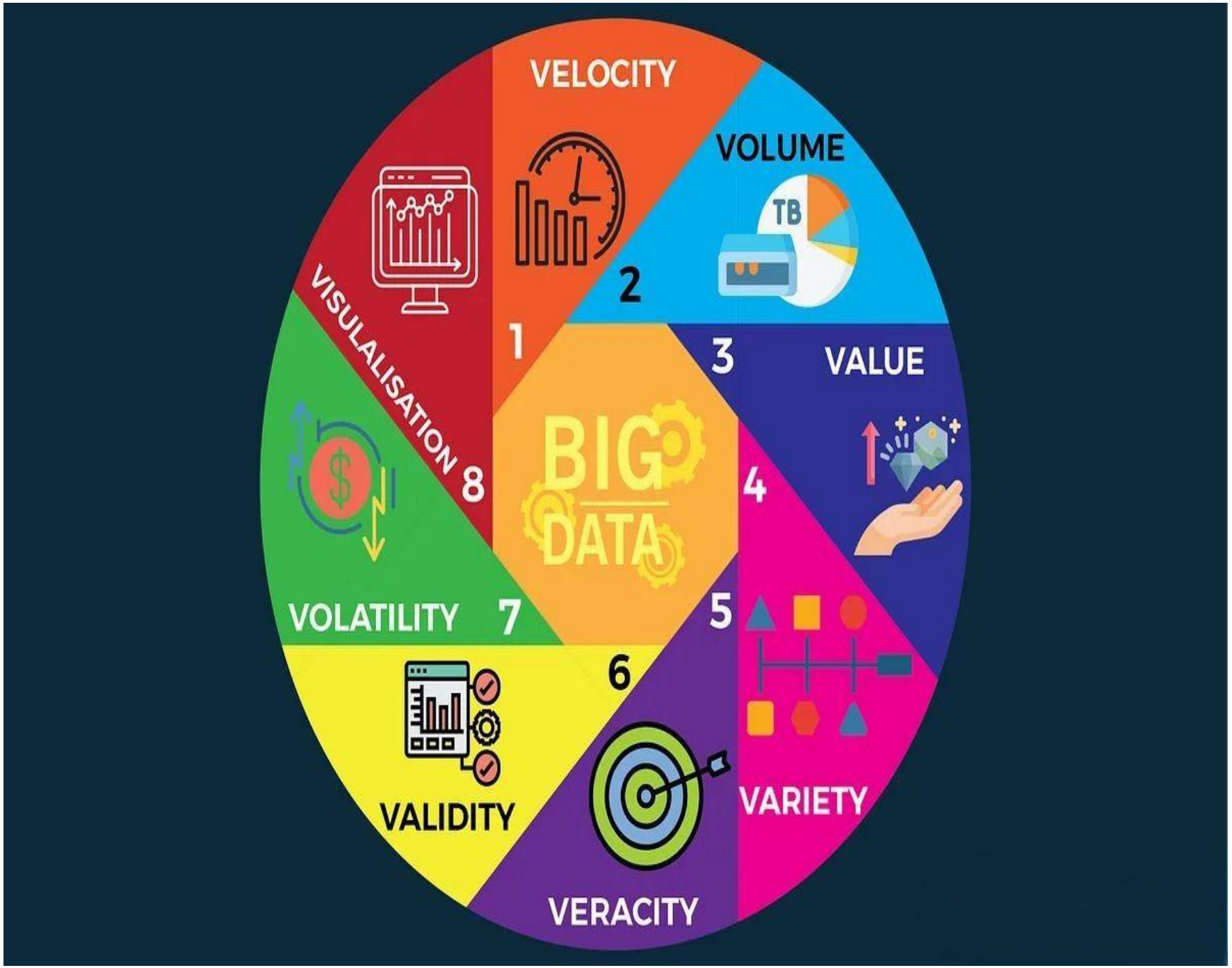
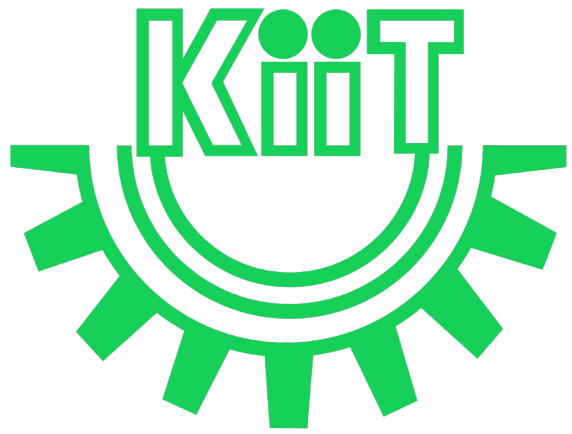


CS 3032: Big Data

Lec-9

In this Discussion . . .

- Data Streams
  - Big Data Streaming
  - Query Types on Data Streams
  - Data Sampling
  - Why is Sampling Important
  - Filter
  - Bloom Filter



Big Data Streaming

- Big Data streaming involves processing continuous streams of data in order to extract real-time insights from it.
- Big Data Streaming is important because some data requires action within seconds or milliseconds after the triggered incident. For example :
  - Delay in tsunami prediction can cost people life.
  - Delay in traffic jam prediction cost extra time.
  - Advertisement can lose popularity, if not targeted correctly.

Big Data Streaming

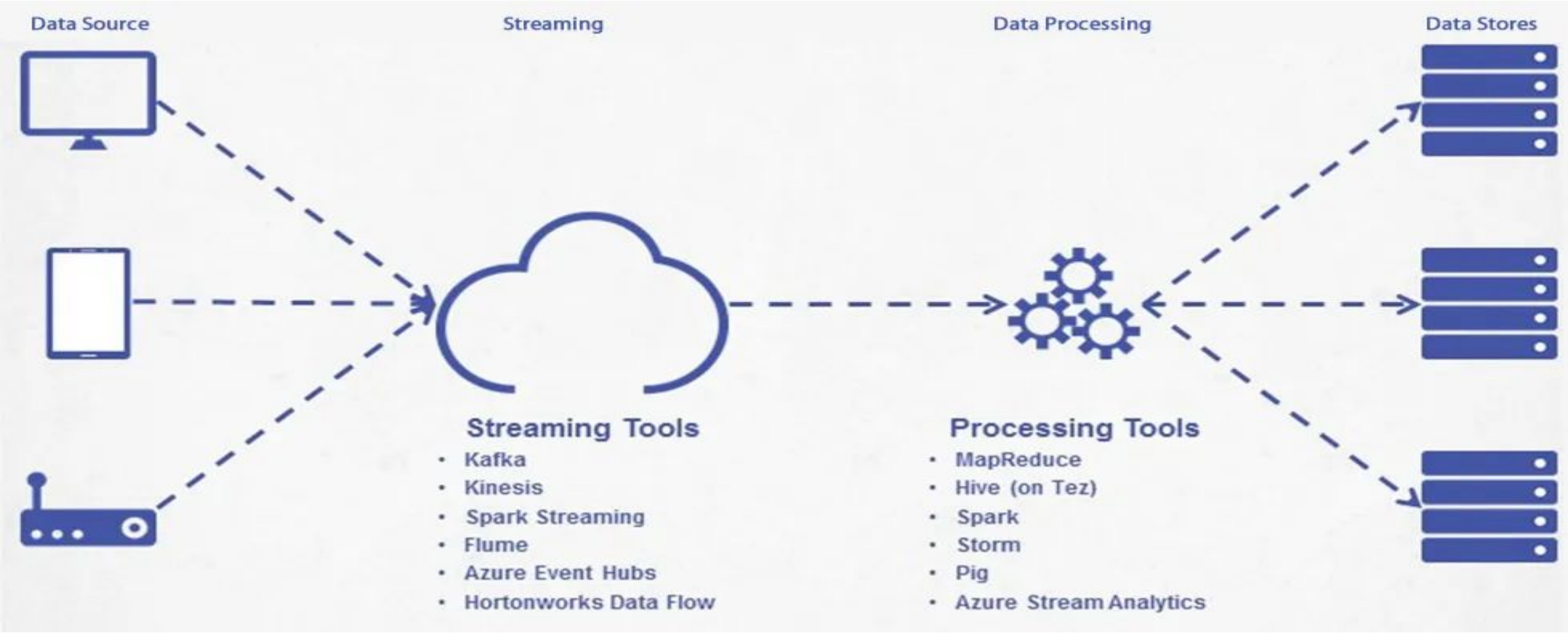
- The data on which processing is done is the data in motion. Big data streaming is ideally a speed-focused approach wherein a continuous stream of data is processed.

For real-time big data stream processing, following three key attributes are required:

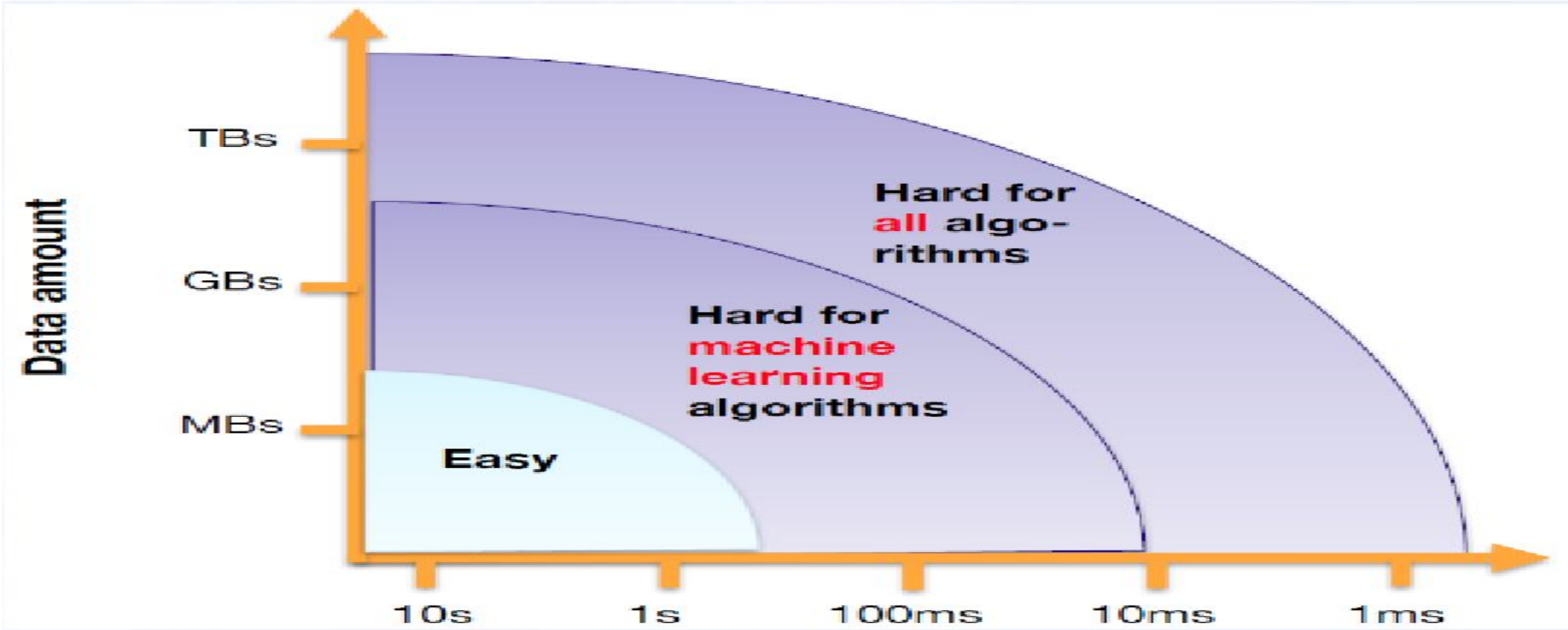
- System to collect the big data generated in real time
- System to handle the massive parallel processing of this data
- Event correlation and event processing engine for generating analytics

All the above mentioned attributes / components need to be fault tolerant, scalable and distributed, with low latency for each system.

Big Data Streaming



Why is handling Big Data Streaming hard



Big Data Streaming Process

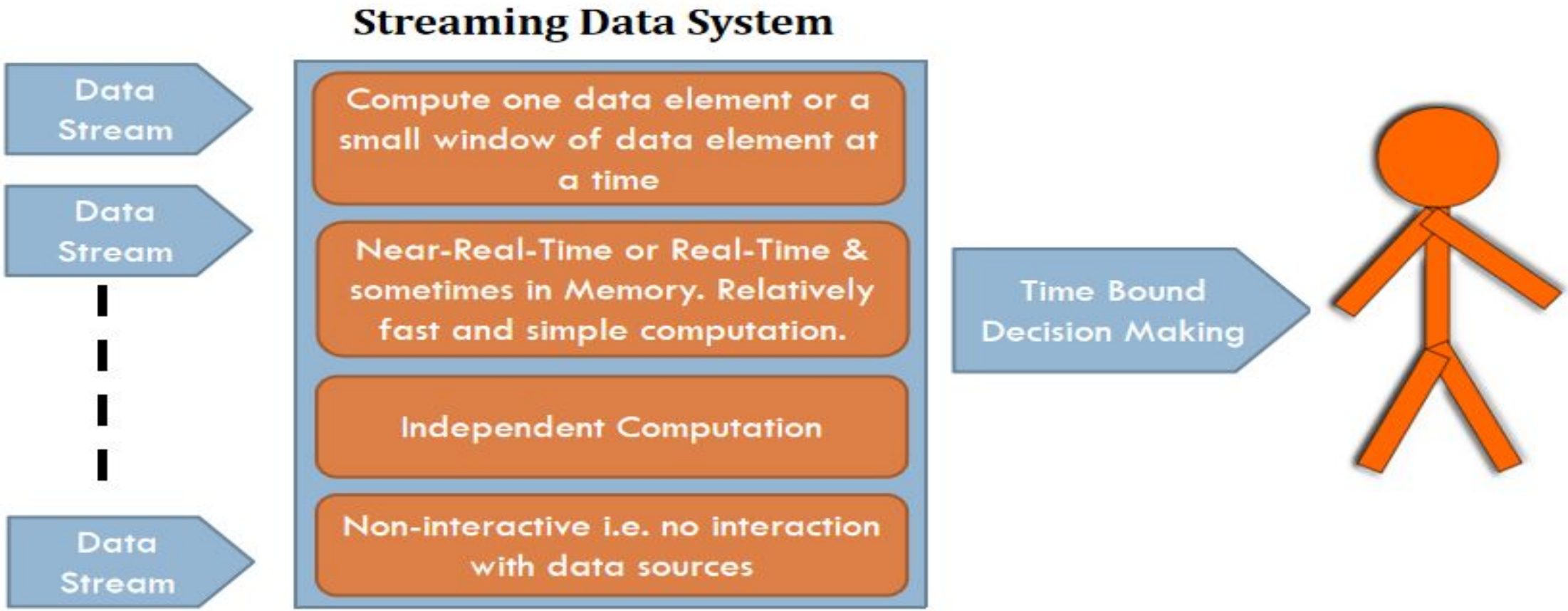
- Network device Logs
- Click Stream for Web Application
- Mobile device UX interpretation
- Stock Market data etc
- Credit Card Fraud Analytics

- Listen data from sources in real time at very large volumes
- Listeners are scalable, flexible, fault tolerant data pipes.

- Real time, fault tolarant, scalable in-stream data processing
- Complex Processing engine to extract the meaningful information from this moving stream

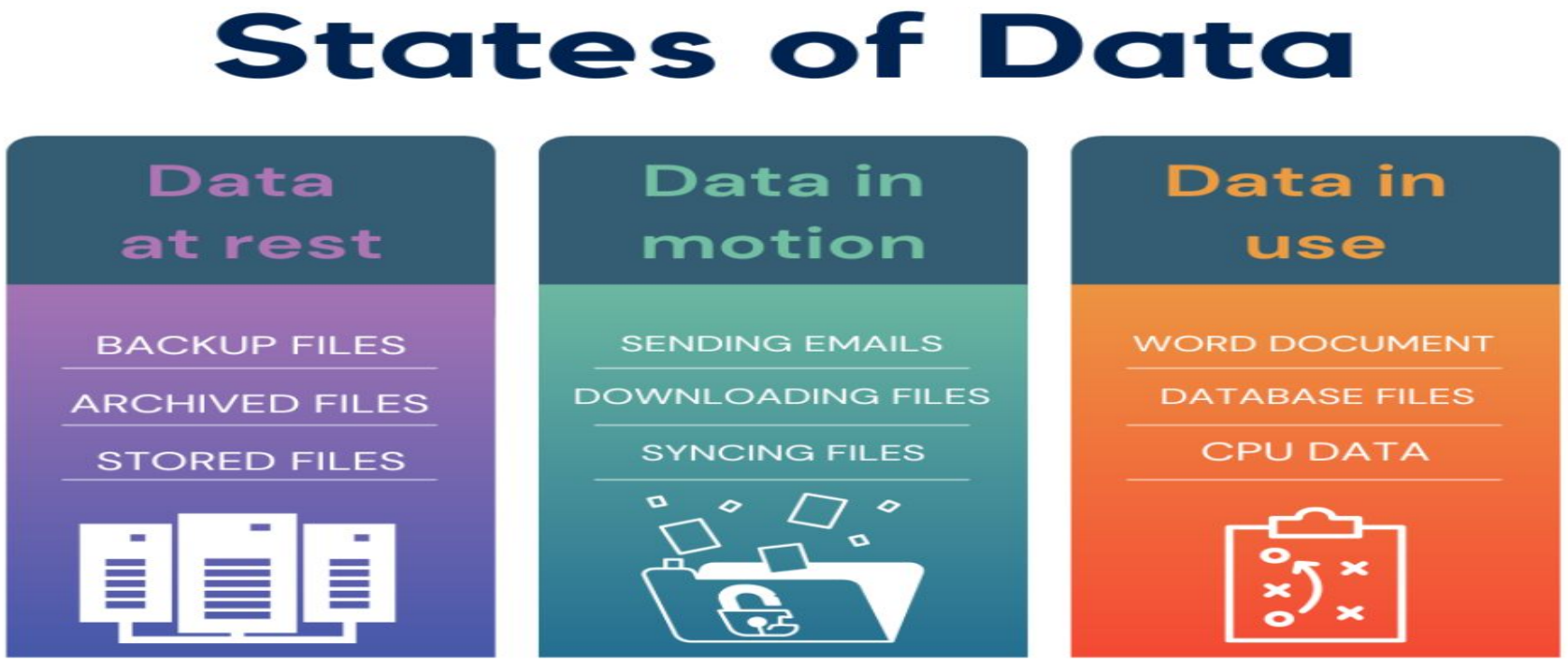
- Results are stored in destinations
- Businiess Intelligence Engine
- Application Input
- Hadoop / NoSQL for further process

Streaming Data System



Three States of Data

- **Data at rest:** Occurs when files aren't actively used, such as those stored in archives, cloud storage, or portable storage devices.
- **Data in motion:** Digital information is transmitted from one location to another.
- **Data in use:** Refers to files currently in use, either by a database, an application, or a device.



Data-at-rest Vs. Data-in-motion

Data-at-rest	Data-in-Motion
• Refers to data that has been collected from various sources and is then analyzed after the event occurs. • The point where the data is analyzed and the point where action is taken on it occur at two separate times.	• The collection process for data in motion is similar to that of data at rest. • However, the difference lies in the analytics. In this case, the analytics occur in real-time as the event happens.

Data-at-rest Vs. Data-in-motion (Contd.)

Data-at-rest	Data-in-Motion
• For example, a retailer analyzes a previous month's sales data and uses it to make strategic decisions about the present month's business activities. • The action takes place after the data-creating event has occurred.	• For example – sensor data from self-driving vehicles.
• For data at rest, a batch processing method would be most likely.	• For data in motion, you'd want to utilize a real-time processing method.

Data-at-rest Vs. Data-in-motion Infrastructure Option

Public Cloud	Bare-Metal Cloud
● Public cloud can be an ideal infrastructure choice in such scenario from a cost standpoint. ● As virtual machines can easily be spun up as needed to analyze the data and spun down when finished.	● Bare-Metal cloud can be an preferable infrastructure choice. ● It involves the use of dedicated servers that offers cloud-like features without the use of virtualization.

Streaming Data Changes over Time

- Change can be periodic or sporadic

Periodic: evening, weekends etc

People post Facebook messages more in the evening in comparison to during day, working hours.

Sporadic: major events

In summary, streaming data:

- Size is unbounded i.e. it continually generated and can't process all at once
- Size and Frequency is unpredictable due to human behavior
- Processing must be relatively fast and simple

Stream Computing

- A high-performance computer system that analyzes multiple data streams from many sources.
- The word **stream** in stream computing is used to mean pulling in streams of data, processing the data and streaming it back out as a single flow.
- Stream computing uses software algorithms that analyzes the data in real time as it streams in to increase speed and accuracy when dealing with data handling and analysis.

Stream Computing

Traditional Computing	Stream Computing
Historical fact finding	Current fact finding
Find and analyze information stored on disk	Analyze data in motion – before it is stored
Batch paradigm, pull model	Low latency paradigm, push model
Query-driven – queries data to static data	Data-driven – bring data to the analytics

Query Types on Data stream

- Types of queries one wants on answer on a data stream:
  - ***Sampling data from a stream***
    - Stream sampling is the process of collecting a representative sample of the elements of a data stream.
    - The sample is usually much smaller than the entire stream, but can be designed to retain many important characteristics of the stream, and can be used to estimate many important aggregates on the stream.

Query Types on Data stream

- Types of queries one wants on answer on a data stream:
  - ***Sampling data from a stream***
    - Unlike sampling from a stored data set, stream sampling must be performed online, when the data arrives.
    - Any element that is not stored within the sample is lost forever, and cannot be retrieved.

Query Types on Data stream

- Types of queries one wants on answer on a data stream:
  - ***Queries over sliding windows***
    - number of items of type x in the last k elements of the stream
  - ***Filtering a data stream***
    - select elements with property x from the stream
  - ***Counting distinct elements***
    - number of distinct elements in the last k elements of the stream or in the entire stream
  - ***Estimating moments***
    - estimate avg./std. dev. of last k elements

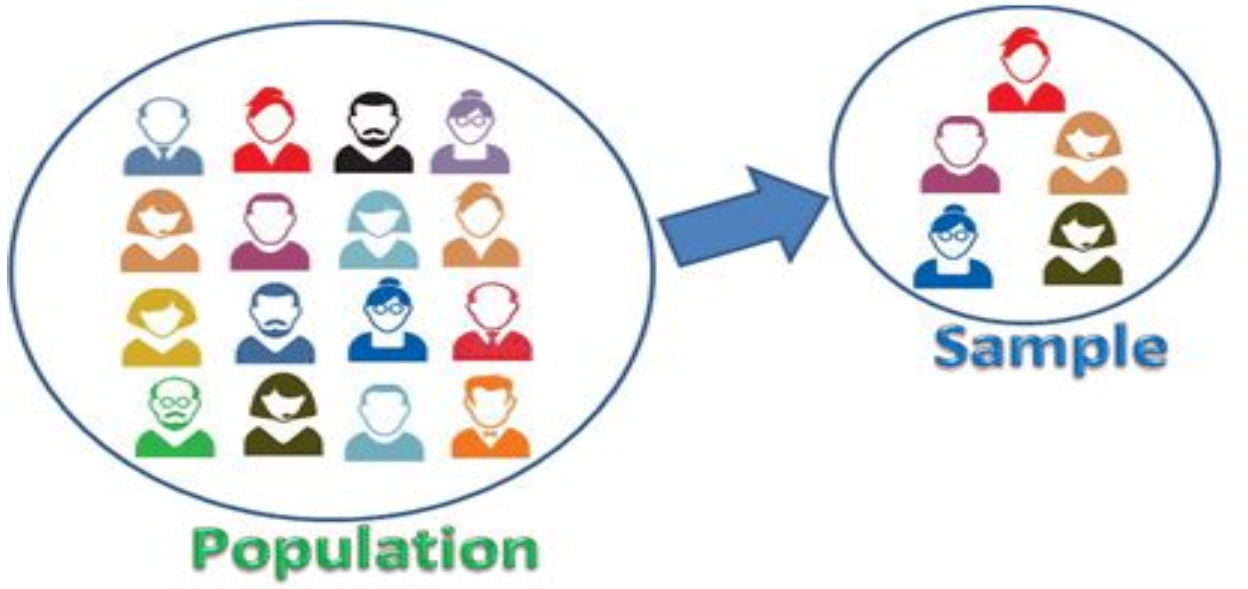
Query Types on Data stream

- Types of queries one wants on answer on a data stream:
  - ***Finding frequent elements***
    - The frequent items problem is to process a stream of items and find all those which occur more than a given fraction of the time.

Data Sampling

- Data Sampling is the selection of statistical samples from the population to estimate the characteristics of the entire population.
- It is the main technique for data collection when you want to create a statistically sound conclusion from a subset of a population of data.
- Data sampling helps to make statistical inferences about the population.

It enables data scientists, predictive modelers and other data analysts to work with a **small, manageable amount of data** to build and run analytical models more quickly, while still producing accurate findings.



Data Sampling: Samples & Population

Population	• Refers to the group of elements which has common characteristics. • It is a collection of observations we would like to make inferences about.
Sample	• A sample is the subset of a population
Sampling	• A collection of samples from the population is a sampling. • In other words, sampling units are an overlapping collection of elements from the population.

Why is Sampling Important?

- **Resource Constraints:**
  - If the target population is not small enough, or if the resources at the disposal don't give the bandwidth to cover the entire population, *it is important to identify a subset of the population to work with*, **i.e., A carefully identified group that is representative of the population. This process is called survey sampling.**

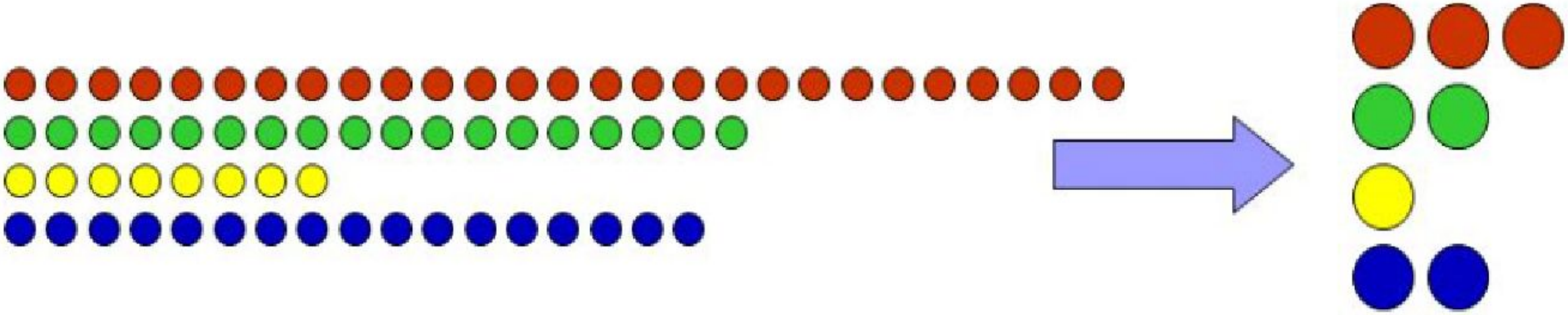
Why is Sampling Important?

- **Resource Constraints:**
  - **A carefully identified group that is representative of the population. This process is called survey sampling.**
  - Whatever the sample size, there are fixed costs associated with any survey. Once the survey has begun, the marginal costs associated with gathering more information, from more people, are proportional to the size of the sample.

Why is Sampling Important?

- **Drawing Inferences About the Population:**
  - Researcher are not interested in the sample itself, but in the understanding that they can potentially infer from the sample and then apply across the entire population.
  - Working within a given resource constraint, sampling may make it possible to study the population of a larger geographical area or to find out more about the same population by examining an area in greater depth through a smaller sample.

Sampling Data in Stream



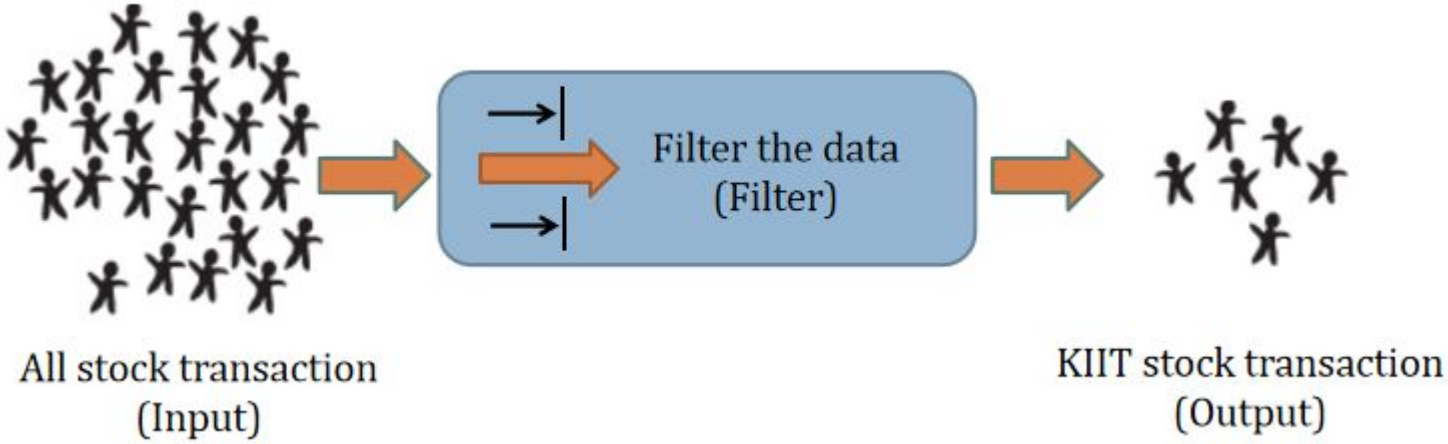
- **Need & how of sampling** - **System cannot store the entire stream conveniently,** so

  - How to make critical calculations about the stream using a limited amount of (primary or secondary) memory?
  - Don't know how long the stream is, so when and how often to sample?

Whenever a list or set is used, and space is consideration, a Bloom filter should be considered.

Filter

- A filter is a is a program or section of code that is designed to examine each input or output stream for certain qualifying criteria and then process or forward it accordingly by producing another stream.



In this example, the streams processing application needs to filter the stock transaction data for KIIT transaction records.

Filter (Contd.)

- If we want to judge whether an element is in a collection, the general idea is to save all the elements in the collection, and then determine it by comparison.
- Thus for the problem statement: **Given a set of elements, suppose we wish to know if a particular element is present in this set,** common data structures like linked list, tree, hash table (also called hash table, Hash table) and storage locations like either disk or memory are generally utilized.

Filter (Contd.)

- When faced with a huge set (millions of elements), even with an efficient search algorithm, storage is a problem. There's also latency due to disk access.
- In many cases, either time is exchanged for space, or space is exchanged for time.

Filter (Contd.)

- In the case of relatively strict response time requirements, if we have the inside, then as the number of elements in the collection increases, we need more and more storage space, and the retrieval time is getting longer and longer, resulting in too much memory overhead and time efficiency becomes lower.

- The problem that needs to be considered at this time is that **in the case of a relatively large amount of data**, **it can meet both the time requirement and the space requirement**, **so we need a data structure and algorithm that consumes less time and space**. **Bloom filter is one possible solution.**

Bloom Filter

The Bloom filter is a space efficient, probabilistic data structure, designed to test the membership of elements to a set.

- **The trade-off for being a space efficient data structure is it may return false positives, but always returns definite negatives: Meaning Bloom filters can accurately test an element for non-membership to a set, but can only with probability test an element for membership.**

Bloom Filter (Contd.)

The Bloom filter is a space efficient, probabilistic data structure, designed to test the membership of elements to a set.

- **Bloom filters find application in circumstances where testing for non-membership saves resources such as calls to a web server, checking a proxy cache. Google uses Bloom filters in the Chrome browser as a preliminary check for malicious URLs.**

Bloom Filter (Contd.)

Bloom filter is a space-efficient probabilistic data structure conceived by Burton Howard Bloom in 1970, that is used to test whether an element is a member of a set.

- **False positive matches are possible, but False Negatives are not – in other words, a query returns either "possibly in set" or "definitely not in set"**
- **False Positive = "possibly in set" or "definitely not in set"**
  False Negative = "possibly not in set" or "definitely in set"

Bloom Filter (Contd.)

Bloom filter is a space-efficient probabilistic data structure conceived by Burton Howard Bloom in 1970, that is used to test whether an element is a member of a set.

False Positive = "possibly in set" or "definitely not in set"
False Negative = "possibly not in set" or "definitely in set"

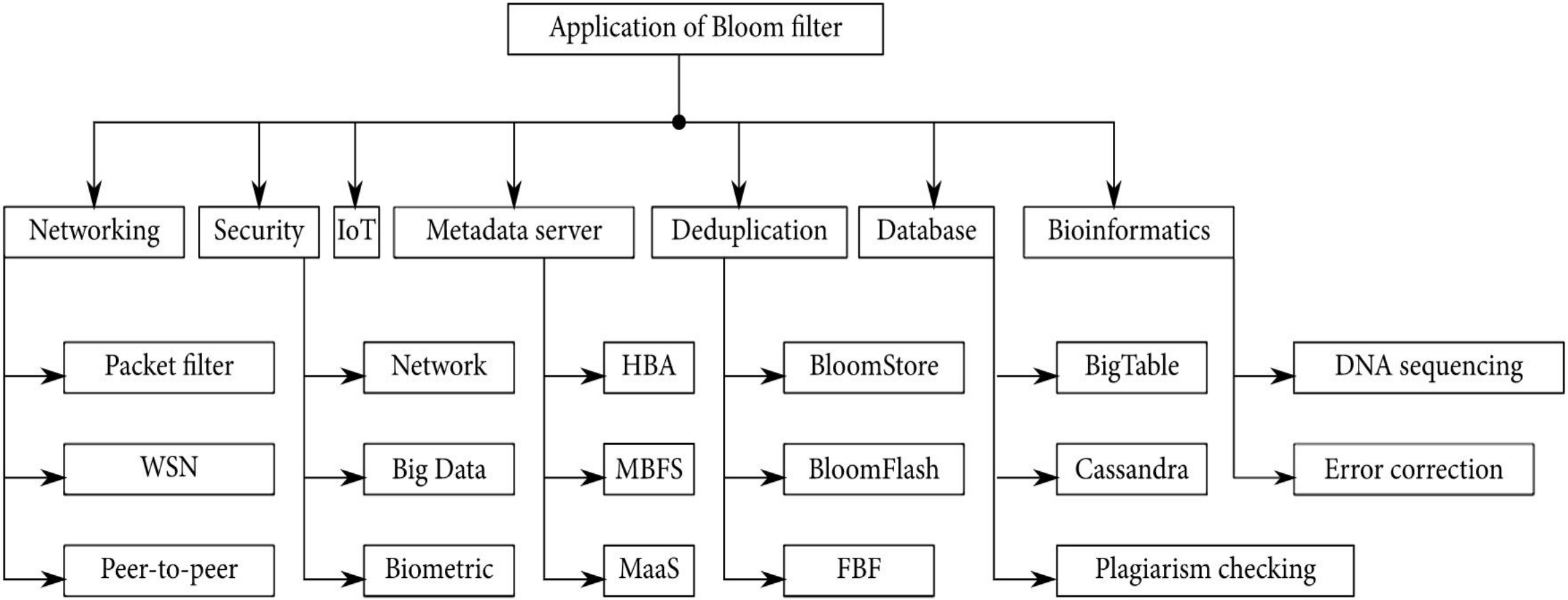
Overview
x : An element
S: A set of elements
Input: x, S **Output:**
-TRUE if x in S
-FALSE if x not in S

Bloom Filter (Contd.)

The Bloom filter is a space efficient, probabilistic data structure, designed to test the membership of elements to a set.

- Bloom filter is a data structure that stores the original set in a more compact form with support for set membership queries, that is, to query if an element is a member of the set.
- Bloom filter is a space-efficient probabilistic data structure.
- With the rise of big data since the mid-2000s, there's been increased interest in Bloom filter.

Applications of Bloom Filter



References

1. https://seeve.medium.com/big-data-streaming-phases-fed1e09a7db3
2. https://www.ibm.com/blogs/nordic-msp/digitization-and-the-climate/. Accessed Aug 18, 2023
3. Visualcapitalist.com. How Much Data is Generated Each Day? URL: https://www.visualcapitalist.com/how-much-data-is-generated-each-day/. Accessed Aug 18, 2023
4. Tilmann Rabl, Big Data Stream Processing
5. https://haisancamau.com/lap-trinh/hadoop-real-time-big-data-stream-processing-3-key-attributes-collect-process-analyze.html
6. https://www.qlik.com/us/streaming-data
7. https://twitter.com/PoweredbyINAP/status/669861450496765952/photo/1
8. https://www.techslang.com/definition/what-is-data-in-motion/
9. https://www.studocu.com/in/document/apj-abdul-kalam-technological-university/big-data-analytics/big-data-analytics-module-4/29363498