# MODULE 1

# Introduction to Big Data Analytics

## LEARNING OBJECTIVES

**After studyingthis chapter,you will be able to:**

LO 1.1 Get conceptual understanding of data and web data; classification of data as structured, semi-, multi- and unstructured data; Big Data characteristics, types, classifications and handling techniques

LO 1.2 Get conceptual understanding of scalability, Massively Parallel Processing (MPP), distributed, cloud and grid computing

LO 1.3 Know the design layers in data-processing architecture for the data management and analytics

LO 1.4 Get introduced to data sources, data quality, data pre-processing, and the export of data store (such as tables, objects and files) to the cloud

LO 1.5 Get conceptual understanding of data storage and analysis; comparison between traditional systems such as Relational Database Management System (RDBMS), enterprise servers, data warehouse and approaches for Big Data storage and analytics

LO 1.6 Get knowledge of use cases and applications of Big Data in various fields.

## 1.1 ! INTRODUCTION

Two Grand Masters, Magnus Carlsen and Sergey Karjakin, played the final in World Chess Championship held on December 1, 2016. Magnus Carlsen won this final and the title of Grand Master. Sergey Karjakin, in order to win, would have to design a new strategy to defeat Carlsen and other players next year. A Grand Master typically studies the moves made in earlier matches played by Grand Masters, analyzes them and then designs his strategies. Evolving strategy to defeat an opponent could even make good use of the data of Gary Kasparov's matches from 1984. Study and analysis of a large number of matches helps in evolving a winning strategy. Similarly, analytics of Big Data could enable discovery of new facts, knowledge and strategy in a number of fields, such as manufacturing, business, finance, healthcare, medicine and education.

### 1.1.1 Need of Big Data

The rise in technology has led to the production and storage of voluminous amounts of data. Earlier megabytes ($10^{6}$ B) were used but nowadays petabytes ($10^{15}$ B) are used for processing, analysis, discovering new facts and generating new knowledge. Conventional systems for storage, processing and analysis pose challenges in large growth in volume of data, variety of data, various forms and formats, increasing complexity, faster generation of data and need of quickly processing, analyzing and usage.

Figure 1.1 shows data usage and growth. As size and complexity increase, the proportion of unstructured data types also increase.

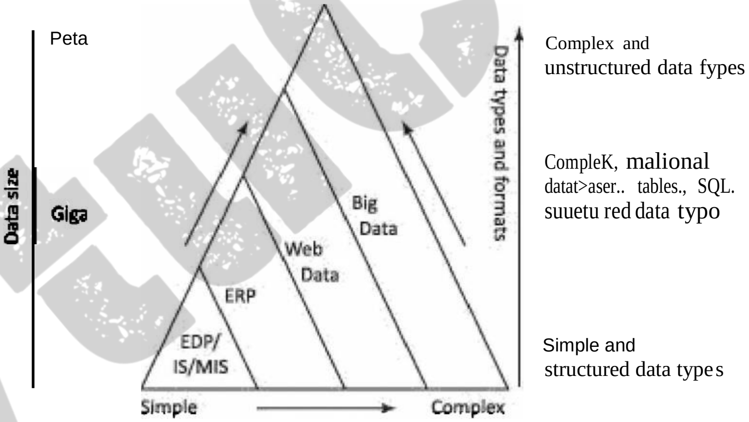


Figure 1.1 Evolution of Big Data and their characteristics

An example of a traditional tool for structured data storage and querying is RDBMS. Volume, velocity and variety (3Vs) of data need the usage of number of programs and tools for analyzing and processing at a very high speed. When integrated with the Internet of Things, sensors and machines data, the veracity of data is an additional V. (Section 1.2.3)

Big Data requires new tools for processing and analysis of a large volume of data. For example, unstructured, NoSQL (not only SQL) data or Hadoop compatible system data.

Following are selected key terms and their meanings, which are essential to understand the topics discussed in this chapter:

*Application* means application software or a collection of software components. For example, software for acquiring, storing, visualizing and analyzing data. An *application* performs a group of coordinated activities, functions and tasks.

*Application Programming Interface* (API) refers to a software component which enables a user to access an application, service or software that runs on a local or remote computing platform. An API initiates running of the application on receiving the message(s) from the user-end. An API sends the user-end messages to the other-end software. The other-end software sends responses or messages to the API and the user.

*Data Model* refers to a map or schema, which represents the inherent properties of the data. The map shows groupings of the data elements, such as records or tables, and their associations. A model does not depend on software using that data.

*Data Repository* refers to a collection of data. A data-seeking program relies upon the data repository for reporting. The examples of repositories are database, flat file and spreadsheet. [Repository in *English* means a group which can be relied upon to look for required things, such as special information or knowledge. For example, a repository of paintings by various artists.]

*Data Store* refers to a data repository of a set of objects. Data store is a general concept for data repositories, such as database, relational database, flat file, spreadsheet, mail server, web server and directory services. The objects in data store model are instances of the classes which the *database schemas* define. A data store may consist of multiple schemas or may consist of data in only one schema. Example of only one scheme for a data store is a relational database.

*Distributed Data Store* refers to a data store distributed over multiple nodes. Apache Cassandra is one example of a distributed data store. (Section 3.7)

*Database* (DB) refers to a grouping of tables for the collection of data. A table ensures a systematic way for accessing, updating and managing data. A database pertains to the applications, which access them. A *database* is a repository for querying the required information for analytics, processes, intelligence and knowledge discovery. The databases can be distributed across a network consisting of servers and data warehouses.

*Table* refers to a presentation which consists of row fields and column fields. The values at the fields can be number, date, hyperlink, image, object or text of a document.

*Flat File* means a file in which data cannot be picked from in between and must be read from the beginning to be interpreted. A file consisting of a single-table file is called a flat file. An example of a flat file is a csv (comma-separated value) file. A flat file is also a data repository.

*Flat File Database* refers to a database in which each record is in a separate row unrelated to each other.

*CSV File* refers to a file with comma-separated values. For example, CSlOl, "Theory of Computations", 7.8 when a student's grade is 7.8 in subject code CSlOl and subject "Theory of Computations".

*Name-Value Pair* refers to constructs used in which a field consists of name and the corresponding value after that. For example, a name value pair is *date, ""Oct. 20, 2018"", chocolates_sold, 178;*

*Key-Value Pair* refers to a construct used in which a field is the key, which pairs with the corresponding value or values after the key. For example, consider a tabular record, ""Oct. 20, 2018""; ""chocolates_sold"", 178. The *date* is the primary key for finding the date of the record and chocolates_sold is the secondary key for finding the number of chocolates sold.

*Hash Key-Value Pair* refers to the construct in which a hash function computes a key for indexing and search, and distributing the entries (key/value pairs) across an array of slots (also called buckets). (Section 3.3.1)

*Spreadsheet* refers to the recording of data in fields within rows and columns. A field means a specific column of a row used for recording information. The values in fields associates a program, such as Microsoft Excel 2013. An example of a spreadsheet application is *accounting.* The application manages, analyzes and enables new values either directly or using formulae which contain the relationships of a field with cells and rows. Examples of functions are SUMIF and COUNTIF, delete duplicate entries, sort using multiple keys, filter single or multiple columns, create a filter using filtering criteria or rules for multi-fields, and create top-n lists for values or percentages.

*Stream Analytics* refers to a method of computing continuously, i.e. even while events take place data flows through the system.

*Database Maintenance* (DBM) refers to a set of tasks which improves a database. DBM uses functions for improving performance (such as by query planning and optimization), freeing-up storage space, updating internal statistics, checking data errors and hardware faults.

*Database Administration* (DBA) refers to the function of managing and maintaining Database Management System (DBMS) software regularly. A database administering personnel has many responsibilities, such as installation, configuration, database design, implementation upgrading, evaluation of database features, reliable backup and recovery methods for the database.

*Database Management System* (DBMS) refers to a software system, which contains a set of programs specially designed for *creation* and *management* of data stored in a database. Transactions can be performed with database/relational database.

*Relational Database* is a collection of data into multiple tables, which relate to each other through special fields, called keys (primary key, foreign key and unique key). Relational databases provide flexibility.

*Relational Database Management System* (RDBMS) refers to a software system used for creation of relational databases and management of data which are stored in a relational database. RDBMS functions perform the *transactions* on the relational database. Examples of RDBMS are MySQL, PostGreSQL(Oracle database created using PL/SQL) and Microsoft SQL server using T-SQL.

*Transaction (*trans + action) means two interrelated sets of operations, actions or instructions. A transaction is a set of actions which accesses, changes, updates, appends or deletes various data. A command 'connect' enables transfers between DBMS software and a database. The database in return connects the DBMS. An example of this is query transfer from a system to a database. The database in return transfers the answer of the query.

*SQL* stands for Structured Query Language. It is a language used for schema creation and schema modifications, data-access control, creating an SQL client and creating an SQL server for a database. It is a language for managing relational databases, and viewing, querying and changing (update, insert, append or delete) databases.

*Database Connection* refers a function DB_connect open() which an application calls to connect to enable the access to the DBMS. The application calls the function DB_connect close () to disable the access.

*Database Connectivity* (DBC) refers to a standard application programming interface (API), which provides connectivity for accessing the DBMSs. A DBC design is independent of the DB system and OS used. An application written using a DBC can therefore perform operations or actions at both the client and the DB server end. Little changes in code suffice for accessing the data. Two examples of DBCs are Open Database Connectivity (ODBC) and Java Database Connectivity 0DBC).

*Database Connectivity Driver* refers to a translation layer which resides between an application using the application and the DBMS. The application uses DBC functions through a DBC driver manager with which it is linked. A DBC driver manager manages the drivers associated with the DBMSs. The DBC driver sends the queries to a DBMS. Drivers exist for many data sources and all major DBMSs.

*DB2* is IBM RDBMS. DBZ has many features. For example, triggers, stored procedures and dynamic bitmapped indexing for number of application types, such as traditional host-based applications, client/ server-based applications and business intelligence applications.

*Data Warehouse* refers to sharable data, data stores and databases in an enterprise. It consists of integrated, subject oriented (such as finance, human resources and business) and non-volatile data stores, which update regularly.

*Data Mart* is a subset of data warehouse. Data mart corresponds to specific business entity on a single subject (or functional area), such as sales or finance data mart is also known as High Performance Query Structures (HPQS).

*Process* means a composition of group of structured activities, tasks or services that lead to a particular goal. For example, purchase process for airline tickets. A *process* specifies activities with relevance rules based on data in the process.

*Process Matrix* refers to a multi-element entity, each element of which relates a set of data or inputs to an activity (or subset of activities).

*Business Process* is an activity, series of activities or a collection of inter-related structured activities, tasks or processes. A business process serves a particular goal, specific result, service or product. The business process is a representation, process *matrix* or flowchart of a sequence of activities with interleaving decision points.

*Business Intelligence* is a process which enables a business service to extract new facts and knowledge that enable intelligent decisions. The new facts and knowledge follow from the previous results of business-data processing, aggregation and analysis.

*Batch Processing* is processing of transactions in batches with no interactions. When one set of transactions finish, the results are stored and the next batch starts processing. Credit card transactions is a good example of the same. The results aggregate at the end of the month for all usages of the card. Batch processing involves the collection of inputs for a specified period and then running them in a scheduled manner.

*Batch Transaction Processing* refers to the execution of a series of transactions without user interactions. Transaction jobs are set up so they can be run to completion. Scripts, command-line arguments, control files or job-control language *predefine* the input parameters for the transactions.

*Streaming Transaction Processing* refers to processing for log streams, event streams, twitter streams and queries. The processing of streaming data needs a specialized software framework. Storm from Twitter, 54 from Yahoo, SPARK streaming, HStreaming and Flume are examples of frameworks for real-time streaming computations.

*In-memory* means operations using CPU memory, such as RAM or caches. Data in-memory is from a disk or external data source. The operations are fast on in-memory accesses of data, table or data sets, columns or rows compared to disk-accesses.

*Interactive Transaction Processing* means processing the transactions which involve continual exchange of information between the computer and user; for example, user interactions during e-shopping or e-banking. The processing here is just the opposite of batch processing. Decision on historical data is fast. Interactive query processing has low latency. Low latencies are obtained by the various approaches: massively parallel processing (MPP), in-memory databases and columnar databases.

*Real-Time Processing* refers to processing for obtaining results for making decisions in real time, processing as and when the data acquires or generates in live data (streaming) with low latency.

*Real-Time Transaction Processing* means that transactions process at the same time as the data arrives from the data sources. An example of such processing is transaction processing at an ATM machine.

*Extract, Transform and Load (ETL)* refers to the process, which enables data retrieval, integration, transformation and storage (load). *Extract* means obtaining data from homogeneous or heterogeneous data sources. *Transform* means transforming or optimizing data for the application, and storing the data in an appropriate structure or format. *Load* means the structured data is loaded in the final target database, i.e. data store or data warehouse.

*Machine* is a computing node or platform for processing, computing and storing. Here, sets of data, programs, applications, DBs or DBMSs reside. When other remote machines access the resources from the machine, it is identified by a name within a network.

*Server* is a processing, computing and storing node. A server generates responses, sends replies and messages, and renders the data sought. *Server* refers to sets of data, programs, applications, data-stores, DBs or DBMSs which the clients access.

*Service* means a mechanism which enables the provisioning of access to one or more capabilities. An interface provides the access capabilities. The access to a capability is consistent with various constraints and policies. A *service description* specifies these constraints and policies. Examples of services are web service, cloud service and BigQueryservice.

*Service-Oriented Architecture (SOA)* is a software architecture model which consists of services, messages, operations and processes. SOA components distribute over a network or the Internet in a high-level business entity. New business applications and an application-integration architecture can be developed using an SOA in an enterprise.

*Descriptive Analytics* refers to deriving additional value from visualizations and reports.

*Predictive Analytics* refers to advanced analytics which enables extraction of new facts and knowledge to predict or forecast.

*Prescriptive Analytics* refers to derivation of additional value and undertaking better decisions for new option(s); for example, maximizing profit.

*Cognitive Analytics* refer to analysis of sentiments, emotions, gestures, facial expressions, and actions similar to ones the humans do. The analytics follow the process of learning, understanding and representing. [Cognitive in English means relating to the process of learning, understanding and representing knowledge. (CollinsDictionary)]

This chapter introduces the readers to the concepts of Big Data, scaling-up and scaling-out of data processing and scalability for storage and analytics. It introduces the concepts of data processing architecture, data sources, data quality and the new technological developments in data management for analysis. These are supported by examples and cases on Big Data analytics. This chapter aims to build a foundation before the in-depth study of Big Data and analytics tools facilitated by the subsequent chapters of the book.

Section 1.2 introduces Big Data and its characteristics, types and classification methods. Section 1.3 describes scalability, scaling up, scaling out of processing and analytics, massively parallel processors, and cloud, grid and distributed computing. Section 1.4 introduces data architecture design and data management. Section 1.5 describes data sources, data quality, data pre-processing and export of data stores to the cloud. Section 1.6 describes traditional systems, such as SQL, Relational Database Management System (RDBMS), enterprise servers and data warehouse for data storage and analysis, as well as the approaches for Big Data storage, processing and analytics. Section 1.7 describes Big Data analytics case studies and applications.

## 1.2 ! BIG DATA

Following subsections describe the definitions of data, web data, Big Data, Big Data characteristics, types, classifications and handling techniques:

LO 1.1

Com:e,prual 11.m derista nd illilg of alat.3, web d:aiv.a) Big Data, Characteris~ics. types (lassmc_atioliils an al harnall ing te:::lmliili(IU@S

### *Definitions* of *Data*

*Data* has several definitions. Usages can be singular or plural. "Data is information, usually in the form of facts or statistics that one can analyze or use for further calculations." [Collins English Dictionary] "Data is information that can be stored and used by a computer program.". [Computing] "Data is information presented in numbers, letters, or other form". [Electrical Engineering, Circuits, Computing and Control] "Data is information from series of observations, measurements or facts".

[Science] "Data is information from series of behavioural observations, measurements or facts". [Social Sciences]

### *Definition of Web Data*

*Web* is large scale integration and presence of data on web servers. Web is a part of the Internet that stores web data in the form of documents and other web resources. URLs enable the access to web data resources.

*Web data* is the data present on web servers (or enterprise servers) in the form of text, images, videos, audios and multimedia files for web users. A user (client software) interacts with this data. A client can access (pull) data of responses from a server. The data can also publish (push) or post (after registering subscription) from a server. Internet applications including web sites, web services, web portals, online business applications, emails, chats, tweets and social networks provide and consume the web data.

Some examples of web data are Wikipedia, Google Maps, McGraw-Hill Connect, Oxford Bookstore and YouTube.

1. Wikipedia is a web-based, free-content encyclopaedia project supported by the Wikimedia Foundation.
2. Google Maps is a provider of real-time navigation, traffic, public transport and nearby places by Google Inc.
3. McGraw-Hill Connect is a targeted digital teaching and learning environment that saves students' and instructors' time by improving student performance for a variety of critical outcomes.
4. Oxford Bookstore is an online book store where people can find any book that they wish to buy from millions of titles. They can order their books online at www .oxfordbookstore.com
5. YouTube allows billions of people to discover, watch and share originally-created videos by Google Inc.

## 1.2.1 Classification of Data-Structured, Semi-structured and Unstructured

Data can be classified as structured, semi-structured, multi-structured and unstructured.

Structured data conform and associate with data schemas and data models. Structured data are found in tables (rows and columns). Nearly 15-20% data are in structured or semi-structured form. Unstructured data do not conform and associate with any data models.

Applications produce continuously increasing volumes of both *unstructured* and *structured* data. Data sources generate data in three forms, viz. structured, semi•structured and unstructured. (Refer online contents associated with the Practice Exercise 1.1 for four forms, viz. structured, semi-structured, multi-structured and unstructured sources.)

### *Using Structured Data*

Structured data enables the following:

*data insert, delete, update and append*

*Indexing* to enable faster data retrieval

*Scalability* which enables increasing or decreasing capacities and data processing operations such as, storing, processing and analytics

*Transactions processing* which follows ACID rules (Atomicity, Consistency, Isolation and Durability)

*encryption and decryption* for data security.

### *Using Semi-Structured Data*

Examples of *semi-structured data* are XML and JSON documents. Semi-structured data contain tags or other markers, which separate semantic elements and enforce hierarchies of records and fields within the data. Semi-structured form of data does not conform and associate with formal data model structures. Data do not associate data models, such as the relational database and table models.

### *Using Multi-Structured Data*

*Multi-structured data* refers to data consisting of multiple formats of data, viz. structured, semi-structured and/or unstructured data. Multi-structured data sets can have many formats. They are found in non-transactional systems. For example, streaming data on customer interactions, data of multiple sensors, data at web or enterprise server or the data- warehouse data in multiple formats.

Large-scale interconnected systems are thus required to aggregate the data and use the widely distributed resources efficiently.

Multi- or semi-structured data has some semantic meanings and data is in both structured and unstructured formats. But as structured data, semi-structured data nowadays represent a few parts of data (5-10%). Semi-structured data type has a greater presence compared to structured data.

Following is an example of multi-structured data.

### EXAMPLE 1.1

Give examples of multi-structured data.

SOLUTION

Structured component of data: Each chess moves is recorded in a table in each match that players refer in future. The records consist of serial numbers (row numbers, which mean move numbers) in the first column and the moves of White and Black in two subsequent vertical columns. Volume of data, i.e. data used for analyzing erroneous or best moves in the matches, keeps growing with more and more tables, and may eventually become 'voluminous data'.

Unstructured component of data: Social media generates data after each international match. The media publishes the analysis of classical matches played between Grand Masters. The data for analyzing chess moves of these matches are thus in a variety of formats.

Multi-structured data: The voluminous data of these matches can be in a structured format (i.e. tables) as well as in unstructured formats (i.e. text documents, news columns, biogs, Facebook etc.). Tools of multi-structured data analytics assist the players in designing better strategies for winning chess championships.

## *Using Unstructured Data*

*Unstructured data* does not possess data features such as a table or a database. Unstructured data are found in file types such as .TXT, .CSV. Data may be as key-value pairs, such as hash key-value pairs. Data may have internal structures, such as in e•mails. The data do not reveal relationships, hierarchy relationships or object-oriented features, such as extendibility. The relationships, schema and features need to be separately established. Growth in data today can be characterised as mostly unstructured data. Following are some examples of unstructured data.

### EXAMPLE 1.2

Give examples of unstructured data.

SOLUTION

Examples of unstructured data are:

Mobile data: Text messages, chat messages, tweets, biogs and comments

Website content data: YouTube videos, browsing data, e-payments, web store data, user-generated maps

Social media data: For exchanging data in various forms

Texts and documents

Personal documents and e-mails

Text internal to an organization: Text within documents, logs, survey results

Satellite images, atmospheric data, surveillance, traffic videos, images from Instagram, Flickr (upload, access, organize, edit and share photos from any device from anywhere in the world).

## 1.2.2 Big Data Definitions

**Big Data** is high-*volume,* high-*velocity* andi or high-*variety* information asset that requires new forms of processing for enhanced *decision making, insight discovery* and *process optimization* (Gartner12012). Other definitions can be found in existing literature.

eanililgs and varjous d~nitions oHme word 'Bigi Data'

Industry analyst Doug Laney described the '3Vs', i.e. volume, variety and/or velocity as the key "data management challenges" for enterprises. Analytics also describe the '4Vs', i.e. volume, velocity, variety and veracity. A number of other definitions are available for Big Data, some of which are given below.

"A collection of data sets so large or complex that traditional data processing applications are inadequate." - *Wikipedia*

"Data of a very large size, typically to the extent that its manipulation and management present significant logistical challenges." [Oxford English Dictionary (traditional database of authoritative definitions)]

"Big Data refers to data sets whose size is beyond the ability of typical database software tool to capture, store, manage and analyze." [The McKinsey Global Institute, 2011]

## 1.2.3 Big Data Characteristics

Characteristics of Big Data, called 3Vs (and 4Vs also used) are:

**Volume** The phrase 'Big Data' contains the term *big,* which is related to size of the data and hence the characteristic. Size defines the amount or quantity of data, which is generated from an application(s). The size determines the processing considerations needed for handling that data.

**Velocity** The term velocity refers to the speed of generation of data. Velocity is a measure of how fast the data generates and processes. To meet the demands and the challenges of processing Big Data, the velocity of generation of data plays a crucial role.

**Variety**Big Data comprises of a variety of data. Data is generated from multiple sources in a system. This introduces variety in data and therefore introduces *'complexity'*. Data consists of various forms and formats. The variety is due to the availability of a large number of heterogeneous platforms in the industry. This means that the type to which Big Data belongs to is also an important characteristic that needs to be known for proper processing of data. This characteristic helps in effective use of data according to their formats, thus maintaining the importance of Big Data.

*Veracity* is also considered an important characteristic to take into account the quality of data captured, which can vary greatly, affecting its accurate analysis.

The 4Vs (i.e. volume, velocity, variety and veracity) data need tools for mining, discovering patterns, business intelligence, artificial intelligence (AI), machine learning (ML), text analytics, descriptive and predictive analytics, and the data visualization tools.

### 1.2.4 Big Data Types

A task team on Big Data classified the types of Big Data (lune 2013)[2]. Another team from IBM developed a different classification for Big Data types.[3]

Following are the suggested types:

1. *Social networks and web data,* such as Facebook, Twitter, e-mails, biogs and YouTube.
2. *Transactions data and Business Processes (BPs) data,* such as credit card transactions, flight bookings, etc. and public agencies data such as medical records, insurance business data etc.
3. *Customer master data,* such as data for facial recognition and for the name, date of birth, marriage anniversary, gender, location and income category,
4. *Machine-generated data,* such as machine-to-machine or Internet of Things data, and the data from sensors, trackers, web logs and computer systems log. Computer generated data is also considered as machine generated data from data store. Usage of programs for processing of data using data repositories, such as database or file, generates data and also machine generated data.
5. *Human-generated data* such as biometrics data, human-machine interaction data, e•mail records with a mail server and MySQL database of student grades. Humans also records their experiences in ways such as writing these in notebooks or

diaries, taking photographs or audio and video clips. Human-sourced information is now almost entirely digitized and stored everywhere from personal computers to social networks. Such data are loosely structured and often ungoverned.

The following examples illustrate machine-generated data.

EXAMPLE 1.3

Give three examples of the machine-generated data.

SOLUTION

Examples of machine-generated data are:

1. Data from computer systems: Logs, web logs, security/ surveillance systems, videos/images etc.
2. Data from fixed sensors: Home automation, weather sensors, pollution sensors, traffic sensors etc.
3. Mobile sensors (tracking) and location data.

Section 1.7 describes Big Data Analytics use cases, case studies and applications in detail. The following example illustrates the usages of Big Data generated from multiple types of data sources for optimizing the services offered, products, schedules and predictive tasks.

EXAMPLE 1.4

Think of a manufacturing and retail marketing company, such as LEGO toys.

How does such a toy company optimize the services offered, products and schedules, devise ways and use Big Data processing and storing for predictions using analytics?

SOLUTION

Assume that a retail and marketing company of toys uses several Big Data sources, such as (i) *machine-generated data* from sensors (RFID readers) at the toy packaging, (ii) *transactions data* of the sales stored as web data for automated reordering by the retail stores and (iii) tweets, Facebook posts, e-mails, messages, and web data for messages and reports.

The company uses Big Data for understanding the toys and themes in present days that are popularly demanded by children, predicting the future types and demands. The company using such predictive analytics, optimizes the product mix and manufacturing processes of toys. The company optimizes the services to retailers by maintaining toy supply schedules. The company sends messages to retailers and children using social media on the arrival of new and popular toys.

The following example illustrates the Big Data features of 3Vs and their applications.

EXAMPLE 1.5

Give an example offeatures of 3Vs in Big Data and application.

SOLUTION

Consider satellite images of the Earth's atmosphere and its regions. The *Volume* of data from the satellites is large. A number of Indian satellites, such as KALPANA, INSAT-lA and INSAT-3D generate this data. Foreign satellites also generate voluminous data continuously. Satellites record the images of full disk and sectors, such as east and west Asia sectors and regions.

*Velocity* is also large. A number of satellites collect this data round the clock. Big Data analytics helps in drawing of maps of wind velocities, temperatures and other whether parameters.

*Variety* of images can be in visible range, such as IR-1 (infrared range -1), IR• Z(infrared range -2), shortwave infrared (SWIR), MIR (medium range IR) and colour composite.

Data *Veracity,* uncertain or imprecise data, is as important as Volume, Velocity and Variety. Uncertainty arises due to poor resolutions used for recording or noise in images due to signal impairments.

Data processing needs increased speed of computations due to higher volumes. Need of data management, storage and increased analytics requires new innovative non-traditional methods.

Big Data of satellites helps in predicting weather, and mapping of different crops and from that estimating the expected crop yield.

The following examples explain the uses of Big Data generated from multiple types of data sources.

EXAMPLE 1.6

How are Big Data used in the following companies and services using analytics?

(i) Chocolate Marketing Company with large number of installed Automatic Chocolate Vending Machines (ACVMs)

(ii) Automotive Components and Predictive Automotive Maintenance Services (ACPAMS) rendering customer services for maintenance and servicing of (Internet) connected cars and its components

(iii) Weather data Recording, Monitoring and Prediction (WRMP) Organization.

SOLUTION

(i) Assume ACVM company. Each ACVM sells five flavours (FLl, FL2, FL3, FL4 and FLS) KitKat, Milk, Fruit and Nuts, Nougat and Oreo. The company uses Big Data types as: *Machine-generated data* on the sale of chocolates, reports of unfilled or filled machine *transaction data. Human-generated data* of buyer-machine interactions at the ACVMs. *Social networks and web data* on feedback and personalized messages based on interactions and human-generated data on facial recognition of the buyers. The company uses Big Data for efficient and optimum planning of fill service for chocolates, sentiment analysis of buyers for specific flavours, ACVMs location and periods of higher-sales analysis, assessing needs of predictive maintenances of machines, additions and relocations of machines, and predictions, strategies and planning for festival sales.

(ii) ACPAMS uses Big Data types as: *machine-generated data* from sensors at automotive components, such as brakes, steering and engine from each car; *transactions data* stored at the service website; social networks and web data in the form of messages, feedback and reports from customers. The service provides messages for scheduled and predictive maintenances. The service generates reports on *social networks and updates the web data* for the manufacturing plant. The service generates reports about components qualities and needed areas for improvement in products of the company.

(iii) WRMP Organization uses Big Data types as: machine-generated data from sensors at weather stations and satellites, social networks and web data and the reports and alerts issued by many centers around the world. The organization stores and processes the weather records generated by its stations, social networks and web data collected from other centers. The organization issues maps and weather warnings, predicts weather, rainfall in various regions, expected dates of arrival of monsoon in different regions, issues forecasts on social networks and web pages, generates social network and web data for areal maps of cloud and wind.

### 1.2.5 Big Data Classification

Big Data can be classified on the basis of its characteristics that are used for designing data architecture for processing and analytics. Table 1.1 gives various classification methods for data and Big Data.

**Table 1.1** Various classification methods for data and Big Data

| Basis of Classification | Examples |
|---|---|
| Data sources (traditional) | Data storage such as records, RDBMs, distributed databases, row-oriented In-memory data tables, column-oriented In-memory data tables, data warehouse, server, machine-generated data, human-sourced data, Business Process (BP) data, Business Intelligence (BI) data |
| Data formats (traditional) | Structured and semi-structured |
| Big Data sources | Data storage, distributed file system, Operational Data Store (ODS), data marts, data warehouse, NoSQL database (MongoDB, Cassandra), sensors data, audit trail of financial transactions, external data such as web, social media, weather data, health records |
| Big Data formats | Unstructured, semi-structured and multi-structured data |
| Data Stores structure | Web, enterprise or cloud servers, data warehouse, row-oriented data for OLTP, column-oriented for OLAP, records, graph database, hashed entries for key/value pairs |
| Processing data rates | Batch, near-time, real-time, streaming |
| Processing Big Data rates | High volume, velocity, variety and veracity, batch, near real-time and streaming data processing, |
| Analysis types | Batch, scheduled, near real-time datasets analytics |
| Big Data processing methods | Batch processing (for example, using MapReduce, Hive or Pig), real-time processing (for example, using SparkStreaming, SparkSQL, Apache Drill) |
| Data analysis methods | Statistical analysis, predictive analysis, regression analysis, Mahout, machine learning algorithms, clustering algorithms, classifiers, text analysis, social network analysis, location-based analysis, diagnostic analysis, cognitive analysis |
|  | Human, business process, knowledge discovery, enterprise applications, Data |

| | |
|---|---|
| Data usages | Stores |

### 1.2.6 Big Data HandlingTechniques

Following are the techniques deployed for Big Data storage, applications, data management and mining and analytics:

Huge data volumes storage, data distribution, high-speed networks and high• performance computing

Applications scheduling using open source, reliable, scalable, distributed file system, distributed database, parallel and distributed computing systems, such as Hadoop (Chapter 2) or Spark (Chapters 5-10)

Open source tools which are scalable, elastic and provide virtualized environment, clusters of data nodes, task and thread management

Data management using NoSQL, document database, column-oriented database, graph database and other form of databases used as per needs of the applications and in-memory data management using columnar or Parquet formats during program execution

Data mining and analytics, data retrieval, data reporting, data visualization and machine-learning Big Data tools.

**Self-Assessment Exercise linkedto LO 1.1**

1. How do you define data, web data and Big Data?
2. How do you classify data as structured, semi-structured, multi-structured and unstructured?
3. Give data example of student records at a University and explain structured data, hierarchical relationships between them.
4. Recall three examples in Example 1.6. How would you classify data which you shall be using for analytics in these examples?
5. Consider the usage examples of Big Data for a car company. Assume that company manufactures five models of cars, and each model is available in five colours and five shades. The company collects inputs from customers and sales centres, and inputs of component malfunctions from service centres for different models. The company also uses social media inputs. Explain 3Vs characteristics in this company's data.

## 1.3 ! SCALABILITY AND PARALLEL PROCESSING

Big Data needs processing of large data volume, and therefore needs intensive computations. Processing complex applications with large datasets (terabyte to petabyte datasets) need hundreds of computing nodes. Processing of this much distributed data within a short time and at minimum cost is problematic.

LO 1.2

Sca'lab ii ify, scal11filg up, scaUng out in C!limilb!.!.!tedl comp1JJJtJi11tg, 21ssively lParal lel Processi11tg ( ~PP), doudl, griiC!l, vo luin,1te@ri11tg compJ.!lti11ilg systems

### *Convergence ofData Environments and Analytics*

Big Data can co-exist with traditional data store. Traditional data stores use RDBMS tables or data warehouse. Big Data processing and analytics requires scaling up and scaling out, both vertical and horizontal computing resources. Computing and storage systems when run in parallel, enable scaling out and increase system capacity.

*Scalability* enables increase or decrease in the capacity of data storage, processing and analytics. *Scalability* is the capability of a system to handle the workload as per the magnitude of the work. System capability needs increment with the increased workloads. When the workload and complexity exceed the system capacity, scale it up and scale it out.

The following subsection describes the concept of analytics scalability.

### 1.3.1 Analytics Scalabilityto Big Data

*Vertical scalability* means scaling up the given system's resources and increasing the system's analytics, reporting and visualization capabilities. This is an additional way to solve problems of greater complexities. *Scaling up* means designing the algorithm according to the architecture that uses resources efficiently. For example, $x$ terabyte of data take time $t$ for processing, code size with increasing complexity increase by factor $n$, then scaling up means that processing takes equal, less or much less than $(n \times t)$.

*Horizontal scalability* means increasing the number of systems working in coherence and scaling out the workload. Processing different datasets of a large dataset deploys horizontal scalability. *Scaling out* means using more resources and distributing the processing and storage tasks in parallel. If $r$ resources in a system process $x$ terabyte of data in time t, then the $(p \times x)$ terabytes process on $p$ parallel distributed nodes such that the time taken up remains $t$ or is slightly more than $t$ (due to the additional time required for Inter Processing nodes Communication (IPC).

The easiest way to scale up and scale out execution of analytics software is to implement it on a bigger machine with more CPUs for greater volume, velocity, variety and complexity of data. The software will definitely perform better on a bigger machine. However, buying faster CPUs, bigger and faster RAM modules and hard disks, faster and bigger motherboards will be expensive compared to the extra performance achieved by efficient design of algorithms. Also, if more CPUs add in a computer, but the software does not exploit the advantage of them, then that will not get any increased performance out of the additional CPUs.

Alternative ways for scaling up and out processing of analytics software and Big Data analytics deploy the Massively Parallel Processing Platforms (MPPs), cloud, grid, clusters, and distributed computing software.

The following subsections describe computing methods for high availability and scalable computations and analysis.

### 1.3.2 Massively ParallelProcessingPlatforms

A solwmonfor B.tg dat~ proces~iliilg is to pell'fol'Jil'il parallel and alistirib!!!t:ea compllilltlilitg in, ::i de udL compm:ilillg envimnment

Scaling uses parallel processing systems. Many programs are so large and/ or complex that it is impractical or impossible to execute them on a single computer system, especially in limited computer memory. Here, it is required to enhance (scale) up the computer system or use massive parallel processing (MPPs) platforms. Parallelization of tasks can be done at several levels: (i) distributing separate tasks onto separate threads on the same CPU, (ii) distributing separate tasks onto separate CPUs on the same computer and (iii) distributing separate tasks onto separate computers.

When making software, draw the advantage of multiple computers (or even multiple CPUs within the same computer) and software which need to be able to parallelize tasks. Multiple compute resources are used in parallel processing systems. The computational problem is broken into discrete pieces of sub-tasks that can be processed simultaneously. The system executes multiple program instructions or sub-tasks at any moment in time. Total time taken will be much less than with a single compute resource.

#### *1.3.2.1 Distributed Computing Model*

A distributed computing model uses cloud, grid or clusters, which process and analyze big and large datasets on distributed computing nodes connected by high-speed networks. Table 1.2 gives the requirements of processing and analyzing big, large and small to medium datasets on distributed computing nodes. Big Data processing uses a parallel, scalable and no-sharing program model, such as MapReduce, for computations on it. (Chapter 2)

**Table** 1.2 Distributed computing paradigms

| Distributed computing on multiple processing nodes/ clusters | Big Data> IOM | Large datasets below 10 M | Small to medium datasets up to 1 M |
|---|---|---|---|
| Distributed computing | Yes | Yes | No |
| Parallel computing | Yes | Yes | No |
| Scalable computing | Yes | Yes | No |
| Shared nothing (No in-between data sharing and inter-processor communication) | Yes | Limited sharing | No |
| Shared in-between between the distributed nodes/ clusters | No | Limited sharing | Yes |

### 1.3.3 Cloud Computing

Wikipedia defines cloud computing as, "Cloud computing is a type of Internet-based computing that provides shared processing resources and data to the computers and other devices on demand."

One of the best approach for data processing is to perform parallel and distributed computing in a cloud-computing environment. Cloud usages circumvent the single point failure due to failing of one node. Cloud design performs as a whole. Its multiple nodes perform automatically and interchangeably. It offers high data security compared to other distributed technologies.

Cloud resources can be Amazon Web Service (AWS) Elastic Compute Cloud (EC2), Microsoft Azure or Apache CloudStack. Amazon Simple Storage Service (S3) provides simple web services interface to store and retrieve any amount of data, at any time, from anywhere on the web. [Amazon EC2 name possibly drives from the feature that EC2 has a simple web service interface, which provides and configures the storage and computing capacity with minimal friction].

Cloud computing features are: (i) on-demand service (ii) resource pooling, (iii) scalability, (iv) accountability, and (v) broad network access. Cloud services can be accessed from anywhere and at any time through the Internet. A local private cloud can also be set up on a local cluster of computers.

Cloud computing allows availability of computer infrastructure and services "on•demand" basis. The computing infrastructure includes data storage device, development platform, database, computing power or software applications.

Cloud services can be classified into three fundamental types:

Olot11d servkes ofifer, la:aS, Saas and Paas as service modalsforprooassinj; a11T1a a nalyzi11T1g time lar,ge datasets on complJJltiFllg lilooles.

1. Infrastructure as a Service (IaaS): Providing access to resources, such as hard disks, network connections, databases storage, data center and virtual server spaces is Infrastructure as a Service (IaaS). Some examples are Tata Communications, Amazon data centers and virtual servers. Apache CloudStack is an open source software for deploying and managing a large network of virtual machines, and offers public cloud services which provide highly scalable Infrastructure as a Service (IaaS).

2. Platform as a Service (PaaS): It implies providing the runtime environment to allow developers to build applications and services, which means cloud Platform as a Service. Software at the clouds support and manage the services, storage, networking, deploying, testing, collaborating, hosting and maintaining applications. Examples are Hadoop Cloud Service (IBM Biglnsight, Microsoft Azure HD Insights, Oracle BigData Cloud Services).

3. Software as a Service (Saas): Providing software applications as a service to end• users is known as Software as a Service. Software applications are hosted by a service provider and made available to customers over the Internet. Some examples are SQL GoogleSQL,IBM BigSQL, HPE Vertica, Microsoft Polybase and Oracle BigData SQL.

### 1.3.4 Grid and Cluster Computing

#### *Grid Computing*

*Grid Computing* refers to distributed computing, in which a group of computers from several locations are connected with each other to achieve a common task. The computer resources are heterogeneously and geographically disperse. A group of computers that might spread over remotely comprise a grid. A grid is used for a variety of purposes. A single grid of course, dedicates at an instance to a particular application only. Grid computing provides large-scale resource sharing which is flexible, coordinated and secure among its users. The users consist of individuals, organizations and resources.

Grid computing suits data-intensive storage better than storage of small objects of few millions of bytes. To achieve the maximum benefit from data grids, they should be used for a large amount of data which can distribute over grid nodes. Besides data grid, the other variation of grid, i.e., computational grid focuses on computationally intensive operations.

**Features of Grid Computing** Grid computing, similar to cloud computing, is scalable. Cloud computing depends on sharing of resources (for example, networks, servers, storage, applications and services) to attain coordination and coherence among resources similar to grid computing. Similarly, grid also forms a distributed network for resource integration.

**Drawbacks of Grid Computing** Grid computing is the single point, which leads to failure in case of underperformance or failure of any of the participating nodes. A system's storage capacity varies with the number of users, instances and the amount of data transferred at a given time. Sharing resources among a large number of users helps in reducing infrastructure costs and raising load capacities.

### *Cluster Computing*

A cluster is a group of computers connected by a network. The group works together to accomplish the same task. Clusters are used mainly for load balancing. They shift processes between nodes to keep an even load on the group of connected computers. Hadoop architecture uses the similar methods (Chapter 2).

Table 1.3 gives a comparison of grid computing with the distributed and cluster computing.

**Table 1.3** Grid computing and related paradigms

| Distributed computing | Cluster computing | Grid computing |
|---|---|---|
| • Loosely coupled<br>• Heterogeneous<br>• Single administration | • Tightly coupled<br>• Homogeneous<br>• Cooperative working | • Large scale<br>• Cross organizational<br>• Geographical distribution<br>• Distributed management |

### 1.3.5 VolunteerComputing

Volunteers provide computing resources to projects of importance that use resources to do distributed computing and/ or storage. **Volunteer computing** is a distributed computing paradigm which uses computing resources of the volunteers. Volunteers are organizations or members who own personal computers. **Projects** examples are science-related projects executed by universities or academia in general.

Some issues with volunteer computing systems are:

1. Volunteered computers heterogeneity
2. Drop outs from the network over time
3. Their sporadic availability

4. Incorrect results at volunteers are unaccountable as they are essentially from anonymous volunteers.

Self-Assessment Exercise linked to LO 1.2

1. Define analytics scalability, horizontal scalability and vertical scalability.
2. How does platform differ from software? When will a program use Saas and when PaaS?
3. List the features of grid computing. How does it differ from cluster and cloud computing?
4. Why do we use distributed computing for analytics of large datasets?

## 1.4 ! DESIGNING DATA ARCHITECTURE

The following subsections describe how to design Big Data architecture layers and how to manage data for analytics.

LO 1.3

IOesi g n of dab arrdh itect11J1,e layers and tlilleir fwnctio ns, and data mallilagtfililent functions for thee ana'lytlics

### 1.4.1 Data Architecture Design

Techopedia defines *Big Data architecture* as follows: "Big Data architecture is the logical and/ or physical layout/structure of how Big Data will be stored, accessed and managed within a Big Data or IT environment. Architecture logically defines how Big Data solution will work, the core components (hardware, database, software, storage) used, flow of information, security and more."

Characteristics of Big Data make designing Big Data architecture a complex process. Further, faster additions of new technological innovations increase the complexity in design. The requirements for offering competing products at lower costs in the market make the designing task more challenging for a Big Data architect.

Data analytics need the number of sequential steps. Big Data architecture design task simplifies when using the logical layers approach. Figure 1.2 shows the logical layers and the functions which are considered in Big Data architecture.

Five vertically aligned textboxes on the left of Figure 1.2 show the layers. Horizontal textboxes show the functions in each layer.

Data processing architecture consists of five layers: (i) identification of data sources, (ii) acquisition, ingestion, extraction, pre-processing, transformation of data, (iii) data storage at files, servers, cluster or cloud, (iv) data-processing, and (v) data consumption in the number of programs and tools.

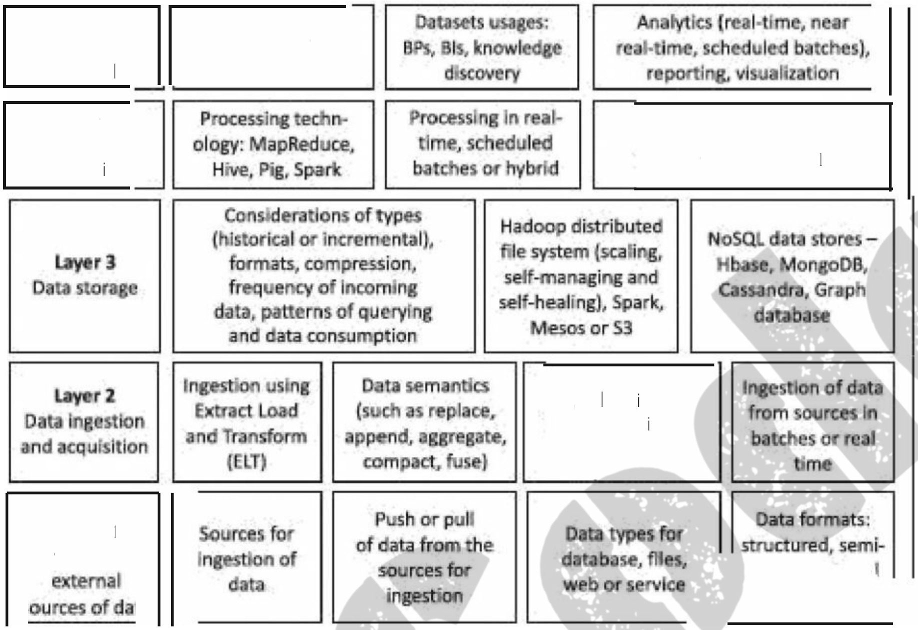

Datasets usages: BPs, BIs, knowledge discovery
Analytics (real-time, near real-time, scheduled batches), reporting, visualization
Processing techn-ology: MapReduce, Hive, Pig, Spark
Processing in real-time, scheduled batches or hybrid
Layer 3 Data storage
Considerations of types (historical or incremental), formats, compression, frequency of incoming data, patterns of querying and data consumption
Hadoop distributed file system (scaling, self-managing and self-healing), Spark, Mesos or S3
NoSQL data stores – Hbase, MongoDB, Cassandra, Graph database
Layer 2 Data ingestion and acquisition
Ingestion using Extract Load and Transform (ELT)
Data semantics (such as replace, append, aggregate, compact, fuse)
Ingestion of data from sources in batches or real time
external ources of da
Sources for ingestion of data
Push or pull of data from the sources for ingestion
Data types for database, files, web or service
Data formats: structured, semi-

**Figure** 1.2 Design of logical layers in a data processing architecture, and functions in the layers

Data consumed for applications, such as business intelligence, data mining, discovering patterns/ clusters, artificial intelligence (AI), machine learning (ML), text analytics, descriptive and predictive analytics, and data visualization.

Data ingestion, pre-processing, storage and analytics require special tools and technologies.

Logical layer 1 (LI) is for identifying data sources, which are external, internal or both. The layer 2 (LZ) is for data-ingestion.

Data ingestion means a process of absorbing information, just like the process of absorbing nutrients and medications into the body by eating or drinking them (Cambridge English Dictionary). Ingestion is the process of obtaining and importing data for immediate use or transfer. Ingestion may be in batches or in real time using pre-processing or semantics.

The L3 layer is for storage of data from the LZ layer. The L4 is for data processing using software, such as MapReduce, Hive, Pig or Spark. The top layer LS is for data consumption. Data is used in analytics, visualizations, reporting, export to cloud or web servers.

L1 considers the following aspects in a design:

Amount of data needed at ingestion layer 2 (L2)

Push from L1 or pull by L2 as per the mechanism for the usages

Source data-types: Database, files, web or service

Source formats, i.e., semi-structured, unstructured or structured.

L2 considers the following aspects:

Ingestion and ETL processes either in real time, which means store and use the data as generated, or in batches. Batch processing is using discrete datasets at scheduled or periodic intervals of time.

L3 considers the followings aspects:

Data storage type (historical or incremental), format, compression, incoming data frequency, querying patterns and consumption requirements for L4 or LS

Data storage using Hadoop distributed file system or NoSQL data stores-HBase, Cassandra, MongoDB.

L4 considers the followings aspects:

Data processing software such as MapReduce, Hive, Pig, Spark, Spark Mahout, Spark Streaming

Processing in scheduled batches or real time or hybrid

Processing as per synchronous or asynchronous processing requirements at LS.

LS considers the consumption of data for the following:

Data integration

Datasets usages for reporting and visualization

Analytics (real time, near real time, scheduled batches), BPs, Bis, knowledge discovery

Export of datasets to cloud, web or other systems

### 1.4.2 Managing Data for Analysis

Data managing means enabling, controlling, protecting, delivering and enhancing the value of data and information asset. Reports, analysis and visualizations need well• defined data. Data management also enables data usage in applications. The process for managing needs to be well defined for fulfilling requirements of the applications.

Data management functions include:

1. Data assets creation, maintenance and protection
2. Data governance, which includes establishing the processes for ensuring the availability, usability, integrity, security and high-quality of data. The processes enable trustworthy data availability for analytics, followed by the decision making at the enterprise.
3. Data architecture creation, modelling and analysis
4. Database maintenance, administration and management system. For example, RDBMS (relational database management system), NoSQL
5. Managing data security, data access control, deletion, privacy and security
6. Managing the data quality
7. Data collection using the ETL process
8. Managing documents, records and contents
9. Creation of reference and master data, and data control and supervision
10. Data and application integration
11. Integrated data management, enterprise-ready data creation, fast access and analysis, automation and simplification of operations on the data,
12. Data warehouse management
13. Maintenance of business intelligence
14. Data mining and analytics algorithms.

Self-Assessment Exercise linked to LO 1.3

1. How are data architecture layers used for analytics?
2. Explain the function of each of the five layers in Big Data architecture design (Figure 1.2).
3. List the functions of the ELT at data ingestion layer and at data storage layer.
4. List the functions in data management.

## 1.5 DATA SOURCES, QUALITY, PRE-PROCESSING AND STORING

The following subsections describe data sources, data quality, data pre-processing and data store export to the cloud.

LO 1.4

Data sources, data quality, data pre-processing, and data store export to the cloud

### 1.5.1 Data Sources

Applications, programs and tools use data. Sources can be *external*, such as sensors, trackers, web logs, computer systems logs and feeds. Sources can be machines, which source data from data-creating programs.

Data sources can be structured, semi-structured, multi-structured or unstructured. Data sources can be social media (L1 in Figure 1.2). A source can be *internal*. Sources can be data repositories, such as database, relational database, flat file, spreadsheet, mail server, web server, directory services, even text or files such as comma-separated values (CSV) files. Source may be a data store for applications (L4 in Figure 1.2).

#### *1.5.1.1 Structured Data Sources*

Data source for ingestion, storage and processing can be a file, database or streaming data. The source may be on the same computer running a program or a networked computer. Examples of structured data sources are SQL Server, MySQL, Microsoft Access database, Oracle DBMS, IBM DB2, Informix, Amazon SimpleDB or a file-collection directory at a server.

A data source name implies a defined name, which a process uses to identify the source. The name needs to be a meaningful name. For example, a name which identifies the stored data in student grades during processing. The data source name could be *StudentName_Data_ Grades.*

A *data dictionary* enables references for accesses to data. The dictionary consists of a set of master lookup tables. The dictionary stores at a central location. The central location enables easier access as well as administration of changes in sources. The name of the dictionary can be *UniversityStudents_DataPlusGrades.* A master-directory server can also be called *NameNode*.

Machine data sources and file data sources in Microsoft applications

Microsoft applications consider two types of sources for processing: (i) machine sources and (ii) file sources.[4]

(i) Machine sources are present on computing nodes, such as servers. A machine identifies a source by the user-defined name, driver-manager name and source-driver name.

(ii) File sources are stored files. An application executing the data, first *connects* to a driver manager of the source. A user, client or application *does not register* with the source, but *connects* to the manager when required. The process of connection is simple when using a file data source in case the file contains a connection string that would otherwise have to be built using a call to a connect-function driver.

Oracle applications consider two types of data sources: (i) *database,* which identifies the database information that the software needs to connect to database, and (ii) *logic-machine,* which identifies the machine which runs batches of applications and master business functions.[5] Source definition identifies the machine. The source can be on a network. The definition in that case also includes network information, such as the name of the server, which hosts the machine functions.

~~~!~~~:f~!~:~rc!s~~~s
iim Oracle applications

The applications consider data sources as the ones where the database tables reside and where the software runs logic objects for an enterprise. Data sources can point to:

1. A database in a specific location or in a data library of OS
2. A specific machine in the enterprise that processes logic
3. A data source master table which stores data source definitions. The table may be at a centralized source (enterprise server) or at server-map for the source.

A database can be in an IBM i data library" [IBM i is a computer operating system in which IBM i considers everything as an object, each possessing persistence. The system IBM i offers Unix-like file directories using an integrated file system.].

Specific database inst..ililce
on1le as data sourcasin
19 ~1 i sy ste rm

IBM applications consider data sources for applications and tools as one which identifies either (i) a specific database instance or (ii) file on a remote system that stores data.[6] Data sources can be shared. The access to source is restricted according to the roles assigned to both the source and the application that use it.

EXAMPLE 1.7

(i) How would you name the data sources of the student grade-sheets?

(ii) How does an analytics application (Analysis_APP)access student grade-sheet data source, using the Data Dictionary or data-source master-table for the grade-sheets of students?
~~~

(iii) How does the application connect and access the data source of students' grade-sheets?

Assume each student can have a grade-sheet for each of the six semesters in UG Computer Science programme.

SOLUTION

(i) Assume SemID is distinct key for a semester. StudID is a key assigned to a student, whether in CS or another subject, and whether in UG or PG. A StudID is unique. Data source can be file data source named 'UG_CS_Sem_StudID_Grades' for all UG CS student grades. UG_CS_Sem_StudID_Grades database consists of maximum six grade sheets UG_CS_SemID_StudID_Grades i.e., one for each semester. Assume that Analysis_APP does not connect or directly links to the data source UG_CS_Sem_StudID_Grades database. Then, the Analysis_APP links to a Data Dictionary or data source master table, which is data repository for the pointers of all six semesters of UG Computer Science program and other subject programs.

(ii) Assume that Analysis_APP associates to Oracle data-source master-table. The table stores the data-source definitions for all UG and PG, and all subjects and semester grades of the students. The data-source master-table stores the pointers of all semester grades. The table thus points to UG_CS_Sem_StudID_Grades DB for the student identified by StudID.

(iii) Assume that application deploys Microsoft DB. Then, first Analysis_APP links to a Driver Manager. The Driver Manager then calls the ODBC functions in the Driver Manager. The application identifies the target driver for the UG_CS_Sem_StudID_Grades data source with a connection handle. When the Driver Manager loads the driver, the Driver Manager builds a table of pointers to the functions in that driver. It uses the connection handle passed by the application to find the address of the function in the target driver and calls that function by address.

### *1.5.1.2 Unstructured Data Sources*

Unstructured data sources are distributed over high-speed networks. The data need high velocity processing. Sources are from distributed file systems. The sources are of file types, such as .txt (text file), .csv (comma separated values file). Data may be as key• value pairs, such as hash key-values pairs. Data may have internal structures, such as in e-mail, Facebook pages, twitter messages etc. The data do not model, reveal relationships, hierarchy relationships or object-oriented features, such as extensibility.

### 1.5.1.3 *Data Sources - Sensors, Signals and GPS*

The data sources can be sensors, sensor networks, signals from machines, devices, controllers and intelligent edge nodes of different types in the industry M2M communication and the GPS systems.

Sensors are electronic devices that sense the physical environment. Sensors are devices which are used for measuring temperature, pressure, humidity, light intensity, traffic in proximity, acceleration, locations, object(s) proximity, orientations and magnetic intensity, and other physical states and parameters. Sensors play an active role in the automotive industry.

RFIDs and their sensors play an active role in RFID based supply chain management, and tracking parcels, goods and delivery.

Sensors embedded in processors, which include machine-learning instructions, and wireless communication capabilities are innovations. They are sources in IoT applications.

## 1.5.2 Data Quality

Data quality is high when it represents the real-world construct to which references are taken. High quality means data, which enables all the required operations, analysis, decisions, planning and knowledge discovery correctly. A definition for high quality data, especially for artificial intelligence applications, can be data with five R's as follows: Relevancy, recency, range, robustness and reliability. Relevancy is of utmost importance.

!Dab gualityd'emnit:iions:
oata, relevanc, .irecency,
ralillge,rroblillstness and
.___irel__ia__b__ilify__i _

A uniform definition of data quality is difficult. A reference can be made to a set of values of quantitative or qualitative conditions, which must be specified to say that data quality is high or low.

### 1.5.2.1 *Data Integrity*

*Data integrity* refers to the maintenance of consistency and accuracy in data over its usable life. Software, which store, process, or retrieve the data, should maintain the integrity of data. Data should be incorruptible. For example, in Example 1.7 the grades of students should remain unaffected upon processing.

### 1.5.2.2 *Data Noise, Outliers, Missing and Duplicate Values*

Noise One of the factors effecting data quality is noise. Noise in data refers to data giving additional meaningless information besides true (actual/required) information. Noise refers to difference in the value measured from true value due to additional influences. Noisy data means data having large additional information. Result of data analysis is adversely affected due to noisy data.

Noise is random in character, which means frequency with which it occurs is variable over time. The values show nearly equal positive and negative deviations. A statistical analysis of deviation can select the noise in data and true values can be retrieved.

Outliers A factor which effects quality is an outlier. An *outlier* in data refers to data, which appears to not belong to the dataset. For example, data that is outside an expected range. Actual outliers need to be removed from the dataset, else the result will be effected by a small or large amount. Alternatively, if valid data is identified as outlier, then also the results will be affected. The outliers are a result of human data-entry errors, programming bugs, some transition effect or phase lag in stabilizing the data value to the true value.

Missing Values Another factor effecting data quality is missing values. *Missing value* implies data *not appearing in the data set.*

Duplicate Values Another factor effecting data quality is duplicate values. *Duplicate value* implies the same data appearing two or more times in a dataset.

The following example explains noise, outliers, missing values and duplicate data.

### EXAMPLE 1.8

Consider use cases of noise, outliers, missing values and duplicate data. Write the effect on the analysis in each case.

SOLUTION

Following are the examples of machine-generated data.

1. *Noise:* Recall WRMP organization for weather recording. Consider noise in wind velocity and direction readings due to external turbulences. The velocity at certain instances will appear too high and sometimes too low. The directions at certain instances will appear inclined more towards the north and sometimes more towards the south.
2. *Outlier:* Consider an outlier in the students' grade-sheets in one subject out of five in the fourth-semester result of a student. A result in a semester shows 9.0 out of 10 points in place of 3.0 out of 10. Data 9.0 is an outlier. The student semester grade point average (SGPA) will be erroneously declared and the student may even be declared to have failed in that semester.
3. *Missing values:* Consider missing values in the sales figures of chocolates. The values not sent for certain dates from an ACVM. This may be due to the failure

of power supply at the machine or network problems on specific days in a month. The chocolate sales not added for a day can be added in the next day's sales data. The effect over a month on the average sales per day is not significant. However, if the failure occurred on last day of a month, then the analysis will be erroneous.

4. ***Duplicate values:*** Consider duplicate values in the sales figures of chocolates from an ACVM. This may be due to some problem in the system. The number of duplicates for sales when sent and added, then sales result analysis will get affected. It can even result in false alarms to a service, which maintains the supply chain to the ACVMs.

   Assume network problems on certain instances. The ACVM may not get an acknowledgement of the sales figures from the server, leading to sending an incorrect sales record once again. If this happens then sales figures of chocolates get recorded twice at that instance. For example, if the chocolate sales data gets added twice in a specific day's sales data, the calculation of monthly sales data is adversely affected.

## 1.5.3 Data Pre-processing

Data pre-processing is an important step at the ingestion layer (Figure 1.2). For example, consider grade point data in Example 1.8. The outlier needs to be removed. Pre-processing is a must before data mining and analytics. Pre-processing is also a must before running a Machine Learning (ML) algorithm. Analytics needs prior screening of data quality also. Data when being exported to a cloud service or data store needs pre• processing.

Need of dab jpre-prucessingfoiraata store r.ortabilicy aimd '-!|sa1bilit};· in appli'ca;tions allild se11Vices

Pre-processing needs are:

(i) Dropping out of range, inconsistent and outlier values

(ii) Filtering unreliable, irrelevant and redundant information

(iii) Data cleaning, editing, reduction and/ or wrangling

(iv) Data validation, transformation or transcoding

(v) ELT processing.

***Data Cleaning***

*Data cleaning* refers to the process of removing or correcting incomplete, incorrect, inaccurate or irrelevant parts of the data after detecting them. For example, in Example

1.8 correcting the grade outliers or mistakenly entered values means cleaning and correcting the data.

**Data Cleaning Tools** Data cleaning is done before mining of data. Incomplete or irrelevant data may result into misleading decisions. It is not always possible to create well-structured data. Data can generate in a system in many formats when it is obtained from the web. Data cleaning tools help in refining and structuring data into usable data. Examples of such tools are OpenRefine and DataCleaner.

N@i:!dS of d~lt:1l pre• processilllg usl n g daan iliilQl, en ri ch ment, eoliti ng irech.L:1dlio1i1 al!lld "'1fi:lrng li n g rinet'hods

### *Data Enrichment*

Techopedia definition is as follows: *"Data enrichment* refers to operations or processes which refine, enhance or improve the raw data."

### *Data Editing*

*Data editing* refers to the process of reviewing and adjusting the acquired datasets. The editing controls the data quality. Editing methods are (i) interactive, (ii) selective, (iii) automatic, (iv) aggregating and (v) distribution.

### *Data Reduction*

*Data reduction* enables the transformation of acquired information into an ordered, correct and simplified form. The reductions enable ingestion of meaningful data in the datasets. The basic concept is the reduction of multitudinous amount of data, and use of the meaningful parts. The reduction uses editing, scaling, coding, sorting, collating, smoothening, interpolating and preparing tabular summaries.

### *Data Wrangling*

*Data wrangling* refers to the process of transforming and mapping the data. Results from analytics are then appropriate and valuable. For example, mapping enables data into another format, which makes it valuable for analytics and data visualizations.

### *Data Format used during Pre-Processing*

Examples of formats for data transfer from (a) data storage, (b) analytics application, (b) service or (d) cloud can be:

Need of dara fol'TillTlat conver• sion ot1 dlata ('SV, llSON, ke -v-aluep.riirs orotl'ii@r ct~ta from D:rta Store;: for example.. illil tihe form otrtables

(i) Comma-separated values CSV (Example 1.9)

(ii) Java Script Object Notation 0SON) as batches of object arrays or resource arrays (Example 3.3)

(iii) Tag Length Value (TLV)

(iv) Key-value pairs (Section 3.3.1)

(v) Hash-key-value pairs (Example 3.2).

### *CSVFormat*

An example is a table or Microsoft Excel file which needs conversion to CSV format. A student_record.xlsx converts to student_record.csv file. Comma-separated values (CSV) file refers to a plain text file which stores the table data of numbers and text. When processing for data visualization of Excel format file, the data conversion will be done from csv to xlsx format.

Each CSV file line is a data record. Each record consists of one or more fields, separated from each other by commas. RFC 4180 standard specifies the various specifications. A CSV file may also use space, tab or delimiter tab-separated formats for the values in the fields. This is a loose terminology. The following example explains the conversion process.

EXAMPLE 1.9

Consider the example of a table in a grade sheet. A CSV file is easily understandable when the table's first row specifies the column heads. Three columns of the first row are Subject Code, Subject Name and Grade and three columns of the second row are CSlOl, "Theory of Computations" and 7.8, as shown below:

| Subject Code | I I' | I I |
|---|---|---|
| CSHH | ' T~ol'y of Compl!litnitio.tlS ' | |
| CSl | ' Comp'lllter ArcIntectuse" | |

SOLUTION

The first and second lines in the CSV file are:

Subject Code, Subject Name, Grade

CS101, ""Theory of Computations?", 7.8.

CS102, ""Computer Architecture?", 7.8.

The two consecutive double-quotes mean that one of the double quotes is retained in the text "Theory of Computations". That one specifies that characters are inside the double quotes and represent a string.

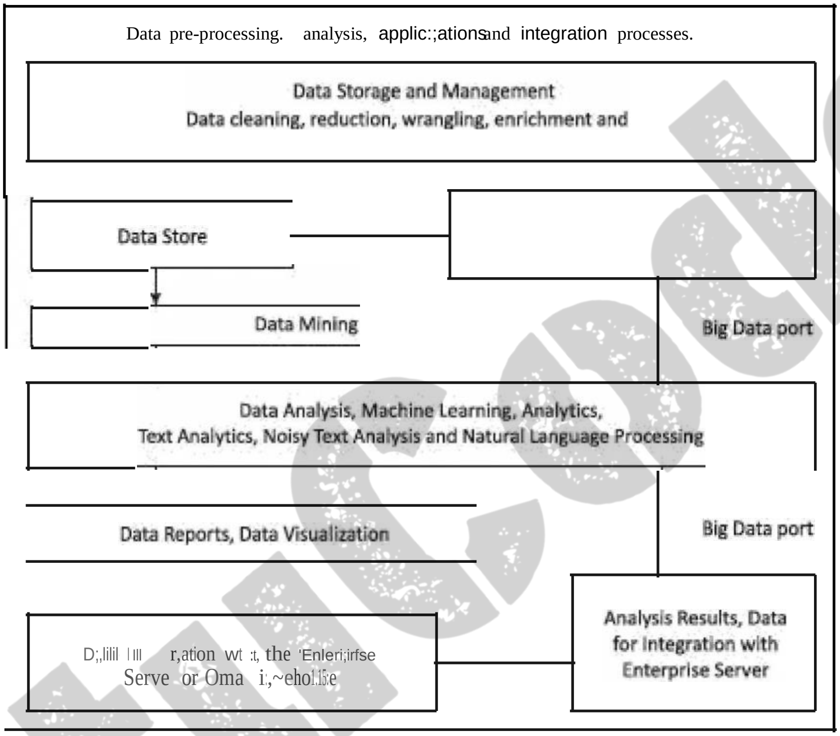
### *Data Format Conversions*

Transferring the data may need pre-processing for data-format conversions. Data sources store need portability and usability. A number of different applications, services and tools need a specific format of data only. Pre-processing before their usages or storage on cloud services is a must.

### 1.5.4 Data Store Export to Cloud

Figure 1.3 shows resulting data pre-processing, data mining, analysis, visualization and data store. The data exports to cloud services. The results integrate at the enterprise server or data warehouse.

Export: olfda:ta fio.rm dab sou 11ees to llli!M, ior:osofit, Oracle, Amamn, Raok~pac@ or Hadbop clo1.1d seNk:es



Data pre-processing. analysis, applic:;ationsand integration processes.
Data Storage and Management
Data cleaning, reduction, wrangling, enrichment and
Data Store
Data Mining
Big Data port
Data Analysis, Machine Learning, Analytics,
Text Analytics, Noisy Text Analysis and Natural Language Processing
Data Reports, Data Visualization
Big Data port
D;,lilil I lll r,ation wt :t, the 'Enleri;irfse
Serve or Oma i:,~ehol.l5:e
Analysis Results, Data for Integration with Enterprise Server

Figure 1.3 Data pre-processing, analysis, visualization, data store export

#### *1.5.4.1 Cloud Services*

Cloud offers various services. (Section 1.3.3) These services can be accessed through a cloud client (client application), such as a web browser, SQL or other client. Figure 1.4 shows data-store export from machines, files, computers, web servers and web services. The data exports to clouds, such as IBM, Microsoft, Oracle, Amazon, Rackspace, TCS, Tata Communications or Hadoop cloud services.

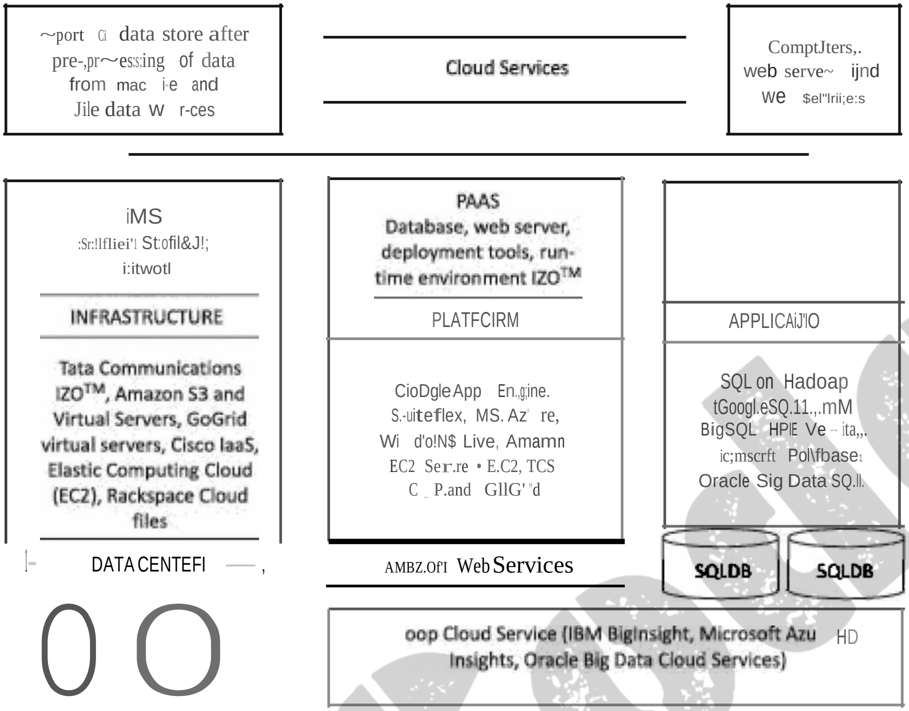

~port Ci data store after pre-,pr~es:s:ing of data from mac i-e and Jile data w r-ces
Cloud Services
ComptJters,. web serve~ ijnd we $el"Irii;e:s
iMS :Sr:!lfliei'! St:ofil&J!; i:itwotl
INFRASTRUCTURE
Tata Communications IZO™, Amazon S3 and Virtual Servers, GoGrid virtual servers, Cisco IaaS, Elastic Computing Cloud (EC2), Rackspace Cloud files
DATA CENTEFI
PAAS
Database, web server, deployment tools, run-time environment IZO™
PLATFCIRM
CioDgle App En.,g;ine. S.-uiteflex, MS. Az re, Wi d'o!N$ Live, Amamn EC2 Se:r.re • E.C2, TCS C P.and GllG' 'd
AMBZ.Of'I Web Services
APPLICAiJ'IO
SQL on Hadoap tGoogl.eSQ.11.,.mM BigSQL HPIE Ve-- ita,,. ic;mscrft Pol\fbase1 Oracle Sig Data SQ.ll.
SQLDB
SQLDB
oop Cloud Service (IBM BigInsight, Microsoft Azu HD Insights, Oracle Big Data Cloud Services)

Figure 1.4 Data store export from machines, files, computers, web servers and web services

### *1.5.4.2 Export of Data to AWS and Rackspace Clouds*

The following example explains the export processes to Amazon and Rackspace clouds.

EXAMPLE 1.10

(a) How do the rows in MySQL database table export to Amazon AWS?

(b) How do the rows in MySQL database table export to Rackspace?

SOLUTION

(a) Following are the steps for export to an EC2 instance:

(i) A process pre-processes the data from data-rows at table in MySQL database and creates a CSV file.

(ii) An EC2 instance provides an AWS data pipeline.

(iii)The CSV file exports to Amazon 53 using pipeline. The CSV file then copies into an 53 bucket. [7] Coping action deploys an EC2 instance.

(iv)AWS notification service (SNS) sends notification on completion.8

(b) Following are the steps for export to Rackspace[9]:

(i) An instance name has maximum 255 characters. One or more databases create a database instance. The process of creation can be configured to create an instance now or later. Each database can have a number of users.

(i) Default port number for binding of MySQL is port 3306.

(ii) A command *mysqldump - u root - p database_name > database_name.sql* exports to Rackspace cloud.

(iii)When a database is at a remote host then a command *mysqldump- h host_name - u user_name - p database_name > database_name.sql* exports to the cloud database.

Google cloud platform provides a cloud service called BigQuery.[10] Figure 1.5 shows BigQuery cloud service at Google cloud platform. The data exports from a table or partition schema, )SON, CSV or AVRO files from data sources after the pre-processing.

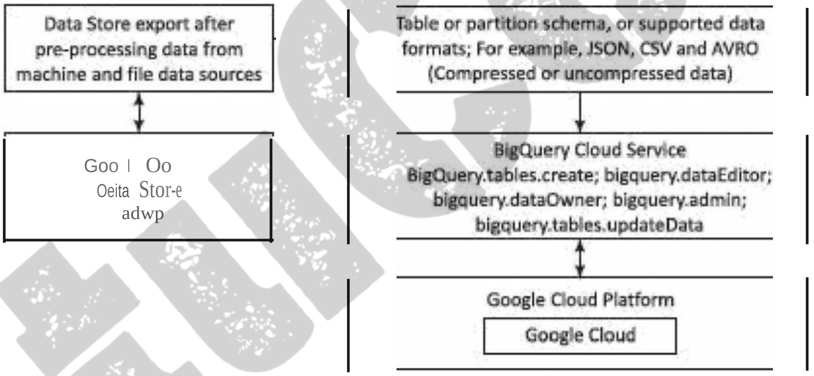

Data Store export after pre-processing data from machine and file data sources
Goo l Oo
Oeita Stor-e
adwp
Table or partition schema, or supported data formats; For example, JSON, CSV and AVRO (Compressed or uncompressed data)
BigQuery Cloud Service
BigQuery.tables.create; bigquery.dataEditor; bigquery.dataOwner; bigquery.admin; bigquery.tables.updateData
Google Cloud Platform
Google Cloud

Figure 1.5 BigQuery cloud service at Google cloud platform

Data Store first pre-processes from machine and file data sources. Pre-processing transforms the data in table or partition schema or supported data formats. For example, )SON, CSV and AVRO. Data then exports in compressed or uncompressed data formats. (Avro is a data serialization system in Hadoop related tools for Big Data.)

Cloud service BigQuery consists of bigquery.tables.create; bigquery.dataEditor; bigquery.dataOwner; bigquery.admin; bigquery.tables.updateData and other service functions. Analytics uses Google Analytics 360. BigQuery cloud exports data to a Google cloud or cloud backup only.

Self-Assessment Exercise linked to LO 1.4

1. Whyis data qualityimportant in discoveringnewknowledgeand decisionmaking?
2. Listthe examples of cloud services for exporting data stores.
3. Howis conversion to CSV file before data store beneficial? Howis conversion to tables from CSV files from data store beneficial?
4. List the usages of three types of services that clouds offer. List Big Data cloud services, to data sources export from data store, and perform cloud during analytics, visualizations and intelligence discovery.
5. Consider databases storing the daily sales figures of chocolates, such as KitKat, Milk, Fruit and Nuts, Nougat and Oreo, each at every machine in Example 1.6(i). How will you name the data sources in ACVMs analytics? How will the ACVMs sales be analyzed for each type of chocolate using the data-source master• tables?

## 1.6 ! DATA STORAGE AND ANALYSIS

The following subsections describe data storage and analysis, and comparison between Big Data management and analysis with traditional database management systems.

LO 1.5

Dab stor;,rige. imalysis. om ṕa nson tietweet!ll a,olitionnal systems.. such as ReJatio!ii1al 1Database Aan• igemel!ilt Syst'E!f.'iil r;RJDB 5), nt:eJprise serve!rs and cfab warehm!lse, Approacliiles for Big Data storage, process• iln11; and an\:l'ly;tlcs

### 1.6.1 Data Storage and Management: Traditional Systems

#### 1.6.1.1 *Data Store with Structured or Semi-Structured Data*

Traditional systems use structured or semi-structured data. The following example explains the sources and data store of structured data.

EXAMPLE 1.11

What are the sources of structured data store?

SOLUTION

The sources of structured data store are:

Traditional relational database-management system (RDBMS) data, such as MySQL DB2, enterprise server and data warehouse

Business process data which stores business events, such as registering a customer, taking an order, generating an invoice, and managing products in pre-defined formats. The data falls in the category of highly structured data. The data consists of transaction records, tables, relationships and metadata that build the information about the business data.

Commercial transactions

Banking/ stock records

E-commerce transactions data.

The following example explains the sources and data store of semi-structured data.

EXAMPLE 1.12

Give examples of sources of data store of semi-structured data.

SOLUTION

Examples of semi-structured data are:

XML and JSON semi-structured documents[7·8]

A comma-separated values (CSV) file. The CSV stores tabular data in plain text. Each line is a data record. A record can have several fields, each filed separated by a comma. Structured data, such as database include multiple relations but CSV does not consider the relations in a single CSV file. CSV cannot represent object-oriented databases or hierarchical data records. A CSV file is as follows:

```
Preeti,1995,MCA,Object  Oriented Prograrnrning,8.75
Kirti,2010, M.Tech., Mobile Operating System, 8.5
```

Data represent the data records for columns and rows of a table. Each row has names, year of passing, degree name, course name and grade point out of 10. Rows are separated by a new line and the columns by a comma.

*JSON* Object Data Formats: CSV does not represent object-oriented records, databases or hierarchical data records. *JSON* and XML represent semi-structured data and represent object-oriented and hierarchical data records. Example 3.5 explains CSV and JSON objects and the hierarchical data records in the JSON file format.

### *1.6.1.2 SQL*

An RDBMS uses SQL (Structured Query Language). SQL is a language for viewing or changing (update, insert or append or delete) databases. It is a language for data access control, schema creation and data modifications.

SQL was originally based on the tuple relational calculus and relational algebra. SQL can embed within other languages using SQL modules, libraries and pre-compilers. SQL does the following:

1. *Create schema*, which is a structure which contains description of objects (base tables, views, constraints) created by a user. The user can describe the data and define the data in the database.
2. *Create catalog,* which consists of a set of schemas which describe the database.
3. *Data Definition Language* (DDL) for the commands which depicts a database, that include creating, altering and dropping of tables and establishing the constraints. A user can create and drop databases and tables, establish foreign keys, create view, stored procedure, functions in the database etc.
4. *Data Manipulation Language* (DML) for commands that maintain and query the database. A user can manipulate (INSERT/UPDATE) and access (SELECT) the data.
5. *Data Control Language* (DCL) for commands that control a database, and include administering of privileges and committing. A user can set (grant, add or revoke) permissions on tables, procedures and views.

SQL is a language for managing the RDBMS. A relational DB is a collection of data in multiple tables, which relate to each other through special fields, called keys (primary key, foreign key and unique key). Relational databases provide flexibilities. Relational database examples are MySQL PostGreSQL Oracle database, Informix, IBM DB2 and Microsoft SQL server.

### 1.6.1.3 Large Data Storage usingRDBMS

RDBMS tables store data in a structured form. The tables have rows and columns. Data management of Data Store includes the provisions for privacy and security, data integration, compaction and fusion. The systems use machine-generated data, human-sourced data, and data from business processes (BP) and business intelligence (BI).

lirradlitiolillal systetlilils Relati CWTtal aatalbase is rnlliediollil *ot* dl:ab ilillt0 llilllll!lltiple t.ables *vih* iohi rel.ates to each other tmirou gh special iffieldis, ca ll~d keys

A set of keys and relational keys access the fields at tables, and retrieve data using queries (insert, modify, append, join or delete). RDBMSs use software for data administration also.

Online content associated with Practice Exercise 1.12 describes the use of tables in relational databases in detail.

### *1.6.1.4 Distributed Database Management System*

A distributed DBMS (DDBMS) is a collection of logically interrelated databases at multiple system over a computer network. The features of a distributed database system are:

> U.shlbutoo ml! is ai colle:tion Olrlogit-ally interrel'atte.d databases at mlllltiple systenris over a comptrter network.

1. A collection of logically related databases.
2. Cooperation between databases in a transparent manner. Transparent means that each user within the system may access all of the data within all of the databases as if they were a single database.
3. Should be 'location independent' which means the user is unaware of where the data is located, and it is possible to move the data from one physical location to another without affecting the user.

### *1.6.1.5 In-Memory Column Formats Data*

A columnar format in-memory allows faster data retrieval when only a few columns in a table need to be selected during query processing or aggregation. Data in a column are kept together in-memory in columnar format. A single memory access, therefore, loads many values at the column. An address increment to a next memory address for the next value is fast when compared to first computing the address of the next value, which is not the immediate next address. The following example explains the in• memory columnar format.

EXAMPLE 1.13

Consider analysis of monthly sales of chocolates on ACVMs (Example 1.6) in company's annual profit reports.

(i) How does sales analysis become easy in-memory columnar format?

(ii) How does during an analysis the access is made to few columns in place of from entire datasets?

SOLUTION

All the column 1 values for several days' record is physically together in-memory at consecutive addresses. All the column 2 values are then physically together at the next successive addresses. Then, the column 3 and other columns store at the columnar database in-memory.

The data stores for each record order in successive columns, so that the lOOth entry at column 1 and the lOOth entry for column 2 belong to the same record and same input accessible from a single row-key. Column vector refers to a vector whose elements are values at column fields.

Analytics, therefore, can be executed faster when data is in the column format, and more rows and few columns need to be selected during analysis. Successive days' sales of each flavour of chocolate stores in successive values in one column from row r to (r + 30) in a month, thirty row-keys for 30 days, and 365 row keys in a year.

Aggregation functions and other analysis functions are easy to run due to successive memory addresses for sales for each day for each flavour. Examples of aggregation functions are sum, count, maximum, minimum, average, minimum and maximum deviation from a specified value.

*Online Analytical Processing* (OLAP) in real-time transaction processing is fast when using in-memory column format tables. OLAP enables real-time analytics. The CPU accesses all columns in a single instance of access to the memory in columnar format in•memory data-storage.

Online Analytical Processing (OLAP) enables online viewing of analyzed data and visualization up to the desired granularity (fineness or coarseness) enables view by rolling up (finer granulates to coarse granulates data) or drilling down (coarser granulates data to finer granulates). OLAP enables obtaining online summarized information and automated reports for a large database.

Metadata describes the data. Pre-storing of calculated values provide consistently fast response. Result formats from the queries are based on Metadata.

### *1.6.1.6 In-Memory* **Row** *Format Databases*

A row format in-memory allows much faster data processing during OLTP (online transaction processing). Refer Example 1.13. Each row record has corresponding values in multiple columns and the on-line values store at the consecutive memory addresses in row format. A specific day's sale of five different chocolate flavours is stored in consecutive columns c to c+S at memory. A single instance of memory accesses loads values of all five flavours at successive columns during online processing. For example, the total number of chocolates sold computes online. Data is in-memory row-formats in stream and event analytics. The stream analytics method does continuous computation that happens as data is flowing through the system. Event analytics does computation on event and use event data for tracking and reporting events.

### *1.6.1.7 EnterpriseData-Store Server and Data Warehouse*

Enterprise data, after *data cleaning* process, integrate with the server data at warehouse. Enterprise data server use data from several distributed sources which store data using various technologies. All data *merge* using an integration tool. Integration enables collective viewing of the datasets at the data warehouse (Figure 1.3).

Enterprise data integration may also include integration with application(s), such as analytics, visualization, reporting, business intelligence and knowledge discovery. Heterogeneous systems execute complex integration processes when integrating at an enterprise server or data warehouse. Complex application-integration means the integration of heterogeneous application architectures and processes with the databases at the enterprise. Enterprise data warehouse store the databases, and data stores after integration, using tools from number of sources.

Online contents associated with Practice Exercises 1.9 and 1.10 give details of commercial solutions for complex application-integration of processes.

Following are some standardised business processes, as defined in the Oracle application-integration architecture:

1. Integrating and enhancing the existing systems and processes
2. Business intelligence
3. Data security and integrity
4. Newbusiness services/products (Web services)
5. Collaboration/knowledge management
6. Enterprise architecture/SOA
7. e-commerce
8. External customer services
9. Supply chain automation/visualization
10. Data centre optimization

Figure 1.6 shows Steps 1 to 5 in enterprise data integration and management with Big Data for high performance computing using local and cloud resources for analytics, applications and services.

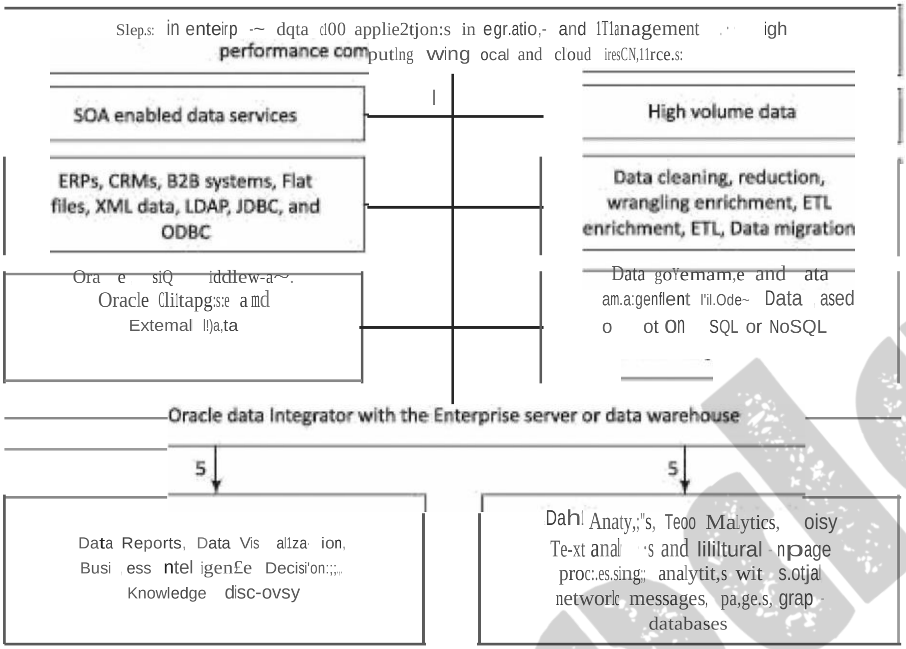

Steps in enterprise data integration and management with high performance computing using local and cloud resources
SOA enabled data services
ERPs, CRMs, B2B systems, Flat files, XML data, LDAP, JDBC, and ODBC
Oracle middleware, Oracle Clitapps and External Data
High volume data
Data cleaning, reduction, wrangling enrichment, ETL enrichment, ETL, Data migration
Data governance and data management Data based on SQL or NoSQL
Oracle data Integrator with the Enterprise server or data warehouse
5
5
Data Reports, Data Visualization, Business Intelligence Decisions, Knowledge discovery
Data Analytics, Text analytics, Text analysis and multilingual webpage processing, analytics with social network messages, pages, graph databases

**Figure 1.6** Steps 1 to 5 in Enterprise data integration and management with Big• Data for high performance computing using local and cloud resources for the analytics, applications and services

## 1.6.2 Big Data Storage

Following subsections describe Big Data storage concepts:

### *1.6.2.1 Big Data NoSQL or Not Only SQL*

NoSQL databases are considered as semi-structured data. Big Data Store uses NoSQL. NOSQL stands for No SQL or Not Only SQL. The stores do not integrate with applications using SQL. NoSQL is also used in cloud data store. Features ofNoSQLare as follows:

NoSQL or Not Only SQL class of non-relational data storage systems, flexible data models and multiple schemas

1. It is a class of non-relational data storage systems, and the flexible data models and multiple schema:

(i) Class consisting of uninterrupted key/value or big hash table [Dynamo (Amazon 53)]

(ii) Class consisting of unordered keys and usingJSON (PNUTS)

(iii)Class consisting of ordered keys and semi-structured data storage systems [BigTable,Cassandra (used in Facebook/Apache) and HBase] (iv)Class consisting ofJSON (MongoDB)

(v) Class consisting of name/value in the text (CouchDB)

(vi)May not use fixed table schema

(vii)Do not use the JOINS

(viii)Data written at one node can replicate at multiple nodes, therefore Data storage is fault-tolerant,

(ix) May relax the ACID rules during the Data Store transactions.

(x) Data Store can be partitioned and follows CAP theorem (out of three properties, consistency, availability and partitions, at least two must be there during the transactions)

*Consistency* means all copies have the same value like in traditional DBs. *Availability* means at least one copy is available in case a partition becomes inactive or fails. For example in web applications, the other copy in other partition is available. *Partition* means parts which are active but may not cooperate as in the distributed DBs.

### *1.6.2.2 Coexistence of Big Data, NoSQLand Traditional Data Stores*

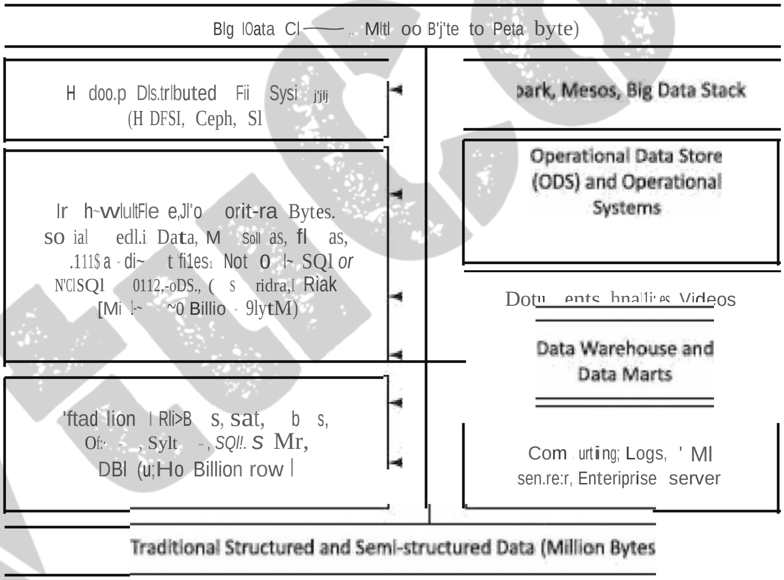
Figure 1.7 shows co-existence of data at server, SQL, RDBMS with NoSQL and Big Data at Hadoop, Spark, Meses, 53 or compatible Clusters.

Table 1.4 gives various data sources for Big Data along with its examples of usages and the tools used.

**Table 1.4** Various data sources and examples of usages and tools

| Data Source | Examples of Usages | Example of Tools |
|---|---|---|
| Relational databases | Managing business applications involving structured data | Microsoft Access, Oracle, IBM DB2, SQL Server, MySQL, PostgreSQL Composite, SQL on Hadoop [HPE (Hewlett Packard Enterprise) Vertica, IBM BigSQL, Microsoft Polybase, Oracle Big Data SQL] |
| Analysis databases (MPP, columnar, In-memory) | High performance queries and analytics | Sybase IQ, Kognitio, Terradata, Netezza, Vertica, ParAccel, ParStream, Infobright, Vectorwise, |
| NoSQL databases (Key-value pairs, Columnar format, documents, | Key-value pairs, fast read/write using collections of name-value pairs for storing any type of data; Columnar format, documents, | Key-value pair databases: Riak DS (Data Store), OrientDB, Column format databases (HBase, Cassandra), Document oriented databases: CouchDB, MongoDB; Graph |

| | | |
|---|---|---|
| Objects, graph) | objects, graph DBs and DSs | databases (Neo4j, Tetan) |
| Hadoop clusters | Ability to process large data sets across a distributed computing environment | Cloudera, Apache HDFS |
| Web applications | Access to data generated from web applications | Google Analytics, Twitter |
| Cloud data | Elastic scalable outsourced databases, and data administration services | Amazon Web Services, Rackspace, GoogleSQL |
| Individual data | Individual productivity | MS Excel, CSV, TLV,JSON,MIME type |
| Multidimensional | Well-defined bounded exploration especially popular for financial applications | Microsoft SQL Server Analysis Services |
| Social media data | Text data, images, videos | Twitter, Linkedin |



Blg l0ata Cl—— Mltl oo B'j'te to Peta byte)
H doo.p Dls.trlbuted Fii Sysi j'jlj
(H DFSI, Ceph, Sl
Ir h~vvlultFle e,Jl'o orit-ra Bytes.
so ial edl.i Data, M Soll as, fl as,
.111$ a - di~ t fi1es Not 0 l~ SQl or
N'ClSQl 0112,-oDS., ( s ridra,l Riak
[Mi l~ ~0 Billio - 9lytM)
'ftad lion l Rli>B s, sat, b s,
Of:· - , Sylt - , SQl!. s Mr,
DBl (u;Ho Billion row l
park, Mesos, Big Data Stack
Operational Data Store (ODS) and Operational Systems
Dotu ents hnalli es Videos
Data Warehouse and Data Marts
Com urting; Logs, ' Ml
sen.re:r, Enteriprise server
Traditional Structured and Semi-structured Data (Million Bytes

**Figure 1.7** Coexistence ofRDBMSfor traditional server data, NoSQL and Hadoop, Spark and compatible BigData Clusters

### 1.6.3 Big Data Platform

A Big Data platform supports large datasets and volume of data. The data generate at a higher velocity, in more varieties or in higher veracity. Managing Big Data requires large resources of MPPs, cloud, parallel processing and specialized tools. Bigdata platform should provision tools and services for:

1. storage, processing and analytics,
2. developing, deploying, operating and managing Big Data environment,
3. reducing the complexity of multiple data sources and integration of applications into one cohesive solution,
4. custom development, querying and integration with other systems, and
5. the traditional as well as Big Data techniques.

Data management, storage and analytics of Big data captured at the companies and services require the following:

1. New innovative non-traditional methods of storage, processing and analytics
2. Distributed Data Stores
3. Creating scalable as well as elastic virtualized platform (cloud computing)
4. Huge volume of Data Stores
5. Massive parallelism
6. High speed networks
7. High performance processing, optimization and tuning
8. Data management model based on Not Only SQL or NoSQL
9. In-memory data column-formats transactions processing or *dual in-memory data* columns as well as row formats for OLAP and OLTP
10. Data retrieval, mining, reporting, visualization and analytics
11. Graph databases to enable analytics with social network messages, pages and data analytics
12. Machine learning or other approaches
13. Big data sources: Data storages, data warehouse, Oracle Big Data, MongoDBNoSQL, Cassandra NoSQL
14. Data sources: Sensors, Audit trail of Financial transactions data, external data such as Web, Social Media, weather data, health records data.

### *1.6.3.1 Hadoop*

Big Data platform consists of Big Data storage(s), server(s) and data management and business intelligence software. Storage can deploy Hadoop Distributed File System (HDFS), NoSQL data stores, such as HBase, MongoDB,Cassandra. HDFS system is an open source storage system. HDFS is a scaling, self-managing and self-healing file system.

The Hadoop system packages application-programming model. Hadoop is a scalable and reliable parallel computing platform. Hadoop manages Big Data distributed databases. Figure 1.8 shows Hadoop based Big Data environment. Small height cylinders represent MapReduce and big ones represent the Hadoop.

### *1.6.3.2 Mesos*

Mesos v0.9 is a resources management platform which enables sharing of cluster of nodes by multiple frameworks and which has compatibility with an open analytics stack [data processing (Hive, Hadoop, HBase, Storm), data management (HDFS)].

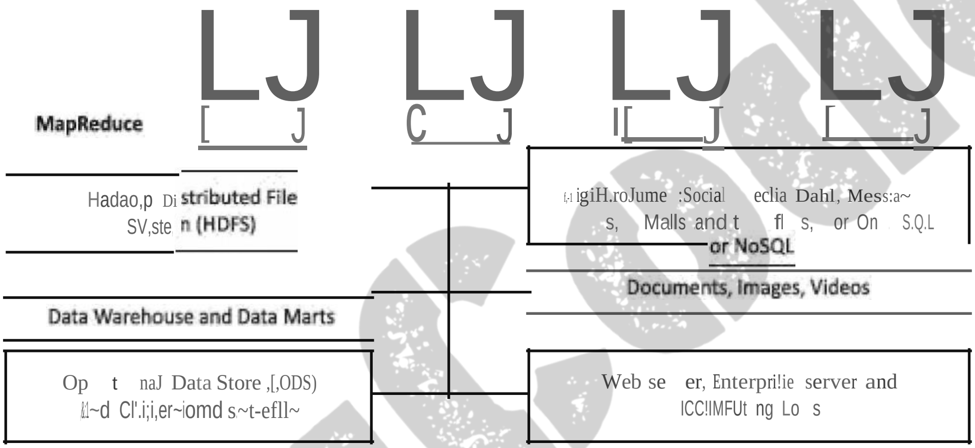

MapReduce
Hadao,p Di stributed File SV,ste n (HDFS)
Data Warehouse and Data Marts
Op t naJ Data Store ,[,ODS)
igiH.roJume :Social eclia Dahl, Mess:a~ s, Malls and t fl s, or On S.Q.L or NoSQL
Documents, Images, Videos
Web se er, Enterpri!ie server and ICC!IMFUt ng Lo s

**Figure 1.8** Hadoop based Big Data environment

### *1.6.3.3 Big Data Stack*

A stack consists of a set of software components and data store units. Applications, machine-learning algorithms, analytics and visualization tools use Big Data Stack (BDS) at a cloud service, such as Amazon EC2, Azure or private cloud. The stack uses cluster of high performance machines.

Table 1.5 gives Big Data management, storage and processing tools.

**Table 1.5** Tools for Big Data environment

| Types | Examples |
|---|---|
| MapReduce | Hadoop, Apache Hive, Apache Pig, Cascading, Cascalog, mrjob (Python MapReduce library), Apache 54, MapR, Apple Acunu, Apache Flume, Apache Kafka |
| NoSQL Databases | MongoDB, Apache CouchDB, Apache Cassandra, Aerospike, Apache HBase, Hypertable |

| | |
|---|---|
| Processing | Spark, IBM BigSheets, PySpark, R, Yahoo! Pipes, Amazon Mechanical Turk, Datameer, Apache Solr /Lucene, ElasticSearch |
| Servers | Amazon EC2, 53, GoogleQuery, Google App Engine, AWS Elastic Beanstalk, Salesforce Heroku |
| Storage | Hadoop Distributed File System, Amazon 53, Mesos |

## 1.6.4 Big Data Analytics

DBMS or RDBMS manages the traditional databases. Data analysis need pre-processing of raw data and gives information useful for decision making. Analysis brings order, structure and meaning to the collection of data. Data is collected and analyzed to answer questions, test the hypotheses or disprove theories.

### *1.6.4.1 Data Analytics Definition*

Data Analytics can be formally defined as the statistical and mathematical data analysis that clusters, segments, ranks and predicts future possibilities. An important feature of data analytics is its predictive, forecasting and prescriptive capability. Analytics uses historical data and forecasts new values or results. Analytics suggests techniques which will provide the most efficient and beneficial results for an enterprise. Data analysis helps in finding business intelligence and helps in decision making.

Data analysis can be defined as,

> "Analysis of data is a process of inspecting, cleaning, transforming and modeling data with the goal of discovering useful information, suggesting conclusions and supporting decision making." *(Wikipedia)*

### *1.6.4.2 Phases in Analytics*

Analytics has the following phases before deriving the new facts, providing business intelligence and generating new knowledge.

1. *Descriptive analytics* enables deriving the additional value from visualizations and reports
2. *Predictive analytics* is advanced analytics which enables extraction of new facts and knowledge, and then predicts/forecasts
3. *Prescriptive analytics* enable derivation of the additional value and undertake better decisions for new option(s) to maximize the profits
4. *Cognitive analytics* enables derivation of the additional value and undertake better

decisions.

Analytics integrates with the enterprise server or data warehouse.

Figure 1.9 shows an overview of a reference model for analytics architecture. The figure also shows on the right-hand side the Big Data file systems, machine learning algorithms and query languages and usage of the Hadoop ecosystem.

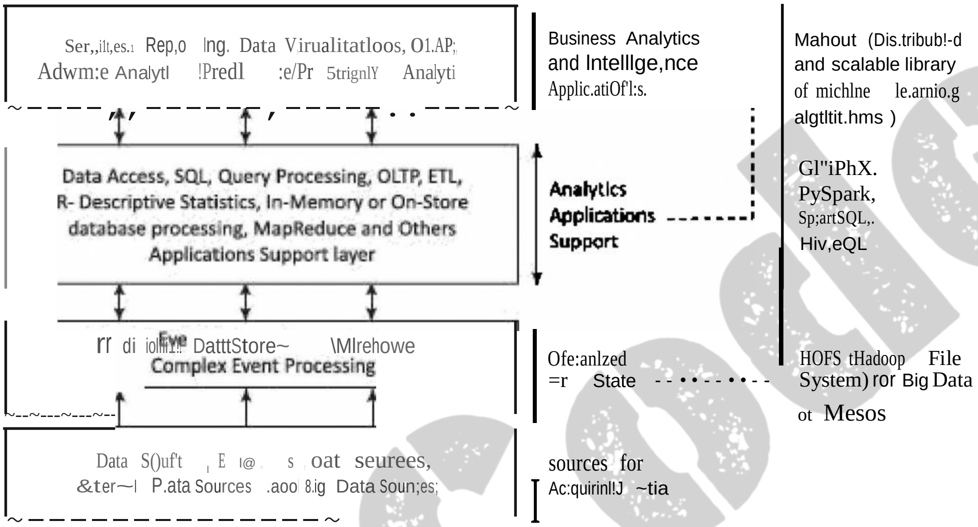

Ser,,ilt,es.1 Rep,o lng. Data Virualitatloos, O1.AP;
Adwm:e Analytl !Predl :e/Pr 5trignlY Analyti
Business Analytics and Intelllge,nce Applic.atiOf'l:s.
Mahout (Dis.tribub!-d and scalable library of michlne le.arnio.g algtltit.hms )
Data Access, SQL, Query Processing, OLTP, ETL, R- Descriptive Statistics, In-Memory or On-Store database processing, MapReduce and Others Applications Support layer
Analytics Applications Support
Gl"iPhX. PySpark, Sp;artSQL,. Hiv,eQL
rr di iolffi1.!! DatttStore~ \Mlrehowe
Eve Complex Event Processing
Ofe:anlzed =r State
HOFS tHadoop File System) ror Big Data
ot Mesos
Data S()uf't E l@ s oat seurees, &ter~l P.ata Sources .aoo 8.ig Data Soun;es;
sources for Ac:quirinl!J ~tia

Figure 1.9 Traditional and Big Data analytics architecture reference model

The captured or stored data require a well-proven strategy to calculate, plan or analyze. When Big Data combine with high-powered data analysis, enterprise achieve valued business-related tasks. Examples are:

Determine root causes of defects, faults and failures in minimum time.

Deliver advertisements on mobiles or web, based on customer's location and buying habits.

Detect offender before that affects the organization or society.

### *1.6.4.3 Berkeley Data Analytics Stack (BDAS)*

The importance of Big Data lies in the fact that what one does with it rather than how big or large it is. Identify whether the gathered data is able to help in obtaining the following findings: 1) cost reduction, 2) time reduction, 3) new product planning and development, 4) smart decision making using predictive analytics and 5) knowledge discovery.

lmAs co111sistingOHhe ~raprocmfnq.~ra ma1r11a9emrnem;andresol!!lrc.e mana9elllllem: layers

Big Data analytics need innovative as well as cost effective techniques. BOAS is an open• source data analytics stack for complex computations on Big Data.11 It supports efficient, large-scale in-memory data processing, and thus enables user applications achieving three fundamental processing requirements; accuracy, time and cost.

Berkeley Data Analytics Stack (BDAS) consists of data processing, data management and resource management layers. Followinglist these:

1. Applications, AMP-Genomicsand Carat run at the BOAS. Data processing software component provides in-memory processing which processes the data efficiently across the frameworks. AMP stands for Berkeley's Algorithms, Machines and Peoples Laboratory.
2. Data processing combines *batch*, *streaming* and *interactive* computations.
3. Resource management software component provides for sharing the infrastructure across various frameworks.

Figure 1.10 shows a four layers architecture for Big Data Stack that consists of Hadoop, MapReduce, Spark core and SparkSQL,Streaming, R, Graphx, MLib, Mahout, Arrow and Kafka.

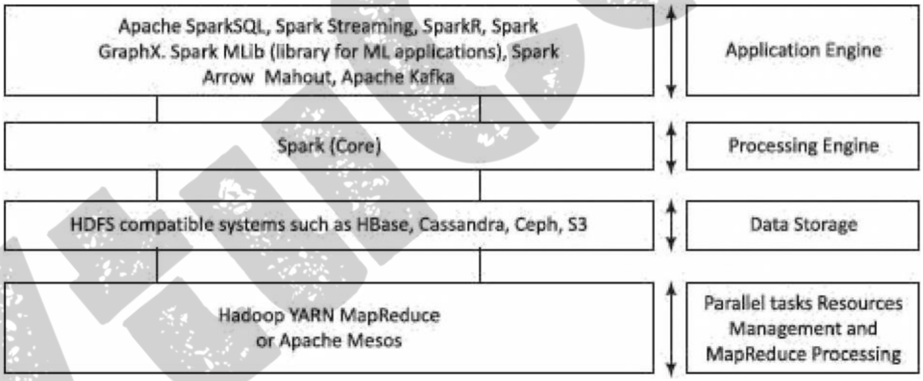

Apache SparkSQL, Spark Streaming, SparkR, Spark GraphX. Spark MLib (library for ML applications), Spark Arrow Mahout, Apache Kafka
Application Engine
Spark (Core)
Processing Engine
HDFS compatible systems such as HBase, Cassandra, Ceph, S3
Data Storage
Hadoop YARN MapReduce or Apache Mesos
Parallel tasks Resources Management and MapReduce Processing

Figure 1.10 Four layers architecture for BigData Stack consisting of Hadoop, MapReduce, Spark core and SparkSQL,Streaming, R, GraphX,MLib, Mahout, Arrow and Kafka

Self-Assessment Exercise linked to LO 1.5

1. What are the traditional systems for data storage? How does in-memory columnar format help in OLAP? Give an example.
2. What are hierarchical and object oriented records?

3. What is enterprise server? How does enterprise server data store differ from a web server?
4. What are the functions of data integration software? How does application integration along with data integration help in business processes, intelligence and analytics?
5. What are the functions in SQL? List the differences between SQL data store and NoSQL data store.
6. How does a Big Data stack help in analytics tasks?
7. How does a Berkeley Data analytics stack help in analytics tasks?

## 1.71 BIG DATA ANALYTICS APPLICATIONS AND CASE STUDIES

Many applications such as social network and social media, cloud applications, public and commercial web sites, scientific experiments, simulators and e-government services generate Big Data. Big Data analytics find applications in many areas. Some of the popular ones are marketing, sales, health care, medicines, advertising etc. Following subsections describe these use cases, applications and case studies.

LO 1.6

Big [)ata lllse cas@S, a~p1icath:ms and c21se srudiii:!S in VariOlJ!JS fiie:lcJs, such as marketil!llg, sales and heallltcarre

### 1.7.1 Big Data in Marketing and Sales

Data are important for most aspect of marketing, sales and advertising. Customer Value (CV) depends on three factors - quality, service and price. Big data analytics deploy large volume of data to identify and derive intelligence using predictive models about the individuals. The facts enable marketing companies to decide what products to sell.

A definition of marketing is the creation, communication and delivery of *value* to customers. Customer (desired) value means what a customer desires from a product. Customer (perceived) value means what the customer believes to have received from a product after purchase of the product. Customer value analytics (CVA) means analyzing what a customer really needs. CVA makes it possible for leading marketers, such as Amazon to deliver the consistent customer experiences. Following are the five application areas in order of the popularity of Big Data use cases:

1. CVA using the inputs of evaluated purchase patterns, preferences, quality, price and

post sales servicing requirements

2. Operational analytics for optimizing company operations
3. Detection of frauds and compliances
4. New products and innovations in service
5. Enterprise data warehouse optimization.

An example of fraud is borrowing money on already mortgage assets. Example of timely compliances means returning the loan and interest installments by the borrowers.

A few examples in service-innovation are as follows: A company develops software and then offers services like Uber. Another example is of a company which develops software for hiring services, and then offers costly construction machinery and equipment. That service company might be rendering the services by hiring themselves from the multiple sources and locations of big construction companies.

Big data is providing marketing insights into (i) most effective content at each stage of a sales cycle, (ii) investment in improving the customer relationship management (CRM), (iii) addition to strategies for increasing customer lifetime value (CLTV), (iv) lowering of customer acquisition cost (CAC). Cloud services use Big Data analytics for CAC, CLTV and other metrics, the essentials in any cloud-based business

Big Data revolutionizes a number of areas of marketing and sales. Louis Columbus12 recently listed the ways of usages. (Refer online content for solution of Practice Exercise 1.14.)

Contextual marketing means using an online marketing model in which a marketer sends to potential customers the targeted advertisements, which are based on the search terms during latest browsing patterns usage by customers.

For example, if a customer is searching an airline for flights on a specific date from Delhi to Bangalore, then a smart travel agency targeting that customer through advertisements will show him/her, at specific intervals, better options for another airline or different but cheap dates for travel or options in which price reduction occurs gradually.

The following example explains the use of search engine optimization.

EXAMPLE 1.14

Why does the search engine at a company product website of a travel agency need optimization?

SOLUTION

Consider a travel agency website offers search results for flights between two destinations *A* and C, which do not connect directly. The search shows the results in order of increasing travel cost through stopover at an intermediate airport *B.* Assume that search results show up just mechanically, without embedding intelligence and optimization. The customers find uncomfortable solutions with such searches. The searches show the cheaper options but sometimes show results such as the customer would reach C through stopover at B after 8 hours or even sometimes on the next day.

The searches at that travel agency do not consider stopover options at different Bs, options available in different airlines to cut short travel time from B to C at cheaper costs, or newly introduced flights. The searches therefore need optimization for parameters of travel cost, multiple intermediate stopovers and airlines that will provide maximum customer convenience as well as cost.

Big data algorithms and advanced analytics techniques enable price optimization for a given product or service, and pricing decisions, especially in the commodity driven industries where products are inelastic. Inelastic product means a situation in which the service, required quantity or supply of a product remains unaffected by the price changes.

### 1.7.1.1 *Big Data Analytics in Detection of Marketing Frauds*

Fraud detection is vital to prevent financial loses to users. Fraud means someone deceiving deliberately. For example, mortgaging the same assets to multiple financial institutions, compromising customer data and transferring customer information to third party, falsifying company information to financial institutions, marketing product with compromising quality, marketing product with service level different from the promised, stealing intellectual property, and much more.

Big dah d~lO}lfflOOt in
ifira111a cM:ection relatedlto
____m__a,__ke__t1__ng__

Big Data analytics enable fraud detection. Big Data usages has the following features-for enabling detection and prevention of frauds:

1. Fusing of existing data at an enterprise data warehouse with data from sources such as social media, websites, biogs, e-mails, and thus enriching existing data
2. Using multiple sources of data and connecting with many applications
3. Providing greater insights using querying of the multiple source data
4. Analyzing data which enable structured reports and visualization
5. Providing high volume data mining, new innovative applications and thus leading to new business intelligence and knowledge discovery

6. Making it less difficult and faster detection of threats, and predict likely frauds by using various data and information publicly available.

### *1.7.1.2 Big Data Risks*

Large volume and velocity of Big Data provide greater insights but also associate risks with the data used. Data included may be erroneous, less accurate or far from reality. Analytics introduces new errors due to such data.

> Ei.gi Data large vol umil!', veloGit.y :and verac'iity data provide gmmr in~ g hts for r:datl secu1nity, pliivacy, 1mcre21si ng costs, and ba.dl a na tymcs a nd bad data 1nisks a.sscx:ia teis

Big Data can cause potential harm to individuals. For example, when someone puts false or distorted data about an individual in a blog, Facebook post, WhatsApp groups or tweets, the individual may suffer loss of educational opportunity, job or credit for his/her urgent needs. A company may suffer financial losses.

Five data risks, described by Bernard Marr are data security, data privacy breach, costs affecting profits, bad analytics and bad data.13 (Solutions in online content accompanying the book for Practice Exercise 1.15)

Companies need to take risks of using Big Data and design appropriate risk management procedures. They have to implement robust risk management processes and ensure reliable predictions. Corporate, society and individuals must act with responsibility.

### *1.7.1.3 Big Data Credit Risk Management*

Financial institutions, such as banks, extend loans to industrial and household sectors. These institutions in many countries face credit risks, mainly risks of (i) loan defaults, (ii) timely return of interests and principal amount. Financing institutions are keen to get insights into the following:

1. Identifying high credit rating business groups and individuals,
2. Identifying risk involved before lending money
3. Identifying industrial sectors with greater risks
4. Identifying types of employees (such as daily wage earners in construction sites) and businesses (such as oil exploration) with greater risks
5. Anticipating liquidity issues (availability of money for further issue of credit and rescheduling credit installments) over the years.

The insight using Big Data decreases the default rates in returning of loan, greater accuracy in issuing credit and faster identification of the non-payment or fraud issues of the loan receiving entities. (Example of fraud is using the same assets for drawing credit from two or more institutions or hiding earlier outstanding loans and loan defaults.)

One innovative way to manage credit risks and liquidity risks is use of available data and Big Data. High volume of data analysis gives greater insight into the default patterns, emerging patterns and thus credit risks.

Big Data analytics monitors social media, interactions data, contact addresses, mobile numbers, website, financial status, activities or job changes to find the emerging credit risk that may affect a customer loan returning capacity. Digital footprints across social media provide a valuable alternative data source for credit risk analysis. The data companies assist in rating the customer in application processing and also during the period of repayment of a loan. Friends on Facebook and their credit rating, comments and assets posted also help in determining the risks.

The data insights from the analytics lead to credit and liquidity risk management and faster reactions. Three benefits are (i) minimize the non-payments and frauds, (ii) identifying new credit opportunities, new customers and revenue streams, thereby broadening the company high credit rating customers base and (iii) marketing to low risk businesses and households.

### *1.7.1.4 Big Data And Algorithmic Trading*

Wikipedia gives a definition of algorithm trading as follows: "Algorithmic trading is a method of executing a large order (too large to fill all at once) using automated pre• programmed trading instructions accounting for variables such as time, price and volume." Complex mathematics computations enable algorithmic trading and business investment decisions to buy and sell. The input data are insights gathered from the risk analysis of market data. Big data bigger volume, velocity and variety in the trading provide an edge over other trading entities

## 1.7.2 Big Data and Healthcare

Big Data analytics in health care use the following data sources: (i) clinical records, (ii) pharmacy records, (3) electronic medical records (4) diagnosis logs and notes and (v) additional data, such as deviations from person usual activities, medical leaves from job, social interactions. Healthcare analytics using Big Data can facilitate the following:

Eig Oat:a large volume. velocity and veracity data proviale grener insights in, health caire s'!{S!erm,s and mooiGili1le

1. Provisioning of value-based and customer-centric healthcare,
2. Utilizing the 'Internet of Things' for health care
3. Preventing fraud, waste, abuse in the healthcare industry and reduce healthcare costs (Examples of frauds are excessive or duplicate claims for clinical and hospital treatments. Example of waste is unnecessary tests. Abuse means unnecessary use of medicines, such as tonics and testing facilities.)

4. Improving outcomes
5. Monitoring patients in real time.

*Value-based and customer-centric healthcare* means cost effective patient care by improving healthcare quality using latest knowledge, usages of electronic health and medical records and improving coordination among the healthcare providing agencies, which reduce avoidable overuse and healthcare costs.

*Healthcare Internet of Things* create unstructured data. The data enables the monitoring of the devices data for patient parameters, such as glucose, BP, ECGs and necessities of visiting physicians.

*Prevention of fraud, waste, and abuse* uses Big Data predictive analytics and help resolve excessive or duplicate claims in a systematic manner. The analytics of patient records and billing help in detecting, anomalies such as overutilization of services in short intervals, different hospitals in different locations simultaneously, or identical prescriptions for the same patient filed from multiple locations.

Improving outcomes is possible by accurately diagnosing patient conditions, early diagnosis, predicting problems such as congestive heart failure, anticipating and avoiding complications, matching treatments with outcomes and predicting patients at risk for disease or readmission.

*Patient real-time monitoring* uses machine learning algorithms which process real-time events. They provide physicians the insights to help them make life-saving decisions and allow for effective interventions. The process automation sends the alerts to care providers and informs them instantly about changes in the condition of a patient.

### 1.7.3 Big Data in Medicine

Big Data analytics deploys large volume of data to identify and derive intelligence using predictive models about individuals. Big Data driven approaches help in research in medicine which can help tpatients. Big Data offers potential to transform medicine and the healthcare system-Dr. Eric Schadt and Sastry Chilukuri.14

Following are some findings: building the health profiles of individual patients and predicting models for diagnosing better and offer better treatment,

1. Aggregating large volume and variety of information around from multiple sources the DNAs, proteins, and metabolites to cells, tissues, organs, organisms, and ecosystems, that can enhance the understanding of biology of diseases. Big data creates patterns and models by data mining and help in better understanding and research,

2. Deploying wearable devices data, the devices data records during active as well as inactive periods, provide better understanding of patient health, and better risk profiling the user for certain diseases,

### 1.7.4 Big Data in Advertising

The impact of Big Data is tremendous on the digital advertising industry. The digital advertising industry sends advertisements using SMS, e-mails, WhatsApp, Linkedln, Facebook, Twitter and other mediums.

Eig data ir,,9:;;il time a ngly.tics foir faster insights, emergil"llg trends ancll patterns, .illltdl gain actiollilab'le ilil1Siglrrrt:s for facing corn petitiolillS fi:"orin similarpiroducts in oligibl adverlDisi ng and bl!.llilcdillltg r@latioliilslrriips

Big Data technology and analytics provide insights, patterns and models, which relate the media exposure of all consumers to the purchase activity of all consumers using multiple digital channels. Big Data help in identity management and can provide an advertising mix for building better branding exercises.

Big Data captures data of multiple sources in large volume, velocity and variety of data unstructured and enriches the structured data at the enterprise data warehouse. Big data real time analytics provide emerging trends and patterns, and gain actionable insights for facing competitions from similar products. The data helps digital advertisers to discover new relationships, lesser competitive regions and areas.

Success from advertisements depend on collection, analyzing and mining. The new insights enable the personalization and targeting the online, social media and mobile for advertisements called hyper-localized advertising.

Nielson Inc. CEO, Mitch Barns described Big Data's big impact on the future of advertising. Advertising nowadays limits no longer to TV, radio and print. Advertisers use along with these multiple devices and mediums. For example, advertisement of the introduction of new courses by an institution or introduction of new flights by an Airline needs media other than TV and requires targeted and cost effective solutions.

Advertising on digital medium needs optimization. Too much usage can also effect negatively. Phone calls, SMSs, e-mail-based advertisements can be nuisance if sent without appropriate researching on the potential targets. The analytics help in this direction. The usage of Big Data after appropriate filtering and elimination is crucial enabler of BigData Analytics with appropriate data, data forms and data handling in the right manner.