G TEC COMPUTER EDUCATION
www.gteceducation.com

G NEWS
The voice of G-TEC

G-TEC COLLEGE FOR ADVANCED STUDIES GCAS
www.gteccollege.com

G-TEC GENSMART ACADEMY
www.gensmartacademy.com

# Big Data

## Introduction

**Big data** is a blanket term for the non-traditional strategies and technologies needed to collect, organize, process, and gather insights from large datasets. While the problem of working with data that exceeds the computing power or storage of a single computer is not new, the pervasiveness, scale, and value of this type of computing has greatly expanded in recent years.

In this article, we will talk about big data on a fundamental level and define common concepts you might come across while researching the subject. We will also take a high-level look at some of the processes and technologies currently being used in this space.

## What Is Big Data?

An exact definition of "big data" is difficult to nail down because projects, vendors, practitioners, and business professionals use it quite differently. With that in mind, generally speaking, **big data** is:

- Large datasets
- The category of computing strategies and technologies that are used to handle large datasets

In this context, "large dataset" means a dataset too large to reasonably process or store with traditional tooling or on a single computer. This means that the common scale of big datasets is constantly shifting and may vary significantly from organization to organization.

## Examples Of Big Data

Following are some the examples of Big Data-

The New York Stock Exchange generates about one terabyte of new trade data per day.

## Social Media

The statistic shows that 500+terabytes of new data get ingested into the databases of social media site Facebook, every day. This data is mainly generated in terms of photo and video uploads, message exchanges, putting comments etc.

A single Jet engine can generate 10+terabytes of data in 30 minutes of flight time. With many thousand flights per day, generation of data reaches up to many Petabytes.

## Types Of Big Data

BigData' could be found in three forms:

1. **Structured**
2. **Unstructured**
3. **Semi-structured**

## Structured

Any data that can be stored, accessed and processed in the form of fixed format is termed as a 'structured' data. Over the period of time, talent in computer science has achieved greater success in developing techniques for working with such kind of data (where the format is well known in advance) and also deriving value out of it. However, nowadays, we are foreseeing issues when a size of such data grows to a huge extent, typical sizes are being in the rage of multiple zettabytes.

### Examples Of Structured Data

An 'Employee' table in a database is an example of Structured Data

| Employee_ID | Employee_Name | Gender | Department | Salary_In_lacs |
|---|---|---|---|---|
| 2365 | Rajesh Kulkarni | Male | Finance | 650000 |
| 3398 | Pratibha Joshi | Female | Admin | 650000 |
| 7465 | Shushil Roy | Male | Admin | 500000 |
| 7465 | Shubhojit Das | Male | Finance | 500000 |
| 7465 | Priya Sane | Female | Finance | 550000 |

## Unstructured

Any data with unknown form or the structure is classified as unstructured data. In addition to the size being huge, un-structured data poses multiple challenges in terms of its processing for deriving value out of it. A typical example of unstructured data is a heterogeneous data source containing a combination of simple text files, images, videos etc. Now a days organizations have wealth of data available with them but unfortunately, they don't know how to derive value out of it since this data is in its raw form or unstructured format.

### Examples Of Un-structured Data

The output returned by 'Google Search'

## Semi-Structured

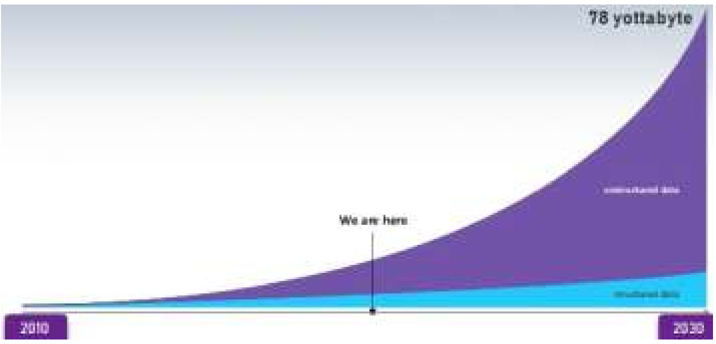
Semi-structured data can contain both the forms of data. We can see semi-structured data as a structured in form but it is actually not defined with e.g. a table definition in relational DBMS. Example of semi-structured data is a data represented in an XML file.

Examples Of Semi-structured Data

Personal data stored in an XML file-

```
<rec><name>Prashant Rao</name><sex>Male</sex><age>35</age></rec>
<rec><name>Seema  R.</name><sex>Female</sex><age>41</age></rec>
<rec><name>Satish  Mane</name><sex>Male</sex><age>29</age></rec>
<rec><name>Subrato  Roy</name><sex>Male</sex><age>26</age></rec>
<rec><name>Jeremiah J.</name><sex>Male</sex><age>35</age></rec>
```

## Data Growth over the years



## Why Are Big Data Systems Different?

The basic requirements for working with big data are the same as the requirements for working with datasets of any size. However, the massive scale, the speed of ingesting and processing, and the characteristics of the data that must be dealt with at each stage of the process present significant new challenges when designing solutions. The goal of most big data systems is to surface insights and connections from large volumes of heterogeneous data that would not be possible using conventional methods.In 2001, Gartner's Doug Laney first presented what became known as the "three Vs of big data" to describe some of the characteristics that make big data different from other data processing:

### Volume

The sheer scale of the information processed helps define big data systems. These datasets can be orders of magnitude larger than traditional datasets, which demands more thought at each stage of the processing and storage life cycle.

Often, because the work requirements exceed the capabilities of a single computer, this becomes a challenge of pooling, allocating, and coordinating resources from groups of computers. Cluster management and algorithms capable of breaking tasks into smaller pieces become increasingly important.

### Velocity

Another way in which big data differs significantly from other data systems is the speed that information moves through the system. Data is frequently flowing into the system from multiple sources and is often expected to be processed in real time to gain insights and update the current understanding of the system.

This focus on near instant feedback has driven many big data practitioners away from a batch-oriented approach and closer to a real-time streaming system. Data is constantly being added, massaged, processed, and analyzed in order to keep up with the influx of new information and to surface valuable information early when it is most relevant. These ideas require robust systems with highly available components to guard against failures along the data pipeline.

### Variety

Big data problems are often unique because of the wide range of both the sources being processed and their relative quality.

Data can be ingested from internal systems like application and server logs, from social media feeds and other external APIs, from physical device sensors, and from other providers. Big data seeks to handle potentially useful data regardless of where it's coming from by consolidating all information into a single system.

The formats and types of media can vary significantly as well. Rich media like images, video files, and audio recordings are ingested alongside text files, structured logs, etc. While more traditional data processing systems might expect data to enter the pipeline already labeled, formatted, and organized, big data systems usually accept and store data closer to its raw state. Ideally, any transformations or changes to the raw data will happen in memory at the time of processing.

### Other Characteristics

Various individuals and organizations have suggested expanding the original three Vs, though these proposals have tended to describe challenges rather than qualities of big data. Some common additions are:

- **Veracity**: The variety of sources and the complexity of the processing can lead to challenges in evaluating the quality of the data (and consequently, the quality of the resulting analysis)
- **Variability**: Variation in the data leads to wide variation in quality. Additional resources may be needed to identify, process, or filter low quality data to make it more useful.
- **Value**: The ultimate challenge of big data is delivering value. Sometimes, the systems and processes in place are complex enough that using the data and extracting actual value can become difficult.

## What Does a Big Data Life Cycle Look Like?

So how is data actually processed when dealing with a big data system? While approaches to implementation differ, there are some commonalities in the strategies and software that we can talk about generally. While the steps presented below might not be true in all cases, they are widely used.

The general categories of activities involved with big data processing are:

- Ingesting data into the system
- Persisting the data in storage
- Computing and Analysing data
- Visualizing the results

Before we look at these four workflow categories in detail, we will take a moment to talk about **clustered computing**, an important strategy employed by most big data solutions. Setting up a computing cluster is often the foundation for technology used in each of the life cycle stages.

Big data is a broad, rapidly evolving topic. While it is not well-suited for all types of computing, many organizations are turning to big data for certain types of workloads and using it to supplement their existing analysis and business tools. Big data systems are uniquely suited for surfacing difficult-to-detect patterns and providing insight into behaviours that are impossible to find through conventional means. By correctly implement systems that deal with big data, organizations can gain incredible value from data that is already available.

# Hadoop

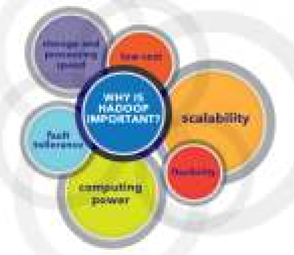
Hadoop is an open-source software framework for storing data and running applications on clusters of commodity hardware. It provides massive storage for any kind of data, enormous processing power and the ability to handle virtually limitless concurrent tasks or jobs.

## Hadoop History

As the World Wide Web grew in the late 1900s and early 2000s, search engines and indexes were created to help locate relevant information amid the text-based content. In the early years, search results were returned by humans. But as the web grew from dozens to millions of pages, automation was needed. Web crawlers were created, many as university-led research projects, and search engine start-ups took off (Yahoo, AltaVista, etc.).

One such project was an open-source web search engine called Nutch  the brainchild of Doug Cutting and Mike Cafarella. They wanted to return web search results faster by distributing data and calculations across different computers so multiple tasks could be accomplished simultaneously. During this time, another search engine project called Google was in progress. It was based on the same concept – storing and processing data in a distributed, automated way so that relevant web search results could be returned faster.

In 2006, Cutting joined Yahoo and took with him the Nutch project as well as ideas based on Google's early work with automating distributed data storage and processing. The Nutch project was divided – the web crawler portion remained as Nutch and the distributed computing and processing portion became Hadoop (named after Cutting's son's toy elephant). In 2008, Yahoo released Hadoop as an open-source project. Today, Hadoop's framework and ecosystem of technologies are managed and maintained by the non-profit Apache Software Foundation (ASF), a global community of software developers and contributors.



## Why is Hadoop important?

- **Ability to store and process huge amounts of any kind of data, quickly**. With data volumes and varieties constantly increasing, especially from social media and the Internet of Things (IoT), that's a key consideration.
- **Computing power**. Hadoop's distributed computing model processes big data fast. The more computing nodes you use; the more processing power you have.
- **Fault tolerance**. Data and application processing are protected against hardware failure. If a node goes down, jobs are automatically redirected to other nodes to make sure the distributed computing does not fail. Multiple copies of all data are stored automatically.
- **Flexibility**. Unlike traditional relational databases, you don't have to pre-process data before storing it. You can store as much data as you want and decide how to use it later. That includes unstructured data like text, images and videos.
- **Low cost**. The open-source framework is free and uses commodity hardware to store large quantities of data.
- **Scalability**. You can easily grow your system to handle more data simply by adding nodes. Little administration is required.

**MapReduce programming is not a good match for all problems**. It's good for simple information requests and problems that can be divided into independent units, but it's not efficient for iterative and interactive analytic tasks. MapReduce is file-intensive. Because the nodes don't intercommunicate except through sorts and shuffles, iterative algorithms require multiple map-shuffle/sort-reduce phases to complete. This creates multiple files between MapReduce phases and is inefficient for advanced analytic computing.

**There's a widely acknowledged talent gap**. It can be difficult to find entry-level programmers who have sufficient Java skills to be productive with MapReduce. That's one reason distribution providers are racing to put relational (SQL) technology on top of Hadoop. It is much easier to find programmers with SQL skills than MapReduce skills. And, Hadoop administration seems part art and part science, requiring low-level knowledge of operating systems, hardware and Hadoop kernel settings.

**Data security**. Another challenge centres around the fragmented data security issues, though new tools and technologies are surfacing. The Kerberos authentication protocol is a great step toward making Hadoop environments secure.

**Full-fledged data management and governance**. Hadoop does not have easy-to-use, full-feature tools for data management, data cleansing, governance and metadata. Especially lacking are tools for data quality and standardization.

## GST COUNCIL GIVES FIRMS MORE FLEXIBILITY ON USE OF INPUT TAX CREDIT

Any company would now be eligible to use credit available against paid integrated GST (IGST) to set off tax liabilities of state GST (SGST) and central GST (CGST) in any proportion and in any order

In yet another simplification, the Goods and Services Tax (GST) Council has added flexibility into the way a company can utilise the available input tax credit. Any company would now be eligible to use credit available against paid integrated GST (IGST) to set off tax liabilities of state GST (SGST) and central GST (CGST) in any proportion and in any order, the GST Council said in a circular sent to field formations on Tuesday.

Previously, the order of using the IGST credit was kept flexible — it was the company's choice to set off CGST or SGST first — in a notification dated March 29. However, it was not clear whether a company would be able to use IGST credit to set off SGST liability and CGST liability partially at the same time. It was construed that if a company chooses to set off SGST liability first, it would have to exhaust the entire SGST liability before using the IGST credit to set off CGST liability.

In yet another simplification, the Goods and Services Tax (GST) Council has added flexibility into the way a company can utilise the available input tax credit. Any company would now be eligible to use credit available against paid integrated GST (IGST) to set off tax liabilities of state GST (SGST) and central GST (CGST) in any proportion and in any order, the GST Council said in a circular sent to field formations on Tuesday.

Previously, the order of using the IGST credit was kept flexible — it was the company's choice to set off CGST or SGST first — in a notification dated March 29. However, it was not clear whether a company would be able to use IGST credit to set off SGST liability and CGST liability partially at the same time. It was construed that if a company chooses to set off SGST liability first, it would have to exhaust the entire SGST liability before using the IGST credit to set off CGST liability.

## FINANCE MINISTRY SIMPLIFIES 'SELF-ASSESSED' TO FILE GST COMPOSITION SCHEME

Businesses had to use a seven-page form called GSTR-4 when they filed tax returns every quarter if they opted for composition scheme.

Businesses that use the composition scheme to pay GST can file 'self-assessed tax' return on a quarterly basis in a simplified form, the finance ministry has said.

Businesses had to use a seven-page form called GSTR-4 when they filed tax returns every quarter if they opted for composition scheme, which is meant for small taxpayers and allows them relief from GST formalities.

As per a Central Board of Indirect Taxes and Customs (CBIC) notification, composition scheme taxpayers will now file GSTR-4 annually by April 30 for the previous financial year ending March 31.

The CBIC has notified the simplified 'statement for payment of self-assessed tax' in Form GST CMP08 to be filed by taxpayers who have opted for composition scheme, under which businesses have to pay lower rate of tax on their turnover.

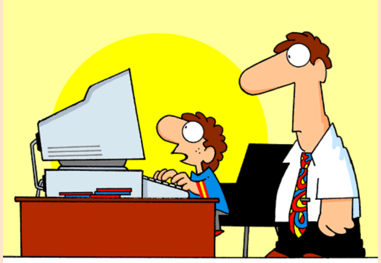
The CMP08, which has to be filed by the 18th day of the subsequent month following the end of a quarter, will include details like outward supplies, inward supplies attracting reverse charge including import of services; tax, interest payable; and taxes and interest paid.

Composition scheme businesses will file the April-June quarter returns in July as per the new format.

Small traders and manufacturers with a turnover of Rs 1.5 crore pay a 1 per cent GST, while service providers and suppliers of both goods and services up to a turnover of Rs 50 lakh pays 6 per cent.

Businesses who have not opted for composition scheme have to file GST returns every month and also pay taxes as per the GST slabs decided for the goods and services they deal in. Currently there is a 4-tier GST- 5, 12, 18 and 28 per cent.

### Smile Corner



"What program does God use to upload the sun every morning?"

G NEWS

Publisher
Mehroof I. Manalody

Chief Editor
S. Thulaseedharan Pillai
AGM - Operations

Editor
Joseph M Thomas
HOD - Academics

Email: gnews@gteceducation.com For internal circulation only

# OUR PRIDE

Congratulations
TALLY ACE TOPPERS MARCH 2019
FIRST
SECOND
THIRD
AYANA GRACE PAUL
ALBIN POULOSE

Congratulations
TALLY GCC VAT TOPPERS MARCH 2019
FIRST
SECOND
THIRD
MUHAMMED ANWAR C P
ANN SEBASTIAN
AFSAL P.A
FASEELA
KAVYA MOHAN M

61st IAB - UK
INTERNATIONAL EXAMINATION TOPPERS
Congratulations
FIRST
SECOND
THIRD
DISNI C SAJI
IAB-DIPA 1269363
G-TEC KUNNAMKULAM
THRISSUR
STEPHY.T.R
IAB-DIPA 1290267
G-TEC GURUVAYOOR
THRISSUR
HASNA.P.N
IAB-DIPA 1269705
G-TEC CHAVAKKAD
THRISSUR
Whole G-TEC family appreciate your achievement and recognize the effort and dedication of G-TEC staff members who helped you earn this.
Team G-TEC

Certified Ethical Hacker
Congratulations
G TEC COMPUTER EDUCATION
TOPPERS
TOPPERS
Mohammed Renais K K
Abhilash Vijay
EC-Council

G TEC COMPUTER EDUCATION
PDIFAS
FEB 2019
Congratulations
EXAMINATION TOPPERS
FIRST
SECOND
SAJILA.T
REENU JAYAN
ASHWINI.K
THIRD
ABHINAV SHAJI
ANJALI A
JESS MON THOMAS
KAVYA KRISHNAN
LIVIN ANTONY
SAPARNA K
SHANU B K
SINDHU.J
MOHAMMED NAJEEB K S
SHIMNA M P
GOPIKA.K.S
MFAS
EXAMINATION TOPPERS
91%
ALBIN ANTONY
FEB - 2019
G-TEC MUVATTUPUZHA, EKM
REG No. 1226565
Whole G-TEC family appreciate your achievement and recognize the effort and dedication of G-TEC staff members who helped you earn this.
Team G-TEC

MTA EXAMINATION TOPPERS
Congratulations
G TEC COMPUTER EDUCATION
Introduction to Programming using HTML and CSS
MTA
HTML & CSS
Introduction to Programming using Java
MTA
JAVA
Introduction to Programming using Python
MTA
Python
Database Fundamentals
MTA
Database
Windows® Operating System Fundamentals
MTA
Operating System
Software Development Fundamentals
MTA
Software
Microsoft
Technology Associate