientific data:** This includes seismic imagery, atmospheric data, and high energy physics.
- **Photographs and video:** This includes security, surveillance, and traffic video.
- **Radar or sonar data:** This includes vehicular, meteorological, and oceanographic seismic profiles.
- The following list shows a few examples of human-generated unstructured data:
- **Text internal to your company:** Think of all the text within documents, logs, survey results, and e-mails. Enterprise information actually represents a large percent of the text information in the world today.
- **Social media data:** This data is generated from the social media plat- forms such as YouTube, Facebook, Twitter, LinkedIn, and Flicker.
- **Mobile data:** This includes data such as text messages and location information.
- **Website content:** This comes from any site delivering unstructured content, like YouTube, Flickr, or Instagram.

PUTTING BIG DATA TOGETHER

- What you want to do with your structured and unstructured data indicates why you might choose one piece of technology over another one. It also determines the need to understand inbound data structures to put this data in the right place.

Managing different data types

Figure 2-2 shows a helpful table that outlines some of the characteristics of big data and the types of data management systems you might want to use to address

each one. We don't expect you to know what these are yet; they are described in the chapters that follow.

POSSIBLE QUESTIONS

UNIT – I

PART – A (20 MARKS)

(Q.NO 1 TO 20 Online Examinations)

PART – B (6 MARKS)

1. Explain about the fundamentals of Big data
2. Discuss in detail about understanding the waves of managing data.
3. Explain about structured data.
4. Discuss in detail about big data management architecture.
5. Explain about unstructured data.
6. Explain in detail about the defining big data.
7. Explain about the Real – time and Non – time requirements.
8. Explain in detail about the big data journey
9. Explain in detail about the putting big data together.
10. Explain about the Web and content management.

PART – C (10 MARKS)

1. Explain how to manage the big data
2. Explain about how to make big data as an operational business
3. Explain the steps involved for exploring sources of Structured Data
4. Explain the steps involved for exploring sources of UnStructured Data
5. Explain the evolution of Data management

UNIT – II

SYLLABUS

Big Data Stack – Basics of Virtualization – The Importance of Virtualization to Big Data – Server Virtualization – Application Virtualization – Network Virtualization – Processor and Memory Virtualization – Data and Storage Virtualization – Abstraction and Virtualization – Implementing Virtualization to work with Big data

VIRTUALIZATION

- Virtualization is a foundational technology applicable to the implementation of both cloud computing and big data. It provides the basis for many of the platform attributes required to access, store, analyze, and manage the distributed computing components in big data environments. Virtualization — the process of using computer resources to imitate other resources — is valued for its capability to increase IT resource utilization, efficiency, and scalability.
- One primary application of virtualization is server consolidation, which helps organizations increase the utilization of physical servers and potentially save on infrastructure costs. However, you find many benefits to virtualization. Companies that initially focused solely on server virtualization are now recognizing that it can be applied across the entire IT infrastructure, including software, storage, and networks.

UNDERSTANDING THE BASICS OF VIRTUALIZATION

- Virtualization separates resources and services from the underlying physical delivery environment, enabling you to create many virtual systems within a single physical system. Figure 5-1 shows a typical virtualization environment.
- One of the primary reasons that companies have implemented virtualization is to improve the performance and efficiency of processing of a diverse mix of workloads. Rather than assigning a dedicated set of physical resources to each set of tasks, a pooled set of virtual resources can be quickly allocated as needed across all workloads. Reliance on the pool

of virtual resources allows companies to improve latency. This increase in service delivery speed and efficiency is a function of the distributed nature of virtualized environments and helps to improve overall time-to-value.

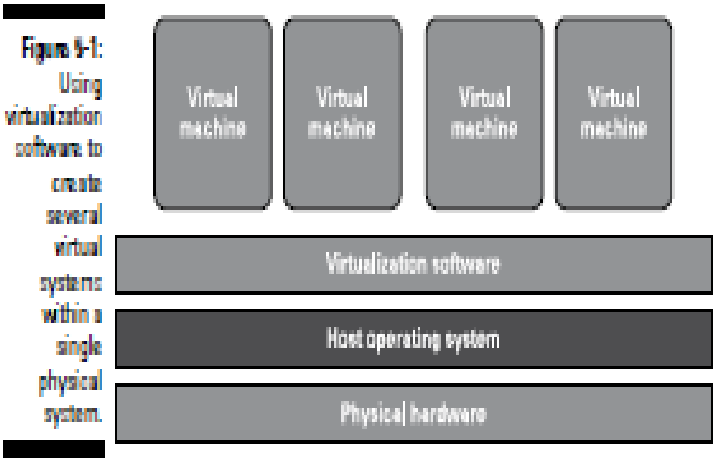


Figure 5-1: Using virtualization software to create several virtual systems within a single physical system.

- Using a distributed set of physical resources, such as servers, in a more flexible and efficient way delivers significant benefits in terms of cost savings and improvements in productivity. The practice has several benefits, including the following:
  - Virtualization of physical resources (such as servers, storage, and networks) enables substantial improvement in the utilization of these resources.
  - Virtualization enables improved control over the usage and performance of your IT resources.
  - Virtualization can provide a level of automation and standardization to optimize your computing environment.
  - Virtualization provides a foundation for cloud computing.
- Solving big data challenges typically requires the management of large volumes of highly distributed data stores along with the use of compute- and data-intensive applications. Therefore, you need a highly efficient IT environ- ment to support big data. Virtualization provides the added level of efficiency to make big data platforms a reality.

Although virtualization is technically not a requirement for big data analysis, software frameworks such as Map Reduce, which are used in big data environments, are more efficient in a virtualized environment.

- Virtualization has three characteristics that support the scalability and operating efficiency required for big data environments:

- **Partitioning:** In virtualization, many applications and operating systems are supported in a single physical system by partitioning (separating) the available resources.
- **Isolation:** Each virtual machine is isolated from its host physical system and other virtualized machines. Because of this isolation, if one virtual instance crashes, the other virtual machines and the host system aren't affected. In addition, data isn't shared between one virtual instance and another.
- **Encapsulation:** A virtual machine can be represented (and even stored) as a single file, so you can identify it easily based on the services it pro- vides. For example, the file containing the encapsulated process could be a complete business service. This encapsulated virtual machine could be presented to an application as a complete entity. Thus, encapsulation could protect each application so that it doesn't interfere with another application.

SERVER VIRTUALIZATION

- In server virtualization, one physical server is partitioned into multiple virtual servers. The hardware and resources of a machine — including the random access memory (RAM), CPU, hard drive, and network controller — can be virtualized (logically split) into a series of virtual machines that each runs its own applications and operating system. A virtual machine (VM) is a software representation of a physical machine that can execute or perform the same functions as the physical machine. A thin layer of software is actually inserted into the hardware that contains a virtual machine monitor, or hypervisor. The hypervisor can be thought of as the technology that manages traffic between the VMs and the physical machine.

- Server virtualization uses the hypervisor to provide efficiency in the use f physical resources. Of course, installation, configuration, and administrative tasks are associated with setting up these virtual machines. This includes license management, network management, and workload administration, as well as capacity planning.
- Server virtualization helps to ensure that your platform can scale as needed to handle the large volumes and varied types of data included in your big data analysis.
- In addition, server virtualization provides the foundation that enables many of the cloud services used as data sources in a big data analysis. Virtualization increases the efficiency of the cloud that makes many complex systems easier to optimize. As a result, organizations have the performance and optimization to be able to access data that was previously either unavailable or very hard to collect. Big data platforms are increasingly used as sources of enormous amounts of data about customer preferences, sentiment, and behaviors. Companies can integrate this information with internal sales and product data to gain insight into customer preferences to make more targeted and personalized offers.

APPLICATION VIRTUALIZATION

- Application infrastructure virtualization provides an efficient way to manage applications in context with customer demand. The application is encapsulated in a way that removes its dependencies from the underlying physical computer system. This helps to improve the overall manageability and portability of the application. In addition, the application infrastructure virtualization software typically allows for codifying business and technical usage policies to make sure that each of your applications leverages virtual and physical resources in a predictable way. Efficiencies are gained because you can more easily distribute IT resources according to the relative business value of your applications. In

other words, your most critical applications can receive top priority to draw from pools of available computing and storage capacity as needed.

- Application infrastructure virtualization used in combination with server virtualization can help to ensure that business service-level agreements (SLAs) are met. Server virtualization monitors CPU and memory usage, but does not account for variations in business priority when allocating resources. For example, you might require that all applications are treated with the same business-level priority. By implementing application infrastructure virtualization in addition to server virtualization, you can ensure that the most high- priority applications have top-priority access to resources.

NETWORK VIRTUALIZATION

- Network virtualization — software-defined networking — provides an efficient way to use networking as a pool of connection resources. Networks are virtualized in a similar fashion to other physical technologies. Instead of relying on the physical network for managing traffic between connections, you can create multiple virtual networks all utilizing the same physical implementation. This can be useful if you need to define a network for data gathering with a certain set of performance characteristics and capacity and another network for applications with different performance and capacity. Limitations in the network layer can lead to bottlenecks that lead to unacceptable latencies in big data environments. Virtualizing the network helps reduce these bottlenecks and improve the capability to manage the large distributed data required for big data analysis.

PROCESSOR AND MEMORY VIRTUALIZATION

- Processor virtualization helps to optimize the processor and maximize performance. Memory virtualization decouples memory from the servers.
- In big data analysis, you may have repeated queries of large data sets and the creation of advanced analytic algorithms, all designed to look for pat- terns and trends that are not yet understood. These advanced analytics can require lots of

processing power (CPU) and memory (RAM). For some of these computations, it can take a long time without sufficient CPU and memory resources. Processor and memory virtualization can help speed the processing and get your analysis results sooner.

DATA AND STORAGE VIRTUALIZATION

- Data virtualization can be used to create a platform for dynamic linked data services. This allows data to be easily searched and linked through a unified reference source. As a result, data virtualization provides an abstract service that delivers data in a consistent form regardless of the underlying physical database. In addition, data virtualization exposes cached data to all applications to improve performance.
- Storage virtualization combines physical storage resources so that they are more effectively shared. This reduces the cost of storage and makes it easier to manage data stores required for big data analysis.
- Data and storage virtualization play a significant role in making it easier and less costly to store, retrieve, and analyze the large volumes of fast and varying types of data. Remember that some big data may be unstructured and not easily stored using traditional methods. Storage makes it easier to store large and unstructured data types. In a big data environment, it is advantageous to have access to a variety of operational data stores on demand. For example, you may only need access to a columnar database infrequently. With virtualization, the database can be stored as a virtual image and invoked whenever it is needed without consuming valuable data center resources or capacity.

ABSTRACTION AND VIRTUALIZATION

- For IT resources and services to be virtualized, they are separated from the underlying physical delivery environment. The technical term for this act of separation is called abstraction. Abstraction is a key concept in big data. Map Reduce and Hadoop are distributed computing environments where everything is abstracted. The detail is abstracted out so that the developer or analyst does not need to be concerned with where the data elements are actually located.

- Abstraction minimizes the complexity of something by hiding the details and providing only the relevant information. For example, if you were going to pick up someone whom you've never met before, he might tell you the location to meet him, how tall he is, his hair color, and what he will be wearing. He doesn't need to tell you where he was born, how much money he has in the bank, his birth date, and so on. That's the idea with abstraction — it's about providing a high-level specification rather than going into lots of detail about how something works. In the cloud, for instance, in an Infrastructure as a Service (IaaS) delivery model, the details of the physical and virtual infra-structure are abstracted from the user.

IMPLEMENTING VIRTUALIZATION TO WORK WITH BIG DATA

- Virtualization helps makes your IT environment smart enough to handle big data analysis. By optimizing all elements of your infrastructure, including hardware, software, and storage, you gain the efficiency needed to process and manage large volumes of structured and unstructured data. With big data, you need to access, manage, and analyze structured and unstructured data in a distributed environment.
- Big data assumes distribution. In practice, any kind of Map Reduce will work better in a virtualized environment. You need the capability to move workloads around based on requirements for compute power and storage.
- Virtualization will enable you to tackle larger problems that have not yet been scoped. You may not know in advance how quickly you will need to scale.
- Virtualization will enable you to support a variety of operational big data stores. For example, a graph database can be spun up as an image.
- The most direct benefit from virtualization is to ensure that Map Reduce engines work better. Virtualization will result in better scale and performance for Map Reduce. Each one of the Map and Reduce tasks needs to be executed

independently. If the Map Reduce engine is parallelized and configured to run in a virtual environment, you can reduce management overhead and allow for expansions and contractions in the task workloads. Map Reduce itself is inherently parallel and distributed. By encapsulating the Map Reduce engine in a virtual container, you can run what you need whenever you need it. With virtualization, you increase your utilization of the assets you have already paid for by turning them into generic pools of resource

POSSIBLE QUESTIONS

UNIT – II

PART – A (20 MARKS)

(Q.NO 1 TO 20 Online Examinations)

PART – B (6 MARKS)

1. Explain about the big data stack.
2. Discuss in detail about the big data analytics and applications.
3. Explain about basics of virtualization.
4. Explain in detail about the importance of virtualization to big data
5. Explain about Server virtualization.
6. Explain about Application virtualization.
7. Explain about the Network virtualization.
8. Discuss in detail about the processor and memory virtualization
9. Explain about the data storage virtualization.
10. Discuss in detail about the abstract and virtualization.

PART – C (10 MARKS)

(Compulsory Question)

1. Explain the process of Data Storage Virtualization
2. Explain how virtualization is implemented to work with big data
3. Comparison between Processor and Memory Virtualization
4. Difference between Network Virtualization and Server Virtualization
5. Explain the benefits and Features of Virtualization

UNIT – III

SYLLABUS

Hadoop – Hadoop Distributed File System – Hadoop Map Reduce – The Hadoop Foundation Ecosystem

HADOOP

- Hadoop was originally built by a Yahoo! engineer named Doug Cutting and is now an open source project managed by the Apache Software Foundation. It is made available under the Apache License v2.0.
- Hadoop is a fundamental building block in our desire to capture and process big data. Hadoop is designed to parallelize data processing across computing nodes to speed computations and hide latency. At its core, Hadoop has two primary components:
- **Hadoop Distributed File System:** A reliable, high-bandwidth, low-cost, data storage cluster that facilitates the management of related files across machines.
- **Map Reduce engine:** A high-performance parallel/distributed data- processing implementation of the Map Reduce algorithm. Hadoop is designed to process huge amounts of structured and unstructured data (terabytes to petabytes) and is implemented on racks of commodity servers as a Hadoop cluster. Servers can be added or removed from the cluster dynamically because Hadoop is designed to be "self-healing." In other words, Hadoop is able to detect changes, including failures, and adjust to those changes and continue to operate without interruption.

Understanding the Hadoop Distributed File System (HDFS)

- The Hadoop Distributed File System is a versatile, resilient, clustered approach to managing files in a big data environment. HDFS is not the final destination for files. Rather, it is a data service that offers a unique set of capabilities needed when data volumes and velocity are high. Because the data is written once and then read many times thereafter, rather than the constant read-writes of other file systems, HDFS is an excellent choice for supporting big data analysis. The service includes a "Name Node"

and multiple "data nodes" running on a commodity hardware cluster and provides the highest levels of performance when the entire cluster is in the same physical ack in the data center. In essence, the Name Node keeps track of where data is physically stored. Figure 9-1 depicts the basic architecture of HDFS.

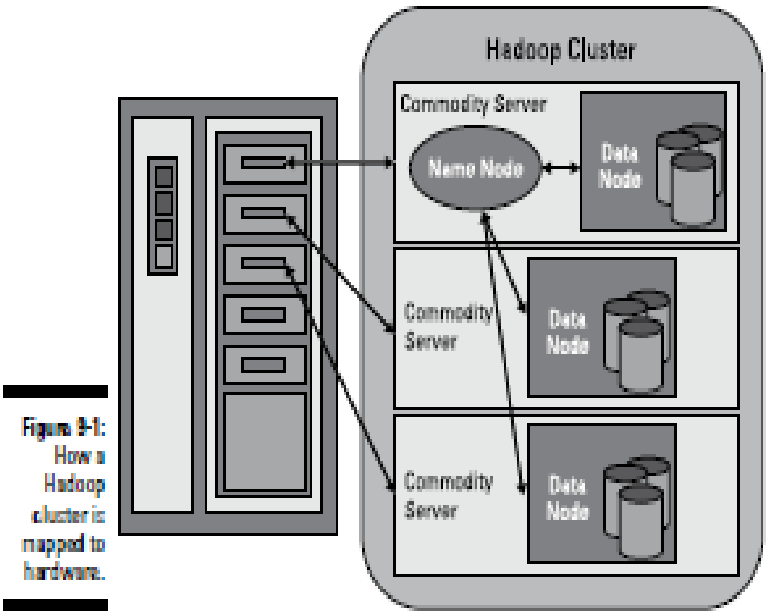


Figure 9-1: How a Hadoop cluster is mapped to hardware.

Name Nodes

- HDFS works by breaking large files into smaller pieces called blocks. The blocks are stored on data nodes, and it is the responsibility of the Name Node to know what blocks on which data nodes make up the complete file. The Name Node also acts as a "traffic cop," managing all access to the files, including reads, writes, creates deletes, and replication of data blocks on the data nodes. The complete collection of all the files in the cluster is sometimes referred to as the file system namespace. It is the Name Node's job to manage this namespace.
- Even though a strong relationship exists between the Name Node and the data nodes, they operate in a "loosely coupled" fashion. This allows the cluster elements to

behave dynamically, adding (or subtracting) servers as the demand increases (or decreases). In a typical configuration, you find one Name Node and possibly a data node running on one physical server in the rack. Other servers run data nodes only.

- Data nodes are not very smart, but the Name Node is. The data nodes constantly ask the Name Node whether there is anything for them to do. This continuous behavior also tells the Name Node what data nodes are out there and how busy they are. The data nodes also communicate among themselves so that they can cooperate during normal file system operations. This is necessary because blocks for one file are likely to be stored on multiple data nodes. Since the Name Node is so critical for correct operation of the cluster, it can and should be replicated to guard against a single point failure.

Data nodes

- Data nodes are not smart, but they are resilient. Within the HDFS cluster, data blocks are replicated across multiple data nodes and access is man- aged by the Name Node. The replication mechanism is designed for optimal efficiency when all the nodes of the cluster are collected into a rack. In fact, the Name Node uses a "rack ID" to keep track of the data nodes in the cluster. HDFS clusters are sometimes referred to as being "rack-aware." Data nodes also provide "heartbeat" messages to detect and ensure connectivity between the NameNode and the data nodes. When a heartbeat is no longer present, the Name Node un maps the data node from the cluster and keeps on operating as though nothing happened. When the heartbeat returns (or a new heartbeat appears), it is added to the cluster transparently with respect to the user or application.
- As with all file systems, data integrity is a key feature. HDFS supports a number of capabilities designed to provide data integrity. As you might expect, when files are broken into blocks and then distributed across different servers in the cluster, any variation in the operation of any element could affect data integrity. HDFS uses transaction logs and checksum validation to ensure integrity across the cluster.
- Transaction logs are a very common practice in file system and database design. They keep track of every operation and are effective in auditing or rebuilding of the file

system should something untoward occur.

- Checksum validations are used to guarantee the contents of files in HDFS. When a client requests a file, it can verify the contents by examining its checksum. If the checksum matches, the file operation can continue. If not, an error is reported. Checksum files are hidden to help avoid tampering.
- Data nodes use local disks in the commodity server for persistence. All the data blocks are stored locally, primarily for performance reasons. Data blocks are replicated across several data nodes, so the failure of one server may not necessarily corrupt a file. The degree of replication, the number of data nodes, and the HDFS namespace are established when the cluster is implemented. Because HDFS is dynamic, all parameters can be adjusted during the operation of the cluster.

HADOOP MAP REDUCE

- To fully understand the capabilities of Hadoop Map Reduce, we need to differentiate between Map Reduce (the algorithm) and an implementation of Map Reduce. Hadoop Map Reduce is an implementation of the algorithm developed and maintained by the Apache Hadoop project. It is helpful to think about this implementation as a Map Reduce engine, because that is exactly how it works. You provide input (fuel), the engine converts the input into output quickly and efficiently, and you get the answers you need. You are using Hadoop to solve business problems, so it is necessary for you to understand how and why it works. So, we take a look at the Hadoop implementation of Map Reduce in more detail.
- Hadoop Map Reduce includes several stages, each with an important set of operations helping to get to your goal of getting the answers you need from big data. The process starts with a user request to run a Map Reduce program and continues until the results are written back to the HDFS. Figure 9-2 illustrates how Map Reduce performs its tasks.

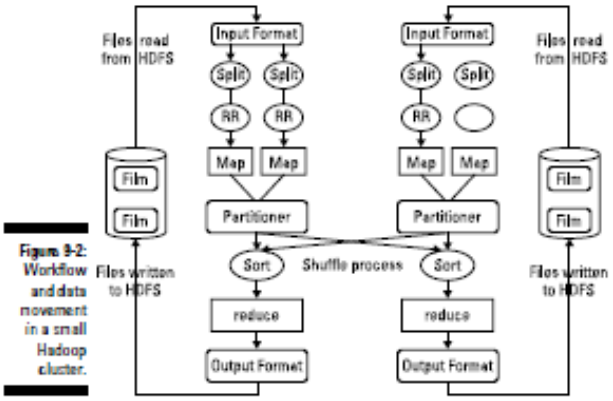


Figure 9-2: Workflow and data movement in a small Hadoop cluster.

HDFS and MapReduce perform their work on nodes in a cluster hosted on racks of commodity servers. To simplify the discussion, the diagram shows only two nodes.

Getting the data ready

- When a client requests a Map Reduce program to run, the first step is to locate and read the input file containing the raw data. The file format is completely arbitrary, but the data must be converted to something the program can process. This is the function of Input Format and Record Reader (RR).
- Input Format decides how the file is going to be broken into smaller pieces for processing using a function called Input Split. It then assigns a Record Reader to transform the raw data for processing by the map. If you read the discussion of map in Chapter 8, you know it requires two inputs: a key and a value. Several types of Record Readers are supplied with Hadoop, offering a wide variety of conversion options. This feature is one of the ways that Hadoop manages the huge variety of data types found in big data problems.

Let the mapping begin

- Your data is now in a form acceptable to map. For each input pair, a distinct instance of map is called to process the data. But what does it do with the processed output, and how can you keep track of them? Map has two additional capabilities to address the

questions. Because map and reduce need to work together to process your data, the program needs to collect the output from the independent mappers and pass it to the reducers. This task is performed by an Output Collector. A Reporter function also provides information gathered from map tasks so that you know when or if the map tasks are complete.

- All this work is being performed on multiple nodes in the Hadoop cluster simultaneously. You may have cases where the output from certain map- ping processes needs to be accumulated before the reducers can begin. Or, some of the intermediate results may need to be processed before reduction. In addition, some of this output may be on a node different from the node where the reducers for that specific output will run. The gathering and shuffling of intermediate results are performed by a partitioner and a sort. The map tasks will deliver the results to a specific partition as inputs to the reduce tasks. After all the map tasks are complete, the intermediate results are gathered in the partition and a shuffling occurs, sorting the output for optimal processing by reduce.

Reduce and combine

- For each output pair, reduce is called to perform its task. In similar fashion to map, reduce gathers its output while all the tasks are processing. Reduce can't begin until all the mapping is done, and it isn't finished until all instances are complete. The output of reduce is also a key and a value. While this is necessary for reduce to do its work, it may not be the most effective output format for your application. Hadoop provides an Output Format feature, and it works very much like Input Format. Output Format takes the key-value pair and organizes the output for writing to HDFS. The last task s to actually write the data to HDFS. This is performed by Record Writer, and it performs similarly to Record Reader except in reverse. It takes the Output Format data and writes it to HDFS in the form necessary for the requirements of the application program.
- The coordination of all these activities was managed in earlier versions of Hadoop by a job scheduler. This scheduler was rudimentary, and as the mix of jobs changed and

grew, it was clear that a different approach was necessary. The primary deficiency in the old scheduler was the lack of resource management. The latest version of Hadoop has this new capability.

- Hadoop Map Reduce is the heart of the Hadoop system. It provides all the capabilities you need to break big data into manageable chunks, process the data in parallel on your distributed cluster, and then make the data available for user consumption or additional processing. And it does all this work in a highly resilient, fault-tolerant manner. This is just the beginning. The Hadoop ecosystem is a large, growing set of tools and technologies designed specifically for cutting your big data problems down to size.

BUILDING A BIG DATA FOUNDATION WITH THE HADOOP ECOSYSTEM

- Trying to tackle big data challenges without a toolbox filled with technology and services is like trying to empty the ocean with a spoon. As core components, Hadoop Map Reduce and HDFS are constantly being improved and provide great starting points, but you need something more. The Hadoop ecosystem provides an ever-expanding collection of tools and technologies specifically created to smooth the development, deployment, and support of big data solutions. Before we look at the key components of the ecosystem, let's take a moment to discuss the Hadoop ecosystem and the role it plays on the big data stage.
- No building is stable without a foundation. While important, stability is not the only important criterion in a building. Each part of the building must support its overall purpose. The walls, floors, stairs, electrical, plumbing, and roof need to complement each other while relying on the foundation for support and integration. It is the same with the Hadoop ecosystem. The foundation is Map Reduce and HDFS. They provide the basic structure and integration services needed to support the core requirements of big data solutions. The remainder of the ecosystem provides the components you need to build and manage purpose-driven big data applications for the real world.
- In the absence of the ecosystem it would be incumbent on developers, data- base administrators, system and network managers, and others to identify and agree on a

set of technologies to build and deploy big data solutions.

- This is often the case when businesses want to adapt new and emerging technology trends. The chore of cobbling together technologies in a new market is daunting. That is why the Hadoop ecosystem is so fundamental to the success of big data. It is the most comprehensive collection of tools and technologies available today to target big data challenges. The ecosystem facilitates the creation of new opportunities for the widespread adoption of big data by businesses and organizations.

POSSIBLE QUESTIONS

UNIT – III

PART – A (20 MARKS)

(Q.NO 1 TO 20 Online Examinations)

PART – B (6 MARKS)

1. Discuss in detail about Hadoop.
2. Explain in detail about the HDFS
3. Explain about how to build a big data foundation with the Hadoop ecosystem.
4. Explain in detail about the importance of virtualization to big data
5. Describe about the mining big data with Hive.
6. Discuss in detail about the interacting with the Hadoop ecosystem.
7. Explain about the mapping begin and Reduce & Combine.
8. Explain about the big data analysis and the data warehouse
9. Explain in detail about how to use big data with HBase.
10. Explain about managing resources and applications with Hadoop.

PART – C (10 MARKS)

(Compulsory Question)

1. Discuss in detail about Hadoop Map-Reduce.
2. Explain the History of Hadoop
3. Comparison between HDFS and Map Reduce
4. Difference between Name node and Data node
5. Explain the benefits and Features of Hadoop Framework

UNIT – IV

SYLLABUS

Big Data Analytics – Text Analytics and Big Data – Customize Approaches for analysis of Big Data

BIG ANALYTICS

BASIC ANALYTICS

- Basic analytics can be used to explore your data, if you're not sure what you have, but you think something is of value. This might include simple visualizations or simple statistics. Basic analysis is often used when you have large amounts of disparate data. Here are some examples:

Slicing and dicing:

- Slicing and dicing refers to breaking down your data into smaller sets of data that are easier to explore. For example, you might have a scientific data set of water column data from many different locations that contains numerous variables captured from multiple sensors. Attributes might include temperature, pressure, transparency, dissolved oxygen, pH, salinity, and so on, collected over time. You might want some simple graphs or plots that let you explore your data across different dimensions, such as temperature versus pH or transparency versus salinity. You might want some basic statistics such as average or range for each attribute, from each height, for the time period. The point is that you might use this basic type of exploration of the variables to ask specific questions in your problem space. The difference between this kind of analysis and what happens in a basic business intelligence system is that you're dealing with huge volumes of data where you might not know how much query space you'll need to examine it and you're probably going to want to run computations in real time.
- Basic monitoring: You might also want to monitor large volumes of data in real time. For example, you might want to monitor the water column attributes in the

preceding example every second for an extended period of time from hundreds of locations and at varying heights in the water column. This would produce a huge data set. Or, you might be interested in monitoring the buzz associated with your product every minute when you launch an ad campaign. Whereas the water column data set might produce a large amount of relatively structured time-sensitive data, the social media campaign is going to produce large amounts of disparate kinds of data from multiple sources across the Internet.

- **Anomaly identification:** You might want to identify anomalies, such as an event where the actual observation differs from what you expected, in your data because that may clue you in that something is going wrong with your organization, manufacturing process, and so on. For example, you might want to analyze the records for your manufacturing operation to determine whether one kind of machine, or one operator, has a higher incidence of a certain kind of problem. This might involve some simple statistics like moving averages triggered by an alert from the problematic machine.

Advanced analytics

- Advanced analytics provides algorithms for complex analysis of either structured or unstructured data. It includes sophisticated statistical models, machine learning, neural networks, text analytics, and other advanced data-mining techniques. (See the sidebar "What is data mining?" later in this chapter, for more detail on data mining.) Among its many use cases, advanced analytics can be deployed to find pat- terns in data, prediction, forecasting, and complex event processing.
- While advanced analytics has been used by statisticians and mathematicians for decades, it was not as big a part of the analytics landscape as it is today. Consider those 20 years ago, statisticians at companies were able to predict who might drop a service using advanced survival analysis or machine learning techniques. However, it was difficult to persuade other people in the organization to understand exactly what this meant and how it could be used to provide a competitive advantage. For one thing, it was difficult to obtain the computational power needed to interpret data that kept

changing through time.

- Today, advanced analytics is becoming more main stream. With increases in computational power, improved data infrastructure, new algorithm develop- ment, and the need to obtain better insight from increasingly vast amounts of data, companies are pushing toward utilizing advanced analytics as part of their decision-making process. Businesses realize that better insights can provide a superior competitive position.
- Here are a few examples of advanced analytics for big data:
- **Predictive modeling:** Predictive modeling is one of the most popular big data advanced analytics use cases. A predictive model is a statistical or data-mining solution consisting of algorithms and techniques that can be used on both structured and unstructured data (together or individually) to determine future outcomes. For example, a telecommunications company might use a predictive model to predict customers who might drop its service. In the big data world, you might have large numbers of predictive attributes across huge amounts of observations. Whereas in the past, it might have taken hours (or longer) to run a predictive model, with a large amount of data on your desktop, you might be able to now run it iteratively hundreds of times if you have a big data infrastructure in place.

TEXT ANALYTICS:

- Unstructured data is such a big part of big data, so text analytics — the process of analyzing unstructured text, extracting relevant information, and transforming it into structured information that can then be leveraged in various ways — has become an important component of the big data ecosystem. The analysis and extraction processes used in text analytics take advantage of techniques that originated in computational linguistics, statistics, and other computer science disciplines. Text analytics is being used in all sorts of analysis, from predicting churn, to fraud, and to social media analytics.
- **Other statistical and data-mining algorithms:** This may include advanced forecasting, optimization, cluster analysis for segmentation or even micro

segmentation, or affinity analysis. Advanced analytics doesn't require big data. However, being able to apply advanced analytics with big data can provide some important results.

Operationalized analytics

- When you operationalize analytics, you make them part of a business process. For example, statisticians at an insurance company might build a model that predicts the likelihood of a claim being fraudulent. The model, along with some decision rules, could be included in the company's claims-processing system to flag claims with a high probability of fraud. These claims would be sent to an investigation unit for further review. In other cases, the model itself might not be as apparent to the end user. For example, a model could be built to predict customers who are good targets for up selling when they call into a call center. The call center agent, while on the phone with the customer, would receive a message on specific additional products to sell to this customer. The agent might not even know that a predictive model was working behind the scenes to make this recommendation.

Monetizing analytics

- Analytics can be used to optimize your business to create better decisions and drive bottom- and top-line revenue. However, big data analytics can also be used to derive revenue above and beyond the insights it provides just for your own department or company. You might be able to assemble a unique data set that is valuable to other companies, as well. For example, credit card providers take the data they assemble to offer value-added analytics products. Likewise, with financial institutions. Telecommunications companies are beginning to sell location-based insights to retailers. The idea is that various sources of data, such as billing data, location data, text-messaging data, or web-browsing data can be used together or separately to make inferences about customer behavior patterns that retailers would find useful. As a regulated industry, they must do so in compliance with legislation and privacy policies.

Understanding Text Analytics

- Numerous methods exist for analyzing unstructured data. Historically, these techniques came out of technical areas such as Natural Language Processing (NLP), knowledge discovery, data mining, information retrieval, and statistics. Text analytics is the process of analyzing unstructured text, extracting relevant information, and transforming it into structured information that can then be leveraged in various ways. The analysis and extraction processes take advantage of techniques that originated in computational linguistics, statistics, and other computer science disciplines.
- The text analytics process uses various algorithms, such as understanding sentence structure, to analyze the unstructured text and then extract information, and transform that information into structured data.

POSSIBLE QUESTIONS

UNIT – II

PART – A (20 MARKS)

(Q.NO 1 TO 20 Online Examinations)

PART – B (6 MARKS)

1. Explain the process of Basic analytics
2. Explain about the modifying business intelligence products to handle big data
3. Explain in detail about big data analytics examples.
4. Explain about the unstructured data and understanding text analytics.
5. Explain about Analysis and Extraction techniques
6. Explain about Text analytics tools for big data.
7. Explain about building new models and approaches to support big data.
8. Explain in detail about understanding different approaches to big data analysis.
9. Explain about the characteristics of a big data analysis framework
10. Explain about big data paradox.

PART – C (10 MARKS)

(Compulsory Question)

1. Discuss in detail about Operationalized Analytics.
2. Explain the process of Monetizing Analytics
3. Discuss in detail about Advanced Analytics
4. Difference between Basic Analytics and Advanced Analytics
5. Explain the benefits and Features of Text Analytics

UNIT – V

SYLLABUS

Integrating Data Sources – Real time Data Streams and Complex Event Processing - Operation

INTEGRATING DATA SOURCES

UNDERSTANDING THE FUNDAMENTALS OF BIG DATA INTEGRATION

- The elements of the big data platform manage data in new ways as compared to the traditional relational database. This is because of the need to have the scalability and high performance required to manage both structured and unstructured data. Components of the big data ecosystem ranging from Hadoop to NoSQL DB, MongoDB, Cassandra, and HBase all have their own approach for extracting and loading data. As a result, your teams may need to develop new skills to manage the integration process across these plat- forms. However, many of your company's data management best practices will become even more important as you move into the world of big data
- While big data introduces a new level of integration complexity, the basic fundamental principles still apply. Your business objective needs to be focused on delivering quality and trusted data to the organization at the right time and in the right context. To ensure this trust, you need to establish common rules for data quality with an emphasis on accuracy and completeness of data. In addition, you need a comprehensive approach to developing enterprise metadata, keeping track of data lineage and governance to support integration of your data.
- At the same time, traditional tools for data integration are evolving to handle the increasing variety of unstructured data and the growing volume and velocity of big data. While traditional forms of integration take on new meanings in a big data world, your integration technologies need a common plat- form that supports data quality and profiling.

- To make sound business decisions based on big data analysis, this information needs to be trusted and understood at all levels of the organization. While it will probably not be cost or time effective to be overly concerned with data quality in the exploratory stage of a big data analysis, eventually quality and trust must play a role if the results are to be incorporated in the business process. Information needs to be delivered to the business in a trusted, controlled, consistent, and flexible way across the enterprise, regardless of the requirements specific to individual systems or applications. To accomplish this goal, three basic principles apply:
- **You must create a common understanding of data definitions.** At the initial stages of your big data analysis, you are not likely to have the same level of control over data definitions as you do with your operational data. However, once you have identified the patterns that are most relevant to your business, you need the capability to map data elements to a common definition. That common definition is then carried forward into operational data, data warehouses, reporting, and business processes.
- **You must develop of a set of data services to qualify the data and make it consistent and ultimately trustworthy.** When you're unstructured and big data sources are integrated with structured operational data, you need to be confident that the results will be meaningful.
- **You need a streamlined way to integrate your big data sources and systems of record.** In order to make good decisions based on the results of your big data analysis, you need to deliver information at the right time and with the right context. Your big data integration process should ensure consistency and reliability.
- To integrate data across mixed application environments, you need to get data from one data environment (source) to another data environment (target). Extract, transform, and load (ETL) technologies have been used to accomplish this in traditional data warehouse environments. The role of ETL is evolving to handle newer data management environments like Hadoop. In a big data environment, you may need to combine tools that support batch integration processes (using ETL) with real-time integration and federation across multiple sources. For example, a pharmaceutical company may need to blend data stored in its Master Data Management (MDM) system with big data sources

on medical outcomes of customer drug usage. Companies use MDM to facilitate the collecting, aggregating, consolidating, and delivering of consistent and reliable data in a controlled manner across the enterprise. In addition, new tools like Sqoop and Scribe are used to support integration of big data environments. You also find an increasing emphasis on using extract, load, and transform (ELT) technologies. These technologies are described next.

Defining Traditional ETL

- ETL tools combine three important functions required to get data from one data environment and put it into another data environment. Traditionally, ETL has been used with batch processing in data warehouse environments. Data warehouses provide business users with a way to consolidate information across disparate sources (such as enterprise resource planning [ERP] and customer relationship management [CRM]) to analyze and report on data relevant to their specific business focus. ETL tools are used to transform the data into the format required by the data warehouse. The transformation is actually done in an intermediate location before the data is loaded into the data warehouse. Many software vendors, including IBM, Informatica, Pervasive, Talend, and Pentaho, provide ETL software tools.
- ETL provides the underlying infrastructure for integration by performing three important functions:
  - **Extract:** Read data from the source database.
  - **Transform:** Convert the format of the extracted data so that it conforms to the requirements of the target database. Transformation is done by using rules or merging data with other data.
  - **Load:** Write data to the target database.
- However, ETL is evolving to support integration across much more than traditional data warehouses. ETL can support integration across transactional systems, operational data stores, BI platforms, MDM hubs, the cloud, and Hadoop platforms. ETL software vendors are extending their solutions to provide big data extraction, transformation, and

loading between Hadoop and traditional data management platforms. ETL and software tools for other data integration processes like data cleansing, profiling, and auditing all work on different aspects of the data to ensure that the data will be deemed trust-worthy. ETL tools integrate with data quality tools, and many incorporate tools for data cleansing, data mapping, and identifying data lineage.

Data transformation

- Data transformation is the process of changing the format of data so that it can be used by different applications. This may mean a change from the format the data is stored in into the format needed by the application that will use the data. This process also includes mapping instructions so that applications are told how to get the data they need to process. The process of data transformation is made far more complex because of the staggering growth in the amount of unstructured data. A business application such as a customer relationship management or sales management system typically has specific requirements for how the data it needs should be stored. The data is likely to be structured in the organized rows and columns of a relational database. Data is semi-structured or unstructured if it does not follow these very rigid format requirements. The information contained in an e-mail message is considered unstructured, for example. Some of a company's most important information is in unstructured and semi-structured Forms such as documents, e-mail messages, complex messaging formats, customer support interactions, transactions, and information coming from pack- aged applications like ERP and CRM.
- Data transformation tools are not designed to work well with unstructured data. As a result, companies needing to incorporate unstructured information into its business process decision making have been faced with a significant amount of manual coding to accomplish the required data integration. Given the growth and importance of unstructured data to decision making, ETL solutions from major vendors are beginning to offer standardized approaches to transforming unstructured data so that it can be more easily integrated with operational structured data.

\

Understanding ELT — Extract, Load, and Transform

- ELT stands for extract, load, and transform. It performs the same functions as ETL, but in a different order. Early databases did not have the technical capability to transform the data. Therefore, ETL tools extracted the data to an intermediary location to perform the transformation before loading the data to the data warehouse. However, this restriction is no longer a problem, thanks to technology advances such as massively parallel processing systems and columnar databases. As a result, ELT tools can transform the data in the source or target database without requiring an ETL server. Why use ELT with big data? The performance is faster and more easily scalable. ELT uses structured query language (SQL) to transform the data. Many traditional ETL tools also offer ELT so that you can use both, depending on which option is best for your situation.

Prioritizing Big Data Quality

- Getting the right perspective on data quality can be very challenging in the world of big data. With the majority of big data sources, you need to assume that you are working with data that is not clean. In fact, the overwhelming abundance of seemingly random and disconnected data in streams of social media data is one of the things that make it so useful to businesses. You start by searching petabytes of data without knowing what you might find after you start looking for patterns in the data. You need to accept the fact that a lot of noise will exist in the data. It is only by searching and pattern matching that you will be able to find some sparks of truth in the midst of some very dirty data. Of course, some big data sources such as data from RFID tags or sensors have better-established rules than social media data. Sensor data should be reasonably clean, although you may expect to find some errors. It is always your responsibility when analyzing massive amounts of data to plan for the quality level of that data. You should follow a two-phase approach to data quality:

- Phase 1: Look for patterns in big data without concern for data quality.
-
- Phase 2: After you locate your patterns and establish results that are important to the

business, apply the same data quality standards that you apply to your traditional data sources. You want to avoid collecting and managing big data that is not important to the business and will potentially corrupt other data elements in Hadoop or other big data platforms.

- As you begin to incorporate the outcomes of your big data analysis into your business process, recognize that high-quality data is essential for a company to make sound business decisions. This is true for big data as well as traditional data. The quality of data refers to characteristics about the data, including consistency, accuracy, reliability, completeness, timeliness, reasonableness, and validity. Data quality software makes sure that data elements are represented in the same way across different data stores or systems to increase the consistency of the data.
- **For example**, one data store may use two lines for a customer's address and another data store may use one line. This difference in the way the data is represented can result in inaccurate information about customers, such as one customer being identified as two different customers. A corporation might use dozens of variations of its company name when it buys products. Data quality software can be used to identify all the variations of the company name in your different data stores and ensure that you know everything that this customer purchases from your business. This process is called providing a single view of customer or product. Data quality software matches data across different systems and cleans up or removes redundant data. The data quality process provides the business with information that is easier to use, interpret, and understand.

STREAMING DATA AND COMPLEX EVENT PROCESSING

- Streaming computing is designed to handle a continuous stream of a large amount of unstructured data. In contrast, Complex Event Processing (CEP) typically deals with a few variables that need to be correlated with a specific business process. In many situations, CEP is dependent on data streams. However, CEP is not required for streaming data. Like streaming data, CEP relies on analyzing streams of data in motion.

DATA STREAMING

- Streaming data is an analytic computing platform that is focused on speed. This is because these applications require a continuous stream of often unstructured data to be processed. Therefore, data is continuously analyzed and transformed in memory before it is stored on a disk. Processing streams of data works by processing "time windows" of data in memory across a cluster of servers.. The primary difference is the issue of velocity. In the Hadoop cluster, data is collected in batch mode and then processed. Speed matters less in Hadoop than it does in data streaming. Some key principles define when using streams is most appropriate:
- When it is necessary to determine a retail buying opportunity at the point of engagement, either via social media or via permission-based messaging
- Collecting information about the movement around a secure site
- To be able to react to an event that needs an immediate response, such as a service outage or a change in a patient's medical condition
- Real-time calculation of costs that are dependent on variables such as usage and available resources
- Streaming data is useful when analytics need to be done in real time while the data is in motion. In fact, the value of the analysis (and often the data) decreases with time. For example, if you can't analyze and act immediately, a sales opportunity might be lost or a threat might go undetected.
- The following are a few examples that can help explain how this is useful.
    - A power plant needs to be a highly secure environment so that unauthorized individuals do not interfere with the delivery of power to customers. Companies often place sensors around the perimeter of a site to detect move- ment. But a problem could exist. A huge difference exists between a rabbit that scurries around the site and a car driving by quickly and deliberately. Therefore, the vast amount of data coming from these sensors needs to be analyzed in real time so that an alarm is sounded only when an actual threat exists.

- A telecommunications company in a highly competitive market wants to make sure that outages are carefully monitored so that a detected drop in service levels can be escalated to the appropriate group. Communications systems generate huge volumes of data that have to be analyzed in real time to take the appropriate action. A delay in detecting an error can seriously impact customer satisfaction.
- An oil exploration company drilling at sea needs to know exactly where the sources of oil are and what other environmental factors might impact their operations. Therefore, it needs to know details such as water depth, temperature, ice flows, and so on. This massive amount of data needs to be analyzed and computed so that mistakes are avoided.
- A medical diagnostic group was required to be able to take massive amounts of data from brain scans and analyze the results in real time to determine where the source of a problem is and what type of action needed to be taken to help the patient.

Using Complex Event Processing

- Both streams and Complex Event Processing (CEP) are intended to manage data in motion. But the uses of these two technologies are quite different. While streams are intended to analyze large volumes of data in real time, Complex Event Processing is a technique for tracking, analyzing, and processing data as an event happens. This information is then processed and communicated based on business rules and processes. The idea behind CEP is to be able to establish the correlation between streams of information and match the resulting pattern with defined behaviors such as mitigating a threat or seizing an opportunity.
- CEP is an advanced approach based on simple event processing that collects and combines data from different relevant sources to discover events and patterns that can result in action.
- Here is an example. A retail chain creates a tiered loyalty program to increase repeat sales — especially for customers who spend more than $1,000 a year. It is important that the

company creates a platform that could keep these critical customers coming back and spending more. Using a CEP platform, as soon as a high-valued customer uses the loyalty program, the system triggers a process that offers the customer an extra discount on a related product. Another process rule could give the customer an unexpected surprise — an extra discount or a sample of a new product. The company also adds a new program that links the loyalty program to a mobile application. When a loyal customer walks near a store, a text message offers the customer a discounted price. At the same time, if that loyal customer writes something negative on a social media site, the customer care department is notified and an apology is issued. It is quite likely that we are dealing with a huge number of customers with a significant number of interactions. But it would not be enough to simply stream the data and analyze that data. To achieve the business goals the retailer wanted to achieve would require executing a process to respond to the results of the analysis.

Operationalizing Big Data

Making Big Data a Part of Your Operational Process

- The best way to start making big data a part of your business process is to begin by planning an integration strategy. The data — whether it is a traditional data source or big data — needs to be integrated as a seamless part of the inner workings of the processes. Can big data be ancillary to the business process? The answer is yes, but only if little or no dependency exists between transactional data and big data. Certainly you can introduce big data to your organization as a parallel activity. However, if you want to get the most from big data, it needs to be integrated into your existing business operating processes. We take a look at how to accomplish this task. In the next section, we discuss the importance of data integration in making big data operational.

Integrating big data

- Just having access to big data sources is not enough. Soon there will be petabytes of data and hundreds of access mechanisms for you to choose from. But which streams and what kinds of data do you need? The identification of the "right" sources of data is similar to

what we have done in the past:

- ✓ Understand the problem you are trying to solve
- ✓ Identify the processes involved
- ✓ Identify the information required to solve the problem
- ✓ Gather the data, process it, and analyze the results

- This process may sound familiar because businesses have been doing a variation of this algorithm for decades. So is big data different? Yes, even though we have been dealing with large amounts of operational data for years, big data introduces new types of data into people's professional and personal lives. Twitter streams, Facebook posts, sensor data, RFID data, security logs, video data, and many other new sources of information are emerging almost daily. As these sources of big data emerge and expand, people are trying to find ways to use this data to better serve customers, partners, and suppliers. Organizations are looking for ways to use this data to predict the future and to take better actions. We look at an example to understand the importance of integrating big data with operating processes.
- Healthcare is one of the most important and complex areas of investment today. It is also an area that increasingly produces more data in more forms than most industries. Therefore, healthcare is likely to greatly benefit by new forms of big data. The healthcare providers, insurers, researchers, and healthcare practitioners often make decisions about treatment options with data that is incomplete or not relevant to specific illnesses. Part of the reason for this disparity is that it is very difficult to effectively gather and process data for individual patients. Data elements are often stored and managed in different places by different organizations. In addition, clinical research that is being conducted all over the world can be helpful in determining the context for how a specific disease or illness might be approached and managed. Big data can help change this problem.
- So, we apply our algorithm to a standard data healthcare scenario:

1. Understand the problem we are trying to solve:
    a. Need to treat a patient with a specific type of cancer
2. Identify the processes involved:
    a. Diagnosis and testing
    b. Results analysis including researching treatment options
    c. Definition of treatment protocol
    d. Monitor patient and adjust treatment as needed
3. Identify the information required to solve the problem:
    a. Patient history
    b. Blood, tissue, test results, and so on
    c. Statistical results of treatment options
4. Gather the data, process it, and analyze the results:
    a. Commence treatment
    b. Monitor patient and adjust treatment as needed

Figure 17-1 illustrates the process.

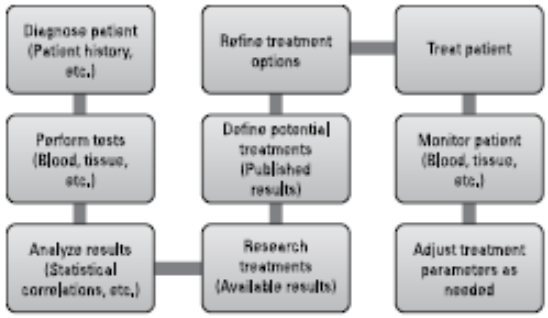


Figure 17-1: Process flow of the traditional patient diagnostic process.

This is how medical practitioners work with patients today. Most of the data is local to a healthcare network, and physicians have little time to go outside the network to find the latest information or practice.

Incorporating big data into the diagnosis of diseases

- Across the world, big data sources for healthcare are being created and made available for integration into existing processes. Clinical trial data, genetics and genetic mutation data, protein therapeutics data, and many other new sources of information can be harvested to improve daily healthcare processes. Social media can and will be used to augment existing data and processes to provide more personalized views of treatment and therapies. New medical devices will control treatments and transmit telemetry data for real- time and other kinds of analytics. The task ahead is to understand these new sources of data and complement the existing data and processes with the new big data types.
- So, what would the healthcare process look like with the introduction of big data into the operational process of identifying and managing patient health? Here is an example of what the future might look like:

1. Understand the problem we are trying to solve:
    a. Need to treat a patient with a specific type of cancer
2. Identify the processes involved:
    a. Diagnosis and testing (identify genetic mutation)
    b. Results analysis including researching treatment options, clinical trial analysis, genetic analysis, and protein analysis
    c. Definition of treatment protocol, possibly including gene or protein therapy
    d. Monitor patient and adjust treatment as needed using new wireless device for personalized treatment delivery and monitoring. Patient uses social media to document overall experience.
3. Identify the information required to solve the problem:
    a. Patient history
    b. Blood, tissue, test results, and so on

c. Statistical results of treatment options

d. Clinical trial data

e. Genetics data

f. Protein data

g. Social media data

4. Gather the data, process it, and analyze the results:

a. Commence treatment

b. Monitor patient and adjust treatment as needed

Figure 17-2 identifies the same operational process as before, but with big data integrations.

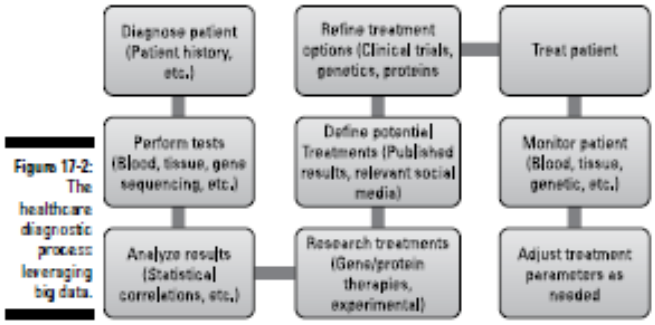


Figure 17-2: The healthcare diagnostic process leveraging big data.

- This represents the optimal case where no new processes need to be created to support big data integrations. While the processes are relatively unchanged, the underlying technologies include the applications that will need to be altered to accommodate the impact of characteristics of big data, including the volume of data, the variety of data sources, and the speed or velocity required to process that data.
- The introduction of big data into the process of managing healthcare will make a big difference in effectiveness to diagnosing and managing healthcare in the future. This same operational approach process can be applied to a variety of industries, ranging from oil and gas to financial markets and retail, to name a few. What are the keys to successfully applying big data to operational processes? Here are some of the most

important issues to consider:

- Fully understand the current process.
- Fully understand where gaps exist in information.
- Identify relevant big data sources.
- Design a process to seamlessly integrate the data now and as it changes.
- Modify analysis and decision-making processes to incorporate the use of big data.

POSSIBLE QUESTIONS

UNIT – V

PART – A (20 MARKS)

(Q.NO 1 TO 20 Online Examinations)

PART – B (6 MARKS)

1. Discuss in detail about integrating data sources.
2. Explain about integrating Big data
3. Discuss in detail about the understanding the ELT
4. Explain about the streaming data and complex event processing
5. Discuss in detail about the how to use streaming data.
6. Explain about the impact of streaming data and CEP on business.
7. Explain about how to make big data as an operational business.
8. Explain about understanding big data workflows
9. Explain about differentiating CEP from streams.
10. Explain about the ensuring validity, veracity and volatility of big data.

PART – C (10 MARKS)

(Compulsory Question)

1. Discuss in detail about ELT Process.
2. Explain the Features and benefits of Data Streams
3. Comparison between Integrating Data source and Integrating Big Data
4. Difference between Streaming Data and Complex Event Processing
5. Explain the process of Big data Workflows

KARPAGAM ACADEMY OF HIGHER EDUCATION

Coimbatore – 641 021.

(For the Candidates admitted from 2016 onwards)

DEPARTMENT OF COMPUTER SCIENCE, CA & IT

UNIT - I : (Objective Type Multiple choice Questions each Question carries one Mark)

BIG DATA ANALYTICS [16CAP505D]

PART - A (Online Examination)

Questions	Opt1	opt2	opt3	opt4	KEY
_____ is a Web search software	Imphala	Nutch	Oozie	Manmgy	Nutch
_____ defines an open application	Bigred	Nuvem	Oozie	Imphala	Nuvem
_____ is OData implementation in Java.	Bigred	Nuvem	Olingo	Onami	Onami
. _____ is an open source SQL query engine for Apache HBase	Pig	Phoenix	Pivot	Manmgy	Phoenix
_____ provides multiple language implementations of the Advanced Messaged Queuing Protocol (AMQP)	RTA	Qpid	RAT	Nuvem	Qpid
_____ is A WEb And SOcial Mashup Engine.	ServiceMix	Samza	Rave	Oozie	Rave
The _____ project will create an ESB and component suite based on the Java Business Interface (JBI) standard – JSR 208.	ServiceMix	Samza	Rave	Pivot	ServiceMix
Which of the following is spatial information system ?	Sling	Solr	SIS	d)SLS	SIS
Stratos will be a polyglot _____ framework	Daas	PaaS	Saas	Raas	PaaS
_____ supports random-writable and advance-able sparse bitsets	Stratos	Kafka	Sqoop	Lucene	Lucene
_____ is an open-source version control system.	Stratos	Kafka	Sqoop	Subversion	Subversion

_____ is a distributed data warehouse system for Hadoop.	Stratos	Tajo	Sqoop	Lucene	Tajo
_____ is a distributed, fault-tolerant, and high-performance realtime computation system	Knife	Storm	Sqoop	Lucene	Storm
_____ is a standard compliant XML Query processor	Whirr	VXQuery	Knife	Lens	VXQuery
Apache _____ is a project that enables development and consumption of REST style web services.	Wives	Wink	Wig	Stratos	Wink
_____ is a log collection and correlation software with reporting and alarming functionalities.	Lucene	ALOIS	Imphal	Storm	ALOIS
_____ is a non-blocking, asynchronous, event driven high performance web framework.	AWS	AWF	AWT	ASW	AWF
_____ is the architectural center of Hadoop that allows multiple data processing engines.	YARN	Hive	Incubator	Chuckwa	YARN
YARN's dynamic allocation of cluster resources improves utilization over more static _____ rules used in early versions of Hadoop.	Hive	MapReduce	Imphala	Wink	MapReduce
The _____ is a framework-specific entity that negotiates resources from the ResourceManager	NodeManager	ResourceManager	ApplicationMaster	d)TaskMaster	ApplicationMaster
Apache Hadoop YARN stands for :	Yet Another Reserve Negotiator	Yet Another Resource Network	Yet Another Resource Negotiator	Yet Any Resource Negotiator	Yet Another Resource Negotiator
MapReduce has undergone a complete overhaul in hadoop :	0.21	0.23	0.24	0.26	0.23
The _____ is the ultimate authority that arbitrates resources among all the applications in the system.	NodeManager	ResourceManager	ApplicationMaster	d)dynamicMaster	ResourceManager

The _____ is responsible for allocating resources to the various running applications subject to familiar constraints of capacities, queues etc.	Manager	Master	Scheduler	d)Task	Scheduler
The CapacityScheduler supports _____ queues to allow for more predictable sharing of cluster resources.	Networked	Hierarchial	Partition	d)Unpartition	Hierarchial
ZooKeeper itself is intended to be replicated over a sets of hosts called :	chunks	ensemble	subdomains	d)unchunks	ensemble
_____ of the guarantee is provided by Zookeeper	Interactivity	Flexibility	Scalability	Reliability	Reliability
ZooKeeper is especially fast in _____ workloads	write	read-dominant	read-write	d)read	read-dominant
When a _____ is triggered the client receives a packet saying that the znode has changed.	event	watch	row	value	watch
The underlying client-server protocol has changed in version _____ of ZooKeeper.	2.0.0	3.0.0	4.0.0	6.0.0	3.0.0
A number of constants used in the client ZooKeeper API were renamed in order to reduce _____ collision	value	namespace	counter	d)signal	namespace
ZooKeeper allows distributed processes to coordinate with each other through registers, known as :	znodes	hnodes	vnodes	rnodes	znodes
The HDFS client software implements _____ checking on the contents of HDFS files.	metastore	parity	checksum	d)unparity	checksum
The _____ machine is a single point of failure for an HDFS cluster.	DataNode	NameNode	ActionNode	UnitNode	NameNode
The _____ and the EditLog are central data structures of HDFS.	DsImage	FsImage	FsImages	d) DsImages	FsImage
_____ support storing a copy of data at a particular instant of time.	Data Image	Datanots	Snapshots	DataFile	Snapshots

Automatic restart and _____ of the NameNode software to another machine is not supported.	failover	end	scalability	d)resource	failover
HDFS, by default, replicates each data block _____ times on different nodes and on at least ____ racks.	3,2	1,2	2,3	d)3,1	3,2
_____ stores its metadata on multiple disks that typically include a non-local file server.	DataNode	NameNode	ActionNode	UnitNode	NameNode
The HDFS file system is temporarily unavailable whenever the HDFS _____ is down.	DataNode	NameNode	ActionNode	UnitNode	NameNode
_____ is a platform for constructing data flows for extract, transform, and load (ETL) processing and analysis of large datasets.	Pig Latin	Oozie	Pig	Hive	Pig
_____ hides the limitations of Java behind a powerful and concise Clojure API for Cascading.	Scalding	HCatalog	Cascalog	d)Catalog	Cascalog
Hive also support custom extensions written in :	C#	Java	C	C++	Java
_____ is the most popular high-level Java API in Hadoop Ecosystem	Scalding	HCatalog	Cascalog	Cascading	Cascading
_____ is general-purpose computing model and runtime system for distributed data analytics.	Mapreduce	Drill	Oozie	ActionNode	Mapreduce
The Pig Latin scripting language is not only a higher-level data flow language but also has operators similar to :	SQL	JSON	XML	d) BSON	SQL
_____ jobs are optimized for scalability but not latency.	Mapreduce	Drill	Oozie	Hive	Hive
_____ is a framework for performing remote procedure calls and data serialization.	Drill	BigTop	Avro	Chukwa	Avro
Avro-backed tables can simply be created by using _____ in a DDL statement.	“STORED AS AVRO”	b) “STORED AS HIVE”	c) “STORED AS AVROHIVE”	d) “STORED AS SERDE”	“STORED AS AVRO”
Types that may be null must be defined as a _____ of that type and Null within Avro.	Union	Intersection	Set	d)Unset	Union

The files that are written by the _____ job are valid Avro files.	Avro	Map Reduce	Hive	Drill	Hive
Use _____ and embed the schema in the create statement.	schema.literal	schema.lit	row.literal	row.lit	schema.literal
_____ is interpolated into the quotes to correctly handle spaces within the schema.	$SCHEMA	$ROW	$SCHEMASPACES	$NAMESPACES	$SCHEMA
_____ was designed to overcome limitations of the other Hive file formats.	ORC	OPC	ODC	d)OLC	ORC
An ORC file contains groups of row data called :	postscript	stripes	script	d)scriptstart	stripes
The Mapper implementation processes one line at a time via _____ method.	map	reduce	mapper	reducer	map
The Hadoop MapReduce framework spawns one map task for each _____ generated by the InputFormat for the job.	OutputSplit	InputSplit	InputSplitStream	OutputSplitStream	InputSplit
Users can control which keys (and hence records) go to which Reducer by implementing a custom :	Partitioner	OutputSplit	Reporter	InputSplit	Partitioner
Applications can use the _____ to report progress and set application-level status messages	Partitioner	OutputSplit	Reporter	InputSplit	Reporter
The right level of parallelism for maps seems to be around _____ maps per-node	1-10	10-100	100-150	150-200	10-100

KARPAGAM ACADEMY OF HIGHER EDUCATION

Coimbatore – 641 021.

(For the Candidates admitted from 2016 onwards)

DEPARTMENT OF COMPUTER SCIENCE, CA & IT

UNIT - II : (Objective Type Multiple choice Questions each Question carries one Mark)

BIG DATA ANALYTICS [16CAP505D]

PART - A (Online Examination)

Questions	Opt1	opt2	opt3	opt4	KEY
_____ is a Web search software	Imphala	Nutch	Oozie	Manmgy	Nutch
_____ defines an open application	Bigred	Nuvem	Oozie	Imphala	Nuvem
_____ is OData implementation in Java.	Bigred	Nuvem	Olingo	Onami	Onami
. _____ is an open source SQL query engine for Apache HBase	Pig	Phoenix	Pivot	Manmgy	Phoenix
_____ provides multiple language implementations of the Advanced Messaged Queuing Protocol (AMQP)	RTA	Qpid	RAT	Nuvem	Qpid
_____ is A WEb And SOcial Mashup Engine.	ServiceMix	Samza	Rave	Oozie	Rave
The _____ project will create an ESB and component suite based on the Java Business Interface (JBI) standard – JSR 208.	ServiceMix	Samza	Rave	Pivot	ServiceMix
Which of the following is spatial information system ?	Sling	Solr	SIS	SLS	SIS
Stratos will be a polyglot _____ framework	Daas	PaaS	Saas	Raas	PaaS
_____ supports random-writable and advance-able sparse bitsets	Stratos	Kafka	Sqoop	Lucene	Lucene
_____ is an open-source version control system.	Stratos	Kafka	Sqoop	Subversion	Subversion

_____ is a distributed data warehouse system for Hadoop.	Stratos	Tajo	Sqoop	Lucene	Tajo
_____ is a distributed, fault-tolerant, and high-performance realtime computation system	Knife	Storm	Sqoop	Lucene	Storm
_____ is a standard compliant XML Query processor	Whirr	VXQuery	Knife	Lens	VXQuery
Apache _____ is a project that enables development and consumption of REST style web services.	Wives	Wink	Wig	Stratos	Wink
_____ is a log collection and correlation software with reporting and alarming functionalities.	Lucene	ALOIS	Imphal	Storm	ALOIS
_____ is a non-blocking, asynchronous, event driven high performance web framework.	AWS	AWF	AWT	ASW	AWF
_____ is the architectural center of Hadoop that allows multiple data processing engines.	YARN	Hive	Incubator	Chuckwa	YARN
YARN's dynamic allocation of cluster resources improves utilization over more static _____ rules used in early versions of Hadoop.	Hive	MapReduce	Imphala	Wink	MapReduce
The _____ is a framework-specific entity that negotiates resources from the ResourceManager	NodeManager	ResourceManager	ApplicationMaster	TaskMaster	ApplicationMaster
Apache Hadoop YARN stands for :	Yet Another Reserve Negotiator	Yet Another Resource Network	Yet Another Resource Negotiator	Yet Any Resource Negotiator	Yet Another Resource Negotiator
MapReduce has undergone a complete overhaul in hadoop :	0.21	0.23	0.24	0.26	0.23
The _____ is the ultimate authority that arbitrates resources among all the applications in the system.	NodeManager	ResourceManager	ApplicationMaster	dynamicMaster	ResourceManager

The _____ is responsible for allocating resources to the various running applications subject to familiar constraints of capacities, queues etc.	Manager	Master	Scheduler	Task	Scheduler
The CapacityScheduler supports _____ queues to allow for more predictable sharing of cluster resources.	Networked	Hierarchial	Partition	Unpartition	Hierarchial
ZooKeeper itself is intended to be replicated over a sets of hosts called :	chunks	ensemble	subdomains	unchunks	ensemble
_____ of the guarantee is provided by Zookeeper	Interactivity	Flexibility	Scalability	Reliability	Reliability
ZooKeeper is especially fast in _____ workloads	write	read-dominant	read-write	read	read-dominant
When a _____ is triggered the client receives a packet saying that the znode has changed.	event	watch	row	value	watch
The underlying client-server protocol has changed in version _____ of ZooKeeper.	2.0.0	3.0.0	4.0.0	6.0.0	3.0.0
A number of constants used in the client ZooKeeper API were renamed in order to reduce _____ collision	value	namespace	counter	signal	namespace
ZooKeeper allows distributed processes to coordinate with each other through registers, known as :	znodes	hnodes	vnodes	rnodes	znodes
The HDFS client software implements _____ checking on the contents of HDFS files.	metastore	parity	checksum	unparity	checksum
The _____ machine is a single point of failure for an HDFS cluster.	DataNode	NameNode	ActionNode	UnitNode	NameNode
The _____ and the EditLog are central data structures of HDFS.	DsImage	FsImage	FsImages	DsImages	FsImage
_____ support storing a copy of data at a particular instant of time.	Data Image	Datanots	Snapshots	DataFile	Snapshots

Automatic restart and _____ of the NameNode software to another machine is not supported.	failover	end	scalability	resource	failover
HDFS, by default, replicates each data block _____ times on different nodes and on at least ____ racks.	3,2	1,2	2,3	3,1	3,2
_____ stores its metadata on multiple disks that typically include a non-local file server.	DataNode	NameNode	ActionNode	UnitNode	NameNode
The HDFS file system is temporarily unavailable whenever the HDFS _____ is down.	DataNode	NameNode	ActionNode	UnitNode	NameNode
_____ is a platform for constructing data flows for extract, transform, and load (ETL) processing and analysis of large datasets.	Pig Latin	Oozie	Pig	Hive	Pig
_____ hides the limitations of Java behind a powerful and concise Clojure API for Cascading.	Scalding	HCatalog	Cascalog	Catalog	Cascalog
Hive also support custom extensions written in :	C#	Java	C	C++	Java
_____ is the most popular high-level Java API in Hadoop Ecosystem	Scalding	HCatalog	Cascalog	Cascading	Cascading
_____ is general-purpose computing model and runtime system for distributed data analytics.	Mapreduce	Drill	Oozie	ActionNode	Mapreduce
The Pig Latin scripting language is not only a higher-level data flow language but also has operators similar to :	SQL	JSON	XML	BSON	SQL
_____ jobs are optimized for scalability but not latency.	Mapreduce	Drill	Oozie	Hive	Hive
_____ is a framework for performing remote procedure calls and data serialization.	Drill	BigTop	Avro	Chukwa	Avro
Avro-backed tables can simply be created by using _____ in a DDL statement.	"STORED AS AVRO"	b) "STORED AS HIVE"	c) "STORED AS AVROHIVE"	"STORED AS SERDE"	"STORED AS AVRO"
Types that may be null must be defined as a _____ of that type and Null within Avro.	Union	Intersection	Set	Unset	Union

The files that are written by the _____ job are valid Avro files.	Avro	Map Reduce	Hive	Drill	Hive
Use _____ and embed the schema in the create statement.	schema.literal	schema.lit	row.literal	row.lit	schema.literal
_____ is interpolated into the quotes to correctly handle spaces within the schema.	$SCHEMA	$ROW	$SCHEMASPACES	$NAMESPACES	$SCHEMA
_____ was designed to overcome limitations of the other Hive file formats.	ORC	OPC	ODC	OLC	ORC
An ORC file contains groups of row data called :	postscript	stripes	script	scriptstart	stripes
The Mapper implementation processes one line at a time via _____ method.	map	reduce	mapper	reducer	map
The Hadoop MapReduce framework spawns one map task for each _____ generated by the InputFormat for the job.	OutputSplit	InputSplit	InputSplitStream	OutputSplitStream	InputSplit
Users can control which keys (and hence records) go to which Reducer by implementing a custom :	Partitioner	OutputSplit	Reporter	InputSplit	Partitioner
Applications can use the _____ to report progress and set application-level status messages	Partitioner	OutputSplit	Reporter	InputSplit	Reporter
The right level of parallelism for maps seems to be around _____ maps per-node	1-10	10-100	100-150	150-200	10-100

KARPAGAM ACADEMY OF HIGHER EDUCATION

Coimbatore – 641 021.

(For the Candidates admitted from 2016 onwards)

DEPARTMENT OF COMPUTER SCIENCE, CA & IT

UNIT - III : (Objective Type Multiple choice Questions each Question carries one Mark)

BIG DATA ANALYTICS [16CAP505D]

PART - A (Online Examination)

Questions	Opt1	opt2	opt3	opt4	KEY
_____ license is Hadoop distributed	Apache	Mozilla Public	Shareware	Commercial	Apache
Hadoop written in _____	Java	Perl	Java	Lua (programming	Java
Hadoop run on _____	Bare metal	Debian	Cross-platform	Unix-like	Cross-platform
Hadoop achieves reliability by replicating the data across multiple hosts, and hence does not require _____ storage on hosts.	RAID	Standard RAID levels	ZFS	Operating system	RAID
Above the file systems comes the _____ engine, which consists of one Job Tracker, to which client applications submit MapReduce jobs.	MapReduce	Google	Functional programming	Facebook	MapReduce
HDFS supports the _____ command to fetch Delegation Token and store it in a file on the local system.	fetdt	fetchdt	fsk	rec	fetchdt
In _____ mode, the NameNode will interactively prompt you at the command line about possible courses of action you can take to recover your data.	full	partial	recovery	commit	recovery
_____ command is used to copy file or directories recursively.	dtcp	distcp	dcp	distc	distcp

_____ mode is a Namenode state in which it does not accept changes to the name space.	Recover	Safe	Rollback	Partial	Rollback
_____ command is used to interact and view Job Queue information in HDFS.	queue	priority	dist	Recover	queue
_____ is used for the MapReduce job Tracker node	mradmin	tasktracker	jobtracker	Admin	jobtracker
A _____ serves as the master and there is only one NameNode per cluster.	Data Node	NameNode	Data block	Replication	NameNode
HDFS works in a _____ fashion.	master-worker	master-slave	worker/slave.	job tracker	master-worker
_____ NameNode is used when the Primary NameNode goes down.	Rack	Data	Secondary	primary	Secondary
_____ is the slave/worker node and holds the user data in the form of Data Blocks.	DataNode	NameNode	Data block	Replication	DataNode
HDFS provides a command line interface called _____ used to interact with HDFS.	"HDFS Shell"	"FS Shell"	"DFS Shell"	" HS Shell"	"FS Shell"
_____ method clears all keys from the configuration.	clear	addResource	getClass	Add	clear
_____ adds a configuration resource	addResource	setDeprecatedProperties	addDefaultResource	Resource	addResource
The LZO compression format is composed of approximately _____ blocks of compressed data.	128k	256k	24k	36k	256k
_____ is the default Partitioner for Mapreduce	MergePartitioner	HashedPartitioner	HashPartitioner	Hashing Partitioner	HashPartitioner
_____ partitions the key space	Partitioner	Compactor	Collector	full	Partitioner
_____ is a generalization of the facility provided by the MapReduce framework to collect data output by the Mapper or the Reducer	OutputCompactor	OutputCollector	InputCollector	Compactor	OutputCollector

_____ is the primary interface for a user to describe a MapReduce job to the Hadoop framework for execution.	JobConfig	JobConf	JobConfiguration	Configuration	JobConf
The _____ executes the Mapper/ Reducer task as a child process in a separate jvm.	JobTracker	TaskTracker	TaskScheduler	Scheduler	JobTracker
Maximum virtual memory of the launched child-task is specified using :	mapv	mapred	mapvim	mapr	mapred
_____ is percentage of memory relative to the maximum heapsize in which map outputs may be retained during the reduce.	mapred.job.shuffle.merge.percent	mapred.job.reduce.input.buffer.percen	mapred.inmem.merge.threshold	io.sort.factor	mapred.job.reduce.input.buffer.percen
In order to read any file in HDFS, instance of _____ is required.	filesystem	datastream	outstream	inputstream	filesystem
_____ is used to read data from bytes buffers .	write()	read()	readwrite()	rewrite()	write()
Interface _____ reduces a set of intermediate values which share a key to a smaller set of values.	Mapper	Reducer	Writable	Readable	Reducer
Reducer is input the grouped output of a _____	Mapper	Reducer	Writable	Readable	Mapper
Apache Hadoop's _____ provides a persistent data structure for binary key-value pairs.	GetFile	SequenceFile	Putfile	copyfile	SequenceFile
_____ formats of SequenceFile are present in Hadoop I/O	2	3	4	5	3
_____ format is more compression-aggressive	Partition Compressed	Record Compressed	Block-Compressed	Uncompressed	Block-Compressed
The _____ is a directory that contains two SequenceFile.	ReduceFile	MapperFile	MapFile	trackfile	MapFile

The _____ file is populated with the key and a LongWritable that contains the starting byte position of the record.	Array	Index	Immutable	mutable	Index
The _____ as just the value field append(value) and the key is a LongWritable that contains the record number, count + 1.	SetFile	ArrayFile	BloomMapFile	unsetfile	ArrayFile
. _____ data file takes is based on avro serializaton framework which was primarily created for hadoop.	Oozie	Avro	cTakes	Lucene	Avro
Avro schemas are defined with _____	JSON	XML	JAVA	HTML	JSON
_____ facilitates construction of generic data-processing systems and languages.	Untagged data	Dynamic typing	No manually-assigned field IDs	tagged data	Dynamic typing
With _____ we can store data and read it easily with various programming languages	Thrift	Protocol Buffers	Avro	Buffers	Avro
_____ are a way of encoding structured data in an efficient yet extensible format.	Thrift	Protocol Buffers	Avro	Buffers	Protocol Buffers
Thrift resolves possible conflicts through _____ of the field.	Name	Static number	UID	dynamic number	Static number
Avro is said to be the future _____ layer of Hadoop.	RMC	RPC	RDC	RBC	RPC
We can declare the schema of our data either in a _____ file.	JSON	XML	SQL	R	SQL
Avro supports _____ kinds of complex types.	3	4	6	7	7
_____ are encoded as a series of blocks.	Arrays	Enum	Unions	Maps	Arrays
_____ instances are encoded using the number of bytes declared in the schema.	Fixed	Enum	Unions	Maps	Fixed
_____ permits data written by one system to be efficiently sorted by another system.	Complex Data type	Order	Sort Order	Unsort Order	Sort Order
_____ are used between blocks to permit efficient splitting of files for MapReduce processing.	Codec	Data Marker	Syncronization markers	Unsyncronization markers	Syncronization markers
The _____ codec uses Google's Snappy compression library.	null	snappy	deflate	delete	snappy

Avro messages are framed as a list of _____	buffers	frames	rows	columns	frames
_____ node is responsible for executing a Task assigned to it by the JobTracker	MapReduce	Mapper	TaskTracker	JobTracker	TaskTracker
_____ function is responsible for consolidating the results produced by each of the Map() functions/tasks.	Reduce	Map	Reducer	Unmap	Reduce
_____ is a utility which allows users to create and run jobs with any executable as the mapper and/or the reducer.	Hadoop Strdata	Hadoop Streaming	Hadoop Stream	Hadoopdata	Hadoop Streaming
The number of maps is usually driven by the total size of _____	inputs	outputs	tasks	process	inputs
Running a _____ program involves running mapping tasks on many or all of the nodes in our cluster.	MapReduce	Map	Reducer	Redcuce	MapReduce
Apache _____ is a data repository containing device information, images and other relevant information for all sorts of mobile devices.	DirectMemory	Directory	DeviceMap	Drill	DeviceMap
_____ is a secure and highly scalable microsharing and micromessaging platform.	ESME	Directory	Empire-db	Entity	ESME
_____ framework is used for building and consuming network services	ESME	DirectoryMap	Empire-db	Etch	Etch
_____ is an open source system for expressive, declarative, fast, and efficient data analysis.	Flume	Flink	Flex	ESME	Flink

KARPAGAM ACADEMY OF HIGHER EDUCATION

Coimbatore – 641 021.

(For the Candidates admitted from 2016 onwards)

DEPARTMENT OF COMPUTER SCIENCE, CA & IT

UNIT - IV : (Objective Type Multiple choice Questions each Question carries one Mark)

BIG DATA ANALYTICS [16CAP505D]

PART - A (Online Examination)

Questions	Opt1	opt2	opt3	opt4	KEY
_____ commodity Hardware in Hadoop	Very	Industry	Discarded	Low specifications	Low
_____ are NOT big data problem(s)	Parsing 5	Processing IPL	Processing online	Parsing	Processing
_____"Velocity" in Big Data mean	Speed of input data generation	Speed of individual machine processors	Speed of ONLY storing data	Speed of storing and processing data	Speed of storing and processing data
The term Big Data first originated from:	Stock Markets Domain	Banking and Finance Domain	Genomics and Astronomy Domain	Social Media Domain	Genomics and Astronomy Domain
Batch Processing instance is NOT an example of _____	Processing 10 GB sales data every 6 hours	Processing flights sensor data	Web crawling app	Trending topic analysis of tweets for last 15 minutes	Trending topic analysis of tweets for last 15 minutes

_____ are example(s) of Real Time Big Data Processing	Complex Event Processing (CEP) platforms & Bank fraud transactions detection	Stock market data analysis	transactions detection	Stock transactions	Complex Event Processing (CEP) platforms & Bank fraud transactions detection
Sliding window operations typically fall in the category of_____.	OLTP Transactions	Big Data Batch Processing	Big Data Real Time Processing	Small Batch Processing	Big Data Real Time Processing
HBase used as _____	Tool for Random and Fast Read/Write operations in Hadoop	Faster Read only query engine in Hadoop	MapReduce alternative in Hadoop	Fast MapReduce layer in Hadoop	Tool for Random and Fast Read/Write operations in Hadoop
Hive used as _____	Hadoop query engine	MapReduce wrapper	Hadoop SQL interface	Hadoop	hadoop
_____ are NOT true for Hadoop	It's a tool for Big Data analysis	It supports structured and unstructured data analysis	It aims for vertical scaling out/in scenarios	It supports structured and semistructured data analysis	It supports structured and unstructured data analysis
_____ are the core components of Hadoop	HDFS	Map Reduce	HBase	Hive	Map Reduce
Hadoop is open source_____	ALWAYS True	True only for Apache Hadoop	True only for Apache and Cloudera Hadoop	ALWAYS False	True only for Apache Hadoop
Hive can be used for real time queries _____	TRUE	FALSE	True if data set is small	True for some distributions	FALSE
_____ is the default HDFS block size	32 MB	64 KB	128 KB	64 MB	64 MB
_____ is the default HDFS replication factor	4	1	3	2	3

_____ is NOT a type of metadata in NameNode	List of files	Block locations of files	No. of file records	File access control information	No. of file records
Which of the following is/are correct	NameNode is the SPOF in Hadoop 1.x	NameNode is the SPOF in Hadoop 2.x	NameNode keeps the image of the file system also	SPOF	NameNode keeps the image of the file system also
The mechanism used to create replica in HDFS is_____.	Gossip protocol	Replicate protocol	HDFS protocol	Store and Forward protocol	HDFS protocol
NameNode tries to keep the first copy of data nearest to the client machine.	ALWAYS true	ALWAYS False	True if the client machine is the part of the cluster	True if the client machine is not the part of the cluster	True if the client machine is the part of the cluster
HDFS data blocks can be read in parallel.	TRUE	FALSE	data	read & write	TRUE
_____ is HDFS replication factor controlled	mapred-site.xml	yarn-site.xml	core-site.xml	hdfs-site.xml	hdfs-site.xml
_____ Hadoop config files is used to define the heap size	hdfs-site.xml	core-site.xml	hadoop-env.sh	Slaves	hadoop-env.sh
_____ is not a valid Hadoop config file	mapred-site.xml	hadoop-site.xml	core-site.xml	Masters	hadoop-site.xml
Read the statement: NameNodes are usually high storage machines in the clusters.	True	False	Depends on cluster size	True if co-located with Job tracker	False
From the options listed below, select the suitable data sources for flume.	Publicly open web sites	Local data folders	Remote web servers	web services	Remote web servers
Read the statement and select the correct options: distcp command ALWAYS needs fully qualified hdfs paths.	True	False	True, if source and destination are in same cluster	False, if source and destination are in same cluster	True
_____ statement(s) are true about distcp command?	It invokes MapReduce in background	It invokes MapReduce if source and destination are in same cluster	It can't copy data from local folder to hdfs folder	You can't overwrite the files through distcp command	It invokes MapReduce in background

_____ is NOT the component of Flume	Sink	Database	Source	Channel	Database
_____ is the correct sequence of MapReduce flow	Combine - Reduce - Map	Map -Combine - Reduce	Reduce -Combine - Map	Map -Reduce - Combine	Reduce - Combine -Map
_____ can be used to control the number of part files in a map reduce program output directory	Number of Mappers	Number of Reducers	Counter	Partitioner	Number of Reducers
_____ operations can't use Reducer as combiner also	Group by Minimum	Group by Maximum	Group by Count	Group by Average	Group by Average
_____ is/are true about combiners	Combiners can be used for mapper only job	Combiners can be used for any Map Reduce operation	Mappers can be used as a combiner class	Combiners are primarily aimed to improve Map Reduce performance	Combiners are primarily aimed to improve Map Reduce performance
Reduce side join is useful for _____	Very large datasets	Very small data sets	One small and other big data sets	One big and other small datasets	Very large datasets
Distributed Cache can be used in _____	Mapper phase only	Reducer phase only	In either phase, but not on both sides simultaneously	In either phase	In either phase
Counters persist the data on hard disk.	True	False			False
_____ is optimal size of a file for distributed cache	<=10 MB	>=250 MB	<=100 MB	<=35 MB	<=100 MB
Number of mappers is decided by the _____	Mappers specified by the programmer	Available Mapper slots	Available heap memory	Input Splits	Input Splits
_____ type of joins can be performed in Reduce side join operation	Equi Join	Left Outer Join	Right Outer Join	Full Outer Join	All of the above
_____ is an upper limit for counters of a Map Reduce job	~5s	~15	~150	~50	~50

_____ class is responsible for converting inputs to key-value Pairs of Map Reduce	FileInputFormat	InputSplit	RecordReader	Mapper	RecordReader
_____ writables can be used to know value from a mapper/reducer	Text	IntWritable	Nullwritable	String	Nullwritable
Distributed cache files can't be accessed in Reducer.	True	False			False
Only one distributed cache file can be used in a Map Reduce job.	True	False			False
A Map reduce job can be written in:	Java	Ruby	Python	Any Language which can read from input stream	Any Language which can read from input stream
Pig is a _____	Programming Language	Data Flow Language	Query Language	Database	Data Flow Language
Pig is good for _____	Data Factory operations	Data Warehouse operations	Implementing complex SQLs	Creating multiple datasets from a single large dataset	Both (and (
Pig can be used for real-time data updates.	True	False			False
Pig jobs have the same run time as the native Map Reduce jobs.	True	False			False
_____ is the correct representation to access ''Skill" from the Bag {'Skills',55, ('Skill', 'Speed'), {2, ('San', 'Mateo')}}	$3.$1	$3.$0	$2.$0	$2.$1	$3.$1
Replicated joins are useful for dealing with data skew.	True	False			False
Maximum size allowed for small dataset in replicated join is _____	10KB	10 MB	100 MB	500 MB	100 MB
Parameters could be passed to Pig scripts from _____	Parent Pig Scripts	Shell Script	Command Line	Configuration File	Command Line

The schema of a relation can be examined through _____	ILLUSTRATE	DESCRIBE	DUMP	EXPLAIN	DESCRIBE
DUMP Statement writes the output in a file.	True	False			False
Data can be supplied to PigUnit tests from _____	HDFS Location	Within Program	HIVE	HBASE	HDFS Location
_____ constructs are valid Pig Control Structures	If-else	For Loop	Until Loop	Loop	Loop
_____ is the return data type of Filter UDF	String	Integer	Boolean	None of the above	Boolean
UDFs can be applied only in FOREACH statements in Pig.	True	False			True
_____ of the following are not possible in Hive	Creating Tables	Creating Indexes	Creating Synonym	Writing Update Statements	Creating Synonym
_____ works well with Avro	Lucene	kafka	MapReduce	DesignPattern	MapReduce

KARPAGAM ACADEMY OF HIGHER EDUCATION

Coimbatore – 641 021.

(For the Candidates admitted from 2016 onwards)

DEPARTMENT OF COMPUTER SCIENCE, CA & IT

UNIT - V : (Objective Type Multiple choice Questions each Question carries one Mark)

BIG DATA ANALYTICS [16CAP505D]

PART - A (Online Examination)

Questions	Opt1	opt2	opt3	opt4	KEY
_____will initiate the mapper	Task tracker	Job tracker	Combiner	Reducer	Task tracker
Categorize the following to the following datatype	– Semi-	PDF Files , Text	Unstructured	systems (DB, CRM) –	Unstructured
Hadoop is a framework that allows the distributed processing of _____	Small Data Sets	Semi-Large Data Sets	Large Data Sets	Large and Small Data sets	Large Data Sets
Sqoop ingest data from _____	Linux File Directory	Oracle &MySQL	HBase	MySQL	Oracle &MySQL
Identify the batch processing scenarios from following:	Sliding Window Averages Job	Facebook Comments Processing Job	Inventory Dynamic Pricing Job	Fraudulent Transaction Identification Job	Inventory Dynamic Pricing Job
_____ is true about Name Node	It is the Master Machine of the Cluster	It is Name Node that can store user data	Name Node is a storage heavy machine	Name Node can be replaced by any Data Node Machine	It is the Master Machine of the Cluster
_____are NOT metadata items	List of HDFS files	HDFS block locations	Replication factor of files	File Records distribution	File Records distribution
_____decides number of Mappers for a MapReduce job	File Location	mapred.map.tasks parameter	Input file size	Input Splits	Input file size
Name Node monitors block replication process of ___	TRUE	FALSE	replication	data node	FALSE

_____ true for Hadoop Pseudo Distributed Mode	It runs on multiple machines	Runs on multiple machines without any daemons	Runs on Single Machine with all daemons	Runs on Single Machine without all daemons	Runs on Single Machine with all daemons
Which of following statement(s) are correct?	Master and slaves files are optional in Hadoop 2.x	Master file has list of all name nodes	Core-site has hdfs and MapReduce related common properties	hdfs-site file is now deprecated in Hadoop 2.x	Core-site has hdfs and MapReduce related common properties
_____ is true for Hive	Hive is the database of Hadoop	Hive supports schema checking	Hive doesn't allow row level updates	Hive can replace an OLTP system	Hive doesn't allow row level updates
_____ is the highest level of Data Model in Hive	Table	View	Database	Partitions	Database
Hive queries response time is in order of _____	Hours at least	Minutes at least	Seconds at least	Milliseconds at least	Seconds at least
Managed tables in Hive _____	Can load the data only from HDFS	Can load the data only from local file system	Are useful for enterprise wide data	Are Managed by Hive for their data and metadata	Are Managed by Hive for their data and metadata
Partitioned tables in Hive _____	Are aimed to increase the performance of the queries	Modify the underlying HDFS structure	Are not useful if the filter columns for query are different from the partition columns	HDFS	Are aimed to increase the performance of the queries
Hive UDFs can only be written in Java	True	False			False
Hive can load the data from _____	Local File system	HDFS File system	Output of a Pig Job	File system	HDFS File system
HBase is a key/value store. Specifically it is _____	Sparse	Multi-dimensional	Distributed	Consistent	Multi-dimensional
_____ is the outer most part of HBase data model	Database	Table	Row key	Column family	Database

Which of the following is/are true:	HBase table has fixed number of Column families	HBase table has fixed number of Columns	HBase doesn't allow row level updates	HBase access HDFS data	HBase table has fixed number of Column families
Data can be loaded in HBase from Pig using _____	PigStorage	SqoopStorage	BinStorage	HbaseStorage	HbaseStorage
Sqoop can load the data in HBase _____	True	False			True
_____ APIs can be used for exploring HBase tables	HBaseDescriptor	HBaseAdmin	Configuration	HTable	HTable
_____ tables in HBase holds the region to key mapping	ROOT	.META.	MAP	REGIONS	.META.
_____ is the data type of version in HBase	INT	LONG	STRING	DATE	LONG
_____ is the data type of row key in HBase	INT	STRING	BYTE	BYTE[]	BYTE[]
HBase first reads the data from _____	Block Cache	Memstore	HFile	WAL	Memstore
The High availability of Namenode is achieved in HDFS2.x using _____	Polled Edit Logs	Synchronized Edit Logs	Shared Edit Logs	Edit Logs Replacement	Shared Edit Logs
The application master monitors all Map Reduce applications in the cluster _____	True	False			False
HDFS Federation is useful for the cluster size of ____	>500 nodes	>900 nodes	> 5000 nodes	> 3500 nodes	> 5000 nodes
Hive managed tables stores the data in ____	Local Linux path	Any HDFS path	HDFS warehouse path	None of the above	HDFS warehouse path
On dropping managed tables, Hive _____	Retains data, but deletes metadata	Retains metadata, but deletes data	Drops both, data and metadata	Retains both, data and metadata	Drops both, data and metadata
Managed tables don't allow loading data from other tables.	True	False			False

External tables can load the data from warehouse Hive directory.	True	False			True
On dropping external tables, Hive _____	Retains data, but deletes metadata	Retains metadata, but deletes data	Drops both, data and metadata	Retains both, data and metadata	Retains data, but deletes metadata
Partitioned tables can't load the data from normal (partitione tables	True	False			False
The partitioned columns in Hive tables are _____	Physically present and can be accessed	Physically absent but can be accessed	Physically present but can't be accessed	Physically absent and can't be accessed	Physically absent but can be accessed
Hive data models represent _____	Table in Metastore DB	Table in HDFS	Directories in HDFS	None of the above	Directories in HDFS
_____ is the earliest point at which the reduce method of a given Reducer can be called.	As soon as at least one mapper has finished processing its input split.	As soon as a mapper has emitted at least one record.	Not until all mappers have finished processing all records.	It depends on the InputFormat used for the job.	Not until all mappers have finished processing all records.
Apache Cassandra is a massively scalable open source __	SQL	NoSQL	NewSQL	Oracle	NoSQL
Cassandra uses a protocol called _____ to discover loc	gossip	intergos	goss	internalgos	gossip
A _____ determines which data centers and racks n	Client reque	Snitch	Partitioner	non-partitioner	Snitch
User accounts may be altered and dropped using the ____	Hive	Cassandra	Sqoop	Lucene	Cassandra
Authorization capabilities for Cassandra use the familiar	COMMIT	GRANT	ROLLBACK	TRIGGER	GRANT
Client-to-node encryption protects data in flight from clie	SSL	SSH	SSN	SLS	SSL
Using _____ file means you don't have to overrid	qlshrc	cqshrc	cqlshrc	qlc	cqlshrc
Internal authentication stores usernames and bcrypt-hashe	system_auth	system_auth.cred	system.credentials	sys_auth.credentials	system_auth.credentials
A _____ grants initial permissions, and subsequently	keyspace	superuser	sudouser	unuser	superuser

_____ is one of many possible IAuthorizer implem	CassandraA	CassandraAutho	CassAuthorizer	Authorizer	CassandraAuthorizer
Cassandra creates a _____ for each table, which al	directory	subdirectory	domain	path	subdirectory
When _____ contents exceed a configurable thres	subtable	memtable	intable	memorytable	memtable
Data in the commit log is purged after its corresponding d	SSHables	SSTable	Memtables	SLSTable	SSTable
For each SSTable, Cassandra creates _____ index .	memory	partition	in memory	syncronize	partition
Cassandra marks data to be deleted using :	tombstone	combstone	tenstone	stone	tombstone
Tombstones exist for a configured time period defined by	gc_grace_m	gc_grace_time	gc_grace_seconds	gc_grace_hours	gc_grace_seconds
_____ is a Cassandra feature that optimizes the clust	Hinted hand	Hinted handoff	Tombstone	Hinted tomb	Hinted handoff
Cassandra searches the _____ to determine the app	partition rec	partition summar	partition search	partition file	partition summary
You configure sample frequency by changing the _____	index_time	index_interval	index_secs	indexed	index_interval
The compression offset map grows to ____ GB per teraby	1-3	10-16	20-22	0-1	1-3

Reg. No.....

[15CAP505D]

KARPAGAM UNIVERSITY

Karpagam Academy of Higher Education
(Established Under Section 3 of UGC Act 1956)
COIMBATORE – 641 021
(For the candidates admitted from 2015 onwards)

MCA DEGREE EXAMINATION, NOVEMBER 2017

Fifth Semester

COMPUTER APPLICATIONS

BIG DATA ANALYTICS

Time: 3 hours Maximum : 60 marks

PART – A (20 x 1 = 20 Marks) (30 Minutes)
(Question Nos. 1 to 20 Online Examinations)

PART B (5 x 6 = 30 Marks)
Answer ALL the Questions

21. a. Explain in detail about big data management architecture.
Or
b. Explain about structured data.

22. a. Explain about basics of virtualization.
Or
b. Explain in detail about the importance of virtualization to big data.

23. a. Explain in detail about Hadoop Map-Reduce.
Or
b. Explain in detail about how to build a big data foundation with the Hadoop ecosystem.

24. a. Explain in detail about big data analytics examples.
Or
b. Explain about the Unstructured data and understanding text analytics.

25. a. Explain in detail about the understanding the ELT.
Or
b. Explain about the streaming data and complex event processing.

PART C (1 x 10 = 10 Marks)
(Compulsory)

26. Explain about managing resources and applications with Hadoop.

Reg No. ……..……..

[16CAP505D]

KARPAGAM ACADEMY OF HIGHER EDUCATION

(Established Under Section 3 of UGC Act 1956)

COIMBATORE – 641 021

MCA Degree Examination

(For the candidates admitted from 2016 onwards)

Fifth Semester

First Internal Exam August 2018

BIG DATA ANALYTICS

Duration: 2 Hrs **Maximum Marks: 50 Marks**

Date & Session: 17.08.2018 & FN **Class: III MCA**

Part - A (20 X 1 = 20 Marks)
(Answer all the Questions)

1. _____ is not a single technology but a combination of old and new technologies that helps companies gain actionable insight.
   a) Data Management b) Big Data c) Dataset d) Dara
2. The term _____ data generally refers to data that has a defined length and format.
   a) Structured b) Unstructured c) Sources d) Daraset
3. _____ data is data that does not follow a specified format
   a) Structured b) Unstructured c) Sources d) Daraset
4. _____ class adds HBase configuration files to its object.
   a) Configuration b) Collector c) Component d) Compiler
5. The _____ class provides the getValue() method to read the values from its instance.
   a) Get b) Result c) Put d) Value
6. _____ communicate with the client and handle data-related operations.
   a) Master Server b) Region Server c) Htable d) Rtable
7. Above the file systems comes the _____ engine, which consists of one Job Tracker, to which client applications submit MapReduce jobs.
   a) MapReduce b) Google
   c) Functional programming d) Facebook
8. HDFS supports the _____ command to fetch Delegation Token and store it in a file on the local system.
   a) fetdt b) fetchdt c) fsk d) rec
9. In _____ mode, the NameNode will interactively prompt you at the command line about possible courses of action you can take to recover your data.
   a) full b) partial c) recovery d) commit
10. _____ command is used to copy file or directories recursively.
    a) dtcp b) distcp c) dcp d) distc
11. The data is increasing at a very fast rate and it is estimated that the volume of data
    a) variety b) volume c) velocity d) veracity
12. Data which can be saved in tables are structured data like the transaction data of the bank.
    a) variety b) volume c) velocity d) veracity
13. The amount of data which we deal with is of very large size of Peta bytes.
    a) variety b) volume c) velocity d) veracity
14. _____ is an open source framework from Apache and is used to store process and analyze data which are very huge in volume
    a) HDFS b) hadoop c) map reduce d) File system
15. _____ is a framework which helps Java programs to do the parallel computation on data using key value pair.
    a) HDFS b) hadoop c) map reduce d) File system

16. _____ states that the files will be broken into blocks and stored in nodes over the distributed architecture.
    a) HDFS b) hadoop c) map reduce d) File system
17. _____ is a foundational technology applicable to the implementation of both cloud computing and big data.
    a) Address space b) Virtualization c) isolation d) Big data
18. _____ virtualization provides an efficient way to manage applications in context with customer demand.
    a) Application b) Server c) Network d) Processor
19. _____ virtualization provides an efficient way to use networking as a pool of connection resources
    a) Application b) Server c) Network d) Processor
20. _____ virtualization helps to optimize the processor and maximize performance.
    a) Application b) Server c) Network d) Processor

Part - A (3 X 2 = 6 Marks)
(Answer all the Questions)

21. Define Big data
22. What is the process of Virtualization?
23. What is meant by hadoop?

Part - B (3 X 8 =24 Marks)
(Answer all the Questions)

24. (a) Explain the Big data Management Architecture.
    (OR)
    (b) Explain the following
    (i) Structured Data (ii) Unstructured Data
25. (a) Discuss in detail about different types of Virtualization.
    (OR)
    (b) Explain about the implementation of virtualization to work with Big Data.
26. (a) Explain about Hadoop Distributed File system
    (OR)
    (b) Describe the Process of Map Reduce framework

Reg No. ……..……..

[16CAP505D]

KARPAGAM ACADEMY OF HIGHER EDUCATION

(Established Under Section 3 of UGC Act 1956)

COIMBATORE – 641 021

MCA Degree Examination

(For the candidates admitted from 2016 onwards)

Fifth Semester

First Internal Exam August 2018

BIG DATA ANALYTICS

Duration: 2 Hrs **Maximum Marks: 50 Marks**

Date & Session: 17.08.2018 & FN **Class: III MCA**

Part - A (20 X 1 = 20 Marks)
(Answer all the Questions)

1. _____ is not a single technology but a combination of old and new technologies that helps companies gain actionable insight.
   a) Data Management **b) Big Data** c) Dataset d) Dara
2. The term _____ data generally refers to data that has a defined length and format.
   a) Structured b) Unstructured c) Sources d) Daraset
3. _____ data is data that does not follow a specified format
   a) Structured **b) Unstructured** c) Sources d) Daraset
4. _____ class adds HBase configuration files to its object.
   a) Configuration b) Collector **c) Component** d) Compiler
5. The _____ class provides the getValue() method to read the values from its instance.
   a) Get b) Result c) Put d) Value
6. _____ communicate with the client and handle data-related operations.
   a) Master Server b) Region Server c) Htable d) Rtable
7. Above the file systems comes the _____ engine, which consists of one Job Tracker, to which client applications submit MapReduce jobs.
   a) MapReduce **b) Google**
   c) Functional programming d) Facebook
8. HDFS supports the _____ command to fetch Delegation Token and store it in a file on the local system.
   a) fetdt **b) fetchdt** c) fsk d) rec
9. In _____ mode, the NameNode will interactively prompt you at the command line about possible courses of action you can take to recover your data.
   a) full b) partial **c) recovery** d) commit
10. _____ command is used to copy file or directories recursively.
    a) dtcp **b) distcp** c) dcp d) distc
11. The data is increasing at a very fast rate and it is estimated that the volume of data
    a) variety b) volume **c) velocity** d) veracity
12. Data which can be saved in tables are structured data like the transaction data of the bank.
    a) variety **b) volume** c) velocity d) veracity
13. The amount of data which we deal with is of very large size of Peta bytes.
    a) variety b) volume c) velocity d) veracity
14. _____ is an open source framework from Apache and is used to store process and analyze data which are very huge in volume
    a) HDFS **b) hadoop** c) map reduce d) File system
15. _____ is a framework which helps Java programs to do the parallel computation on data using key value pair.
    a) HDFS b) hadoop **c) map reduce** d) File system

16. _____ states that the files will be broken into blocks and stored in nodes over the distributed architecture.
a) HDFS b) hadoop c) map reduce d) File system
17. _____ is a foundational technology applicable to the implementation of both cloud computing and big data.
a) Address space **b) Virtualization** c) isolation d) Big data
18. _____ virtualization provides an efficient way to manage applications in context with customer demand.
a) Application b) Server c) Network d) Processor
19. _____ virtualization provides an efficient way to use networking as a pool of connection resources
a) Application b) Server **c) Network** d) Processor
20. _____ virtualization helps to optimize the processor and maximize performance.
a) Application b) Server c) Network **d) Processor**

Part - A (3 X 2 = 6 Marks)
(Answer all the Questions)

21. Define Big data
Answer:
- Big data is defined as any kind of data source that has at least three shared characteristics:
  - Extremely large *Volumes* of data
  - Extremely high *Velocity* of data
  - Extremely wide *Variety* of data
- Big data is important because it enables organizations to gather, store, manage, and manipulate vast amounts data at the right speed, at the right time, to gain the right insights. But before we delve into the details of big data, it is important to look at the evolution of data management and how it has led to big data. Big data is not a stand-alone technology; rather, it is a combination of the last 50 years of technology evolution.

22. What is the process of Virtualization?
Answer:
- Virtualization is a foundational technology applicable to the implementation of both cloud computing and big data. It provides the basis for many of the platform attributes required to access, store, analyze, and manage the distributed computing components in big data environments. Virtualization — the process of using computer resources to imitate other resources — is valued for its capability to increase IT resource utilization, efficiency, and scalability.
- One primary application of virtualization is server consolidation, which helps organizations increase the utilization of physical servers and potentially save on infrastructure costs. However, you find many benefits to virtualization. Companies that initially focused solely on server virtualization are now recognizing that it can be applied across the entire IT infrastructure, including software, storage, and networks.

23. What is meant by hadoop?

Answer:

- Hadoop was originally built by a Yahoo! engineer named Doug Cutting and is now an open source project managed by the Apache Software Foundation. It is made available under the Apache License v2.0.
- Hadoop is a fundamental building block in our desire to capture and process big data. Hadoop is designed to parallelize data processing across computing nodes to speed computations and hide latency. At its core, Hadoop has two primary components:
- **Hadoop Distributed File System:** A reliable, high-bandwidth, low-cost, data storage cluster that facilitates the management of related files across machines.
- **Map Reduce engine:** A high-performance parallel/distributed data- processing implementation of the Map Reduce algorithm. Hadoop is designed to process huge amounts of structured and unstructured data (terabytes to petabytes) and is implemented on racks of commodity servers as a Hadoop cluster. Servers can be added or removed from the cluster dynamically because Hadoop is designed to be "self-healing." In other words, Hadoop is able to detect changes, including failures, and adjust to those changes and continue to operate without interruption.

Part - B (3 X 8 =24 Marks)
(Answer all the Questions)

24. (a) Explain the Big data Management Architecture.

Answer:

DEFINING BIG DATA

- Big data is not a single technology but a combination of old and new technologies that helps companies gain actionable insight. Therefore, big data is the capability to manage a huge volume of disparate data, at the right speed, and within the right time frame to allow real-time analysis and reaction. Big data is typically broken down by three characteristics:
  - **Volume:** How much data
  - **Velocity:** How fast that data is processed
  - **Variety:** The various types of data
- Although it's convenient to simplify big data into the three Vs, it can be misleading and overly simplistic. For example, you may be managing a relatively small amount of very disparate, complex data or you may be processing a huge volume of very simple data. That simple data may be all structured or all unstructured. Even more important is the fourth V: veracity. How accurate is that data in predicting business value? Do

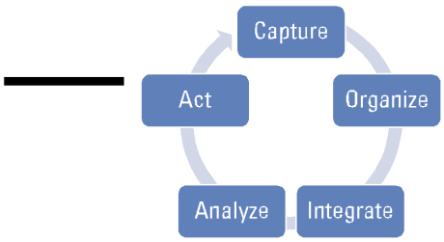
the results of a big data analysis actually make sense?

- It is critical that you don't underestimate the task at hand. Data must be able to be verified based on both accuracy and context. An innovative business may want to be able to analyze massive amounts of data in real time to quickly assess the value of that customer and the potential to provide additional offers to that customer. It is necessary to identify the right amount and types of data that can be analyzed to impact business outcomes. Big data incorporates all data, including structured data and unstructured data from e-mail, social media, text streams, and more. This kind of data management requires that companies leverage both their structured and unstructured data.

BUILDING A SUCCESSFUL BIG DATA MANAGEMENT ARCHITECTURE

- We have moved from an era where an organization could implement a data- base to meet a specific project need and be done. But as data has become the fuel of growth and innovation, it is more important than ever to have an underlying architecture to support growing requirements.
- **Beginning with capture, organize, integrate, analyze, and act**

- Before we delve into the architecture, it is important to take into account the functional requirements for big data. Figure 1-1 illustrates that data must first be captured, and then organized and integrated. After this phase is successfully implemented, data can be analyzed based on the problem being addressed. Finally, management takes action based on the outcome of that analysis. For example, Amazon.com might recommend a book based on a past purchase or a customer might receive a coupon for a discount for a future purchase of a related product to one that was just purchased.



- Although this sounds straightforward, certain nuances of these functions are complicated. Validation is a particularly important issue. If your organization is combining data sources, it is critical that you have the ability to validate that these sources make sense when combined. Also, certain data sources may contain sensitive information, so you must implement sufficient levels of security and governance.

(OR)

(b) Explain the following

(i) Structured Data

Answer:

DEFINING STRUCTURED DATA

- The term structured data generally refers to data that has a defined length and format. Examples of structured data include numbers, dates, and groups of words and numbers called strings (for example, a customer's name, address, and so on). Most experts agree that this kind of data accounts for about 20 percent of the data that is out there. Structured data is the data that you're probably used to dealing with. It's usually stored in a database. You can query it using a language like structured query language (SQL), which we discuss later in the "Defining Unstructured Data" section.
- Your company may already be collecting structured data from "traditional" sources. These might include your customer relationship management (CRM) data, operational enterprise resource planning (ERP) data, and financial data. Often these data elements are integrated in a data warehouse for analysis.

Exploring sources of big structured data

- Although this might seem like business as usual, in reality, structured data is taking on a new role in the world of big data. The evolution of technology provides newer sources of structured data being produced — often in real time and in large volumes. The sources of data are divided into two categories:
- **Computer- or machine-generated:** Machine-generated data generally refers to data that is created by a machine without human intervention.
- **Human-generated:** This is data that humans, in interaction with computers, supply.
- Some experts argue that a third category exists that is a hybrid between machine and human. Here though, we're concerned with the first two categories.
- Machine-generated structured data can include the following:
- **Sensor data:** Examples include radio frequency ID (RFID) tags, smart meters, medical devices, and Global Positioning System (GPS) data. For example, RFID is rapidly becoming a popular technology. It uses tiny computer chips to track items at a distance. An example of this is tracking containers of produce from one location to another. When information is transmitted from the receiver, it can go into a server and then be analyzed. Companies are interested in this for supply chain management and inventory control. Another example of sensor data is smart

phones that contain sensors like GPS that can be used to understand customer behavior in new ways.

- **Web log data:** When servers, applications, networks, and so on operate, they capture all kinds of data about their activity. This can amount to huge volumes of data that can be useful, for example, to deal with service-level agreements or to predict security breaches.
- **Point-of-sale data:** When the cashier swipes the bar code of any product that you are purchasing, all that data associated with the product is generated. Just think of all the products across all the people who purchase them and you can understand how big this data set can be.
- **Financial data:** Lots of financial systems are now programmatic; they are operated based on predefined rules that automate processes. Stock- trading data is a good example of this. It contains structured data such as the company symbol and dollar value. Some of this data is machine generated, and some is human generated.
- Examples of structured human-generated data might include the following:
- **Input data:** This is any piece of data that a human might input into a computer, such as name, age, income, non-free-form survey responses, and so on. This data can be useful to understand basic customer behavior.
- **Click-stream data:** Data is generated every time you click a link on a website. This data can be analyzed to determine customer behavior and buying patterns.
- **Gaming-related data:** Every move you make in a game can be recorded. This can be useful in understanding how end users move through a gaming portfolio.

(ii) Unstructured Data

Answer:

DEFINING UNSTRUCTURED DATA

- Unstructured data is data that does not follow a specified format. If 20 percent of the data available to enterprises is structured data, the other 80 percent is unstructured. Unstructured data is really most of the data that you will encounter. Until recently, however, the technology didn't really support doing much with it except storing it or analyzing it manually.

Exploring Sources of Unstructured Data

- Unstructured data is everywhere. In fact, most individuals and organizations conduct their lives around unstructured

data. Just as with structured data, unstructured data is either machine generated or human generated.

- Here are some examples of machine-generated unstructured data:
- **Satellite images:** This includes weather data or the data that the government captures in its satellite surveillance imagery. Just think about Google Earth, and you get the picture (pun intended).
- **Scientific data:** This includes seismic imagery, atmospheric data, and high energy physics.
- **Photographs and video:** This includes security, surveillance, and traffic video.
- **Radar or sonar data:** This includes vehicular, meteorological, and oceanographic seismic profiles.
- The following list shows a few examples of human-generated unstructured data:
- **Text internal to your company:** Think of all the text within documents, logs, survey results, and e-mails. Enterprise information actually represents a large percent of the text information in the world today.
- **Social media data:** This data is generated from the social media plat- forms such as YouTube, Facebook, Twitter, LinkedIn, and Flicker.
- **Mobile data:** This includes data such as text messages and location information.
- **Website content:** This comes from any site delivering unstructured content, like YouTube, Flickr, or Instagram.

25. (a) Discuss in detail about different types of Virtualization.

Answer:

THE BASICS OF VIRTUALIZATION

- Virtualization separates resources and services from the underlying physical delivery environment, enabling you to create many virtual systems within a single physical system. Figure 5-1 shows a typical virtualization environment.
- One of the primary reasons that companies have implemented virtualization is to improve the performance and efficiency of processing of a diverse mix of workloads. Rather than assigning a dedicated set of physical resources to each set of tasks, a pooled set of virtual resources can be quickly allocated as needed across all workloads.

Reliance on the pool of virtual resources allows companies to improve latency. This increase in service delivery speed and efficiency is a function of the distributed nature of virtualized environments and helps to improve overall time-to-value.

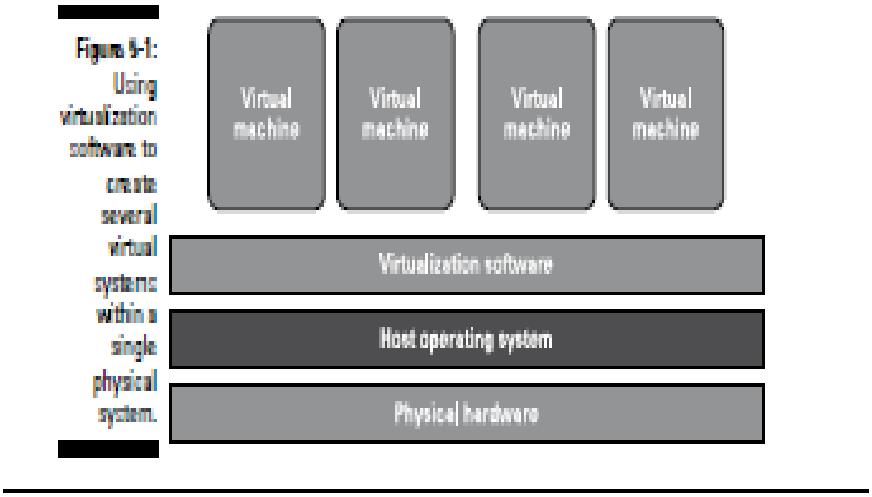


Figure 5-1: Using virtualization software to create several virtual systems within a single physical system.

- Using a distributed set of physical resources, such as servers, in a more flexible and efficient way delivers significant benefits in terms of cost savings and improvements in productivity. The practice has several benefits, including the following:
    - Virtualization of physical resources (such as servers, storage, and networks) enables substantial improvement in the utilization of these resources.
    - Virtualization enables improved control over the usage and performance of your IT resources.
    - Virtualization can provide a level of automation and standardization to optimize your computing environment.
    - Virtualization provides a foundation for cloud computing.
- Virtualization has three characteristics that support the scalability and operating efficiency required for big data environments:
- **Partitioning:** In virtualization, many applications and operating systems are supported in a single physical system by partitioning (separating) the available resources.
- **Isolation:** Each virtual machine is isolated from its host physical system and other virtualized machines. Because of this isolation, if one virtual instance crashes, the other virtual machines and the host system aren't affected. In addition, data isn't shared between one virtual instance and another.

- **Encapsulation:** A virtual machine can be represented (and even stored) as a single file, so you can identify it easily based on the services it pro- vides. For example, the file containing the encapsulated process could be a complete business service. This encapsulated virtual machine could be presented to an application as a complete entity. Thus, encapsulation could protect each application so that it doesn't interfere with another application.

SERVER VIRTUALIZATION

- In server virtualization, one physical server is partitioned into multiple virtual servers. The hardware and resources of a machine — including the random access memory (RAM), CPU, hard drive, and network controller — can be virtualized (logically split) into a series of virtual machines that each runs its own applications and operating system.

APPLICATION VIRTUALIZATION

- Application infrastructure virtualization provides an efficient way to manage applications in context with customer demand. The application is encapsulated in a way that removes its dependencies from the underlying physical computer system. This helps to improve the overall manageability and portability of the application.

NETWORK VIRTUALIZATION

- Network virtualization — software-defined networking — provides an efficient way to use networking as a pool of connection resources. Networks are virtualized in a similar fashion to other physical technologies. Instead of relying on the physical network for managing traffic between connections, you can create multiple virtual networks all utilizing the same physical implementation.

PROCESSOR AND MEMORY VIRTUALIZATION

- Processor virtualization helps to optimize the processor and maximize performance. Memory virtualization decouples memory from the servers.

DATA AND STORAGE **VIRTUALIZATION**

- Data virtualization can be used to create a platform for dynamic linked data services. This allows data to be easily searched and linked through a unified reference source. As a result, data virtualization provides an abstract service that delivers data in a consistent form regardless of the underlying physical

database. In addition, data virtualization exposes cached data to all applications to improve performance.

- Storage virtualization combines physical storage resources so that they are more effectively shared. This reduces the cost of storage and makes it easier to manage data stores required for big data analysis.

(OR)

(b) Explain about the implementation of virtualization to work with Big Data.

Answer:

IMPLEMENTING VIRTUALIZATION TO WORK WITH BIG DATA

- Virtualization helps makes your IT environment smart enough to handle big data analysis. By optimizing all elements of your infrastructure, including hardware, software, and storage, you gain the efficiency needed to process and manage large volumes of structured and unstructured data. With big data, you need to access, manage, and analyze structured and unstructured data in a distributed environment.
- Big data assumes distribution. In practice, any kind of Map Reduce will work better in a virtualized environment. You need the capability to move workloads around based on requirements for compute power and storage.
- Virtualization will enable you to tackle larger problems that have not yet been scoped. You may not know in advance how quickly you will need to scale.
- Virtualization will enable you to support a variety of operational big data stores. For example, a graph database can be spun up as an image.
- The most direct benefit from virtualization is to ensure that Map Reduce engines work better. Virtualization will result in better scale and performance for Map Reduce. Each one of the Map and Reduce tasks needs to be executed independently. If the Map Reduce engine is parallelized and configured to run in a virtual environment, you can reduce management overhead and allow for expansions and contractions in the task workloads. Map Reduce itself is inherently parallel and distributed. By encapsulating the Map Reduce engine in a virtual container, you can run what you need whenever you need it. With virtualization, you increase your utilization of the assets you have already paid for by turning them into generic pools of resource

26. (a) Explain about Hadoop Distributed File system

Answer:

Understanding the Hadoop Distributed File System (HDFS)

- The Hadoop Distributed File System is a versatile, resilient, clustered approach to managing files in a big data environment. HDFS is not the final destination for files.

Rather, it is a data service that offers a unique set of capabilities needed when data volumes and velocity are high. Because the data is written once and then read many times thereafter, rather than the constant read-writes of other file systems, HDFS is an excellent choice for supporting big data analysis. The service includes a "Name Node" and multiple "data nodes" running on a commodity hardware cluster and provides the highest levels of performance when the entire cluster is in the same physical ack in the data center. In essence, the Name Node keeps track of where data is physically stored. Figure 9-1 depicts the basic architecture of HDFS.

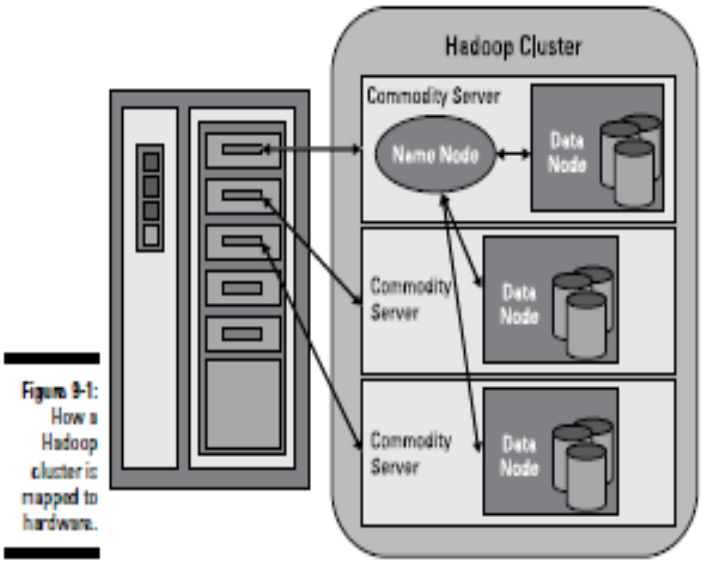


Figure 9-1: How a Hadoop cluster is mapped to hardware.

Name Nodes

- HDFS works by breaking large files into smaller pieces called blocks. The blocks are stored on data nodes, and it is the responsibility of the Name Node to know what blocks on which data nodes make up the complete file. The Name Node also acts as a "traffic cop," managing all access to the files, including reads, writes, creates deletes, and replication of data blocks on the data nodes. The complete collection of all the files in the cluster is sometimes referred to as the file system namespace. It is the Name Node's job to manage this namespace.
- Even though a strong relationship exists between the Name Node and the data nodes, they operate in a "loosely coupled" fashion. This allows the cluster elements to behave dynamically, adding (or subtracting) servers as the demand increases (or decreases). In a typical configuration, you find one Name Node and possibly a data node running on one physical server in the rack. Other servers run data nodes only.
- Data nodes are not very smart, but the Name Node is. The data nodes constantly ask the Name Node whether there is anything for them to do. This continuous behavior

also tells the Name Node what data nodes are out there and how busy they are. The data nodes also communicate among themselves so that they can cooperate during normal file system operations. This is necessary because blocks for one file are likely to be stored on multiple data nodes. Since the Name Node is so critical for correct operation of the cluster, it can and should be replicated to guard against a single point failure.

Data nodes

- Data nodes are not smart, but they are resilient. Within the HDFS cluster, data blocks are replicated across multiple data nodes and access is man- aged by the Name Node. The replication mechanism is designed for optimal efficiency when all the nodes of the cluster are collected into a rack. In fact, the Name Node uses a "rack ID" to keep track of the data nodes in the cluster. HDFS clusters are sometimes referred to as being "rack-aware." Data nodes also provide "heartbeat" messages to detect and ensure connectivity between the NameNode and the data nodes. When a heartbeat is no longer present, the Name Node un maps the data node from the cluster and keeps on operating as though nothing happened. When the heartbeat returns (or a new heartbeat appears), it is added to the cluster transparently with respect to the user or application.
- As with all file systems, data integrity is a key feature. HDFS supports a number of capabilities designed to provide data integrity. As you might expect, when files are broken into blocks and then distributed across different servers in the cluster, any variation in the operation of any element could affect data integrity. HDFS uses transaction logs and checksum validation to ensure integrity across the cluster.
- Transaction logs are a very common practice in file system and database design. They keep track of every operation and are effective in auditing or rebuilding of the file system should something untoward occur.
- Checksum validations are used to guarantee the contents of files in HDFS. When a client requests a file, it can verify the contents by examining its checksum. If the checksum matches, the file operation can continue. If not, an error is reported. Checksum files are hidden to help avoid tampering.
- Data nodes use local disks in the commodity server for persistence. All the data blocks are stored locally, primarily for performance reasons. Data blocks are replicated across several data nodes, so the failure of one server may not necessarily corrupt a file. The degree of replication, the number of data nodes, and the HDFS namespace are established when the cluster is implemented. Because HDFS is dynamic, all parameters can be adjusted during the operation of the cluster.

(OR)

(b) Describe the Process of Map Reduce framework

Answer:

HADOOP MAP REDUCE

- To fully understand the capabilities of Hadoop Map Reduce, we need to differentiate between Map Reduce (the algorithm) and an implementation of Map Reduce. Hadoop Map Reduce is an implementation of the algorithm developed and maintained by the Apache Hadoop project. It is helpful to think about this implementation as a Map Reduce engine, because that is exactly how it works. You provide input (fuel), the engine converts the input into output quickly and efficiently, and you get the answers you need. You are using Hadoop to solve business problems, so it is necessary for you to understand how and why it works. So, we take a look at the Hadoop implementation of Map Reduce in more detail.
- Hadoop Map Reduce includes several stages, each with an important set of operations helping to get to your goal of getting the answers you need from big data. The process starts with a user request to run a Map Reduce program and continues until the results are written back to the HDFS. Figure 9-2 illustrates how Map Reduce performs its tasks.

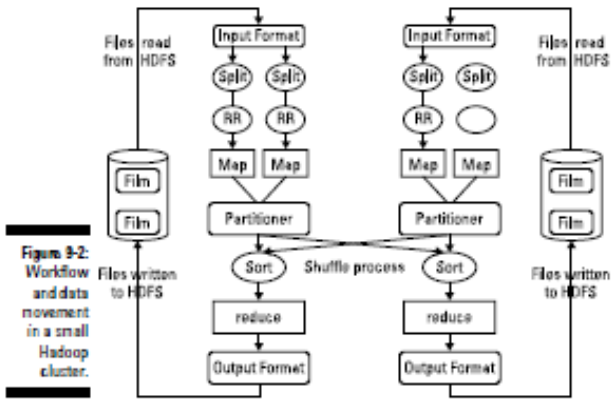


Figure 9-2: Workflow and data movement in a small Hadoop cluster.

HDFS and MapReduce perform their work on nodes in a cluster hosted on racks of commodity servers. To simplify the discussion, the diagram shows only two nodes.

Getting the data ready

- When a client requests a Map Reduce program to run, the first step is to locate and read the input file containing the raw data. The file format is completely arbitrary, but the data must be converted to something the program can process. This is the function of Input Format and Record Reader (RR).

- Input Format decides how the file is going to be broken into smaller pieces for processing using a function called Input Split. It then assigns a Record Reader to transform the raw data for processing by the map. If you read the discussion of map in Chapter 8, you know it requires two inputs: a key and a value. Several types of Record Readers are supplied with Hadoop, offering a wide variety of conversion options. This feature is one of the ways that Hadoop manages the huge variety of data types found in big data problems.

Let the mapping begin

- Your data is now in a form acceptable to map. For each input pair, a distinct instance of map is called to process the data. But what does it do with the processed output, and how can you keep track of them? Map has two additional capabilities to address the questions. Because map and reduce need to work together to process your data, the program needs to collect the output from the independent mappers and pass it to the reducers. This task is performed by an Output Collector. A Reporter function also provides information gathered from map tasks so that you know when or if the map tasks are complete.
- All this work is being performed on multiple nodes in the Hadoop cluster simultaneously. You may have cases where the output from certain map- ping processes needs to be accumulated before the reducers can begin. Or, some of the intermediate results may need to be processed before reduction. In addition, some of this output may be on a node different from the node where the reducers for that specific output will run. The gathering and shuffling of intermediate results are performed by a partitioner and a sort. The map tasks will deliver the results to a specific partition as inputs to the reduce tasks. After all the map tasks are complete, the intermediate results are gathered in the partition and a shuffling occurs, sorting the output for optimal processing by reduce.

Reduce and combine

- For each output pair, reduce is called to perform its task. In similar fashion to map, reduce gathers its output while all the tasks are processing. Reduce can't begin until all the mapping is done, and it isn't finished until all instances are complete. The output of reduce is also a key and a value. While this is necessary for reduce to do its work, it may not be the most effective output format for your application. Hadoop provides an Output Format feature, and it works very much like Input Format. Output Format takes the key-value pair and organizes the output for writing to HDFS. The last task s to actually write the data to HDFS. This is performed by Record Writer, and it performs similarly to Record Reader

except in reverse. It takes the Output Format data and writes it to HDFS in the form necessary for the requirements of the application program.

- The coordination of all these activities was managed in earlier versions of Hadoop by a job scheduler. This scheduler was rudimentary, and as the mix of jobs changed and grew, it was clear that a different approach was necessary. The primary deficiency in the old scheduler was the lack of resource management. The latest version of Hadoop has this new capability.
- Hadoop Map Reduce is the heart of the Hadoop system. It provides all the capabilities you need to break big data into manageable chunks, process the data in parallel on your distributed cluster, and then make the data available for user consumption or additional processing. And it does all this work in a highly resilient, fault-tolerant manner. This is just the beginning. The Hadoop ecosystem is a large, growing set of tools and technologies designed specifically for cutting your big data problems down to size.