Big data concepts: getting to know Big Data

What is Big Data?

Data that is large, complex & difficult to process using traditional methods

Sources of Big Data

4 predominant sources are:

Social media – likes, comments, media

Provides invaluable insight into customer behavior which is used for marketing

Provides a wealth of information for companies to analyze & learn

Email –

Mining data from an organization’s email account gives us insight into activity level of employees, health of business pipeline, customer behavior etc.

Sensor data –

IoT generate & record data

Open data –

Provided by govt. agencies that provide open datasets

Characteristics of Big Data

Volume – huge vol. of data

Velocity – speed in which data is processed

Variety – different types of data can be in big data & it needs to be organized

Variability – the meaning of data is constantly changing

Veracity – making sure the data is accurate

Visualization

Value – the data must have some value to user

Difference between variety & variability

In a coffee shop there are many types of coffee. This is variety. If you order the same blend every day & it tastes different each time, this is called variability.

Big data analytics answers questions traditional BI answers & a lot more:

Traditional BI:

What happened

When? Who? Where?

Advanced big data analytics:

Why did it happen?

Will it happen again?

What else does the data tell us?

BI deliverables:

Reports

Dashboards

KPI metrics

Scorecards

Ad-hoc queries

Big data deliverables

All deliverables of BI

Predictive models

Multivariate & text/image analytics

Advantages of Big Data analytics

Faster & better decision making

New products & services can be offered according to customer requirements obtained by analyzing data

Cost reduction in business operations

Real-time monitoring & forecasting of transactions

identification of hidden points & patters within very large datasets

Ability to accumulate & aggregate data from multiple sources

Major challenges in Big Data

Data growth – storing large datasets is extremely difficult to handle. Companies are using compression, tiering, duplication etc. to handle such data.

Unfamiliarity with big data methods – confusing it with traditional data methods is some of the biggest mistakes made by companies

Data security issues – companies now use access control, real-time security monitoring, data encryption, data segregation, endpoint security, etc.

Data integration – data comes from a variety of sources & integrating them is crucial for analytics

Big Data Concepts: Big Data Essentials

Difference between Data Warehouse & Big Data technologies

Data warehouse:

Can only store DBMS compatible data

Handles only structured data

Doesn’t utilize distributed file systems

Requires SQL knowledge

Query time has a direct correlation with data volume

Big data technologies:

Can store any kind of data

Handles structured, unstructured & semi-structured data

Utilizes distributed file systems

Requires knowledge of alternative tools like Hive or SparkSQL

Query time has no correlation with data volume & large requests may be processed as fast & small requests

Data warehousing in the age of Big Data

Organizations try to make the most of their unstructured data using existing data warehouses.

Many big data tools started as POCs & then were launched in production-like capacity.

Data warehouse is still relevant to this day & age.

Types of computing

Parallel computing – all processors have access to shared memory

Distributed computing – each processor has its own memory & information is exchanged by passing messages

Distributed computing advantages

Allows different users/computers to share information

Allows an application on 1 machine to leverage processing power, memory or storage from another machine

Might enhance performance of stand-alone applications

Allows enhancement of performance or availability

Hadoop components

MapReduce – computational model & software framework for applications run on Hadoop

HDFS (Hadoop Distributed File System) –

Hadoop application storage system

MapReduce consumes data from HDFS

Multiple replicas of data blocks are distributed on compute nodes in a cluster

Hive

HBase

Mahout

Sqoop

Flume

Zookeeper

MapReduce can easily tackle large datasets by distributing processing across many nodes then combining or reducing the results obtained from those nodes.

MapReduce Use Cases:

A social networking site could use MapReduce to determine potential friends, colleagues & other contacts

A booking website could use MapReduce to examine search criteria & historical behaviors

An industrial facility could collect equipment data from sensors across an installation & use MapReduce to tailor maintenance needs

HDFS uses-

HDFS provides a fault tolerant storage structure

World’s most reliable storage system

Enables high throughput access to application data

Hadoop works in a master-slave manner.

HDFS has 2 types of nodes that work in the same manner:

NameNodes

Regulates file access to clients

Maintain & manage slave nodes & assign tasks to them

Executes file system namespace operations – opening, closing, renaming files & directories

This metadata is available in-memory in master for faster data retrieval & copy of metadata is also stored in local disk for persistence. So, typically, NameNode memory should be high.

DataNodes

Manages data storage

There could be up to 1000 DataNodes in HDFS

They are the actual worker nodes that perform block creation, deletion & replication upon instruction from NameNode.

Run on commodity hardware with average configuration

Schema on Read – unprocessed data can be loaded into HDF with a structure applied at processing time based on the requirements of the processing application.

Schema on Write – used in RDBMS where schema need to be defined before the data can be loaded

Migration of Hadoop to cloud

Data platforms for cloud-native use are better than legacy Hadoop environments

Hadoop cloud advantages

Lower infrastructure costs

Data processing & performance improvement

Higher data processing throughput

Centralization of security credentials

Hadoop cloud migration

Hadoop wasn’t designed for cloud infrastructure

While cloud based Hadoop services is better than legacy Hadoop systems, both of them lag behind modern data platform counterparts. While migrating from Hadoop to a modern solution can be difficult, staying with Hadoop could lead to greater costs.

Hadoop cloud migration benefits

Long-term cost savings

Easy access & resource availability

Better collaboration

Better scalability

Hadoop & cloud differences

Cloud computing

On-demand, scalable & adaptable service models

Constitutes various computing concepts which involve many computers

Model in which processing & storage resources can be accessed from any location via internet

Cloud MapReduce relies on infrastructure offered by a CSP

Cloud computing is a storage drive with different OS, applications & frameworks accessible via internet

Hadoop can be installed in any cloud deployment service

In a way cloud is like a computer on which we can even install Hadoop

Hadoop

Focus on extracting value out of vol., variety & velocity

Uses simple programming models to process large datasets across clusters of computers

An ecosystem of open-source software projects which allow inexpensive computing

Hadoop has its own implementation of MapReduce

Software product developed by Apache to deal with data

Hadoop can’t provide middleware applications

Hadoop is a software

Teams migrate from Hadoop & choose cloud solutions for a variety of reasons:

Limitations within Hadoop systems

Modern cloud infrastructure provide new possibilities

HDFS Storage Featues

Distributed storage & replication – data is divided into blocks & stored on nodes inside HDFS structure. After that HDFS creates a replica of each block & stores it on other nodes.

High availability – data is replicated among nodes in case of failure, the data is still accessible

High reliability – reliable data storage on a cluster

High fault tolerance – multiple copies of data blocks exist on machines in the cluster

Replication – replication solves the problem of data loss

Scalability – the nodes in the cluster can be scaled as needed. Both vertical & horizontal scalability is possible

Security in Hadoop

Hadoop achieves security using Kerberos.

3 steps are needed to access a service in Hadoop via Kerberos:

Authentication

Authorization

Service request

In-memory storage systems

Kudu – Open-source storage engine for structured data which supports low-latency random access with efficient analytical access patterns.

Data is described using horizontal partitioning & then replicates each partitioning providing short latency periods

Elasticsearch – a distributed search & analytics engine which stores large volumes of data for fast search, fine-tuned relevancy & powerful analytics that’s easily scalable

Cockroach DB – commercial distributed SQL DBMS that stores copies of data in multiple locations, easily scalable, consistent replication

Need for in-memory storage systems

Performance requirements of big data tools have been hard to align. These tools require storage & processing power. This has lead to the rise of in-memory storage systems.

The application interacts with a master server which in-turn interacts with multiple RAM data partitions. The main memory is accessed only when querying data. This allows for faster access of the data. Data is loaded into in-memory databases in a compressed & non-relational format. It allows for direct navigation from index to row/column.

Eg: Maria DB, Redis, SAP Hana

In-memory storage systems

RAM is volatile & data is lost when hardware crashes. Non-volatile RAM (nvRAM) are being developed to mitigate the problem as they provide more persistent memory. For now, Flash is also used but there is a limit to how many times data can be written & erased on it.

In-memory database system can also act as a read-only analytic database. This can be used to store historical data on metrics for BI applications. This eliminates data indexing & could reduce IT costs.

Lack of translation & caching enables high speed. The data is used in the same form as the application that contains it.