ISSN (O) 2321-2004, ISSN (P) 2321-5526

IJIREEICE

International Journal of Innovative Research in
Electrical, Electronics, Instrumentation and Control Engineering

Vol. 9, Issue 11, November 2021

DOI: 10.17148/IJIREEICE.2021.91104

A Thorough Analysis of Big Data, Fast Data and Data Lake Concepts

Sheikh Md Zubair Md Zahoor

Former Research Scholar, Computer Science, OPJS University, Churu, Rajasthan, India

Abstract: We are now seeing the emergence of two new Big Data concepts: data lakes and fast data. Are they just new marketing labels for old Big Data IT, or are they truly new? As a result, the paper's main purpose is to establish a link between these three concepts.

Keywords: Big Data, Fast Data, Data Lake

I. INTRODUCTION

The amount of data utilized by businesses for better decision-making and more efficient operations has exploded in recent decades. Almost every modern business receives a massive amount of information concerning the present state of their IT infrastructure (ITI). To find information helpful for business objectives, these data must be handled quickly and correctly. The majority of this information is in an unstructured format. The amount of unstructured data in 2020 is predicted to be over 44 ZB, according to the IDC report "The Digital Universe of Opportunities: Rich Data and the Increasing Value of the Internet of Things" (IDC, 2014). Among the numerous different big data application areas, there are two where big data and real-world insight are combined: 1) Providing big data IT as services (ready functional modules) in the implementation of other IT (in particular, search technology, deep data analytics to identify hidden patterns), the primary sources of information search and retrieval of the main content (semantics) in extra-large arrays of documents without their direct reading by a human, and so on); and 2) Analytical processing of data about the ITI's state to identify anomalies in the system functioning, IS incidence, and so on.

All of this information should not be viewed as a collection of disparate data items. Maintaining the documented relationships of every file execution and change, registry alteration, network connection, and executed binary in your environment, among other things, is a necessary. Furthermore, it is a data stream with the following distinct characteristics: large or possibly infinite volume, dynamically changing, flowing in and out in a predetermined order, requiring a quick (sometimes real-time) response time, and so on. Different types of time-series data and data produced in a dynamic ITI environment, such as network traffic, telecommunications, video surveillance, Website click streams, sensor networks, and so on, are typical examples of data streams.

At this time, no standard nomenclature in the field of big data has been defined. First and foremost, we had information. Another two notions that have recently emerged are data lakes and rapid data. Are they just new marketing labels for old Big Data IT, or are they truly new? As a result, the paper's main purpose is to establish a link between these three concepts. The following is how it's laid out. In Sections 2-4, three concepts, namely big data, data lakes, and fast data, are all presented in detail. The report concludes with their interrelationship and potential research areas.

II. BIG DATA CONCEPT

The following is a possible explanation of the big data notion. That is, datasets of such size and complexity that exceed the capabilities of typical programming tools (databases, software, etc.) for data collection, storage, and processing in an acceptable amount of time, and a-fortiori exceed the human perception capability of such datasets. Data might be structured, semi-structured, or unstructured, making it difficult to manage and handle in a typical manner (Miloslavskaya, 2014). There are three "V" criteria for establishing the difference between big data IT and traditional IT: Variety – weak structured data, which is primarily understood as data structure irregularity and difficulty extracting homogeneous data from a stream and identifying some correlations; volume – very large volumes of data; velocity – very high data transfer rate; volume – very large volumes of data; velocity – very high data transfer rate; variety – weak structured data, which is primarily understood as data structure irregularity and difficulty extracting homogeneous data from a stream and identifying some correlations. Later, four more "Vs" were added to them: veracity, variability, value, and visibility.

There are three forms of big data processing (Hornbeck, 2013):

1) Batch processing in pseudo-real or soft real-time, in which data already stored in non-volatile memory is processed (only the stored data is processed), and the probability and time characteristics of the data conversion

ISSN (O) 2321-2004, ISSN (P) 2321-5526

IJIREEICE

International Journal of Innovative Research in
Electrical, Electronics, Instrumentation and Control Engineering

Vol. 9, Issue 11, November 2021

DOI: 10.17148/IJIREEICE.2021.91104

process are primarily determined by the requirements of the applied problems. Because it can use more data and, for example, better train predictive models, this model gives performance gains.

2) Hard real-time stream processing, in which collected data is processed without being saved to non-volatile media (only the processing operations results are saved), and the probability and time characteristics of the data conversion process are primarily determined by the incoming data rate, because the appearance of queues at the processing nodes results in irreversible data loss. This architecture is appropriate for applications in which a quick response time is important.
3) Hybrid processing (also known as Lambda Architecture (Marz, 2013)) uses a hybrid model with three architectural principles: robustness (the system must be able to handle hardware, software, and human errors); data immutability (raw data is stored indefinitely and never modified); and recomputation (results can always be obtained by (re)-computing the stored raw data) and implemented by a four-layer architecture: batch layer (contains the immutable, continuously growing master dataset stored on a distributed file system and computes batch views from this raw data); serving layer (contains the immutable, constantly growing master dataset stored on a distributed file system and computes batch views from this raw data); batch layer (contains the immutable, constantly growing (loads and exposes the batch views in a data store for further querying), The speed layer (which only deals with fresh data and compensates for the serving layer's high latency updates by computing real-time views) and the combination layer (which only deals with old data and compensates for the serving layer's high latency updates by computing real-time views) (for synchronization, results composition and other non-trivial issues).

Big Data IT differs from traditional IT in that it is data-centric and data-driven. Whereas in traditional IT, a processing device or medium (computer, cluster, Cloud) is placed at the heart of the data processing process to process various requests (orders, etc.), big data IT is primarily viewed as a continuously flowing substance, with processing mechanisms built into the streams themselves. Where the downstream rate for data incoming for processing and the rate at which results are sent should not be less than the stream rate, since this would result in an infinite growth of queues or wasteful storage of infinitely rising volumes of raw data.

Theoretical underpinnings of big data (Rajaraman, 2011): IT is a subset of computing known as data science, which includes the following: Methodologies for distributed file systems and transforming datasets to generate processes for parallel and distributed processing of very big data sets are being developed. Similarity search, which includes key minhashing algorithms and hashing that is sensitive to locality; For fast arriving data that must be handled instantly, data-stream processing and specific algorithms are used. Large-scale datasets, rating search results, link-spam detection, and the hubs-and-authorities approach are all examples of search engine technology. Associative rules, market baskets, the a-priori method, and its enhancements are all examples of frequent-itemset data mining. Clustering algorithms for very large, high-dimensional datasets; Problems with web applications include ad management and recommendation systems. Algorithms for mining and studying the structure of very large graphs (such as social networks); Singular-value decomposition and latent semantic indexing are two dimensionality reduction techniques for discovering significant properties of a huge dataset. Perceptrons, support-vector machines, and gradient descent are examples of machine-learning techniques that can be applied to very large-scale data.

Let's look at a few key properties of big data: Be precise: data must be accurate and obtained from a reputable (trustworthy) source. Be timely: data must be current and represent the most up-to-date ITI status, and historic data should be provided as needed; Be thorough: data must be gathered and entered into a model that depicts the entire situation, is adaptable, integrated, and simple to distill into valuable data; Data should be adapted to a specific business goal. Be relevant: data must be relevant to and current for the organization that will use it.

Big data processing is generally targeted at data mining, which is the process of extracting or «mining» (discovering) knowledge from massive amounts of data. Databases and data warehouses, statistics, machine learning, high-performance computing, pattern recognition, neural networks, data visualization, information retrieval, image and signal processing, and spatial or temporal data analysis are all part of data mining.

III. DATA LAKE CONCEPT

A new concept of "data lakes" or "data hubs" emerged a few years ago (in 2010). The word was coined by James Dixon (Dixon, 2010), but it is sometimes dismissed as merely a marketing moniker for a Hadoop-supporting product. Another perspective is that yesterday's unified storage is today's enterprise data lake (McClure, 2016).

A data lake is a massively scalable storage repository that stores a large amount of raw data in its natural format («as is») until it is needed, as well as processing tools (engines) that can consume data without affecting its structure (Laskowski, 2016). Data lakes are often designed to manage enormous volumes of unstructured data that arrive quickly (as opposed to the highly structured data found in data warehouses), from which further insights can be gained. As a

ISSN (O) 2321-2004, ISSN (P) 2321-5526

IJIREEICE

International Journal of Innovative Research in
Electrical, Electronics, Instrumentation and Control Engineering

Vol. 9, Issue 11, November 2021

DOI: 10.17148/IJIREEICE.2021.91104

result, the lakes employ dynamic (rather than pre-built static) analytical applications. As soon as the data in the lake is created, it becomes available (again in contrast to data warehouses designed for slowly changing data).

A semantic database, a conceptual model that uses the same standards and technology as are used to construct Internet hyperlinks, is frequently included in data lakes. This layer of context establishes the meaning of the data and its interrelationships with other data. SQL and NoSQL database techniques, as well as online analytics processing (OLAP) and online transaction processing (OLTP) capabilities, can be combined in data lake solutions.

Unlike a hierarchical data warehouse, which stores data in files or folders, the data lake has a flat architecture, with each data element having a unique identification and a set of expanded metadata tags. The data lake does not necessitate a rigorous structure or the handling of data of all forms and sizes, but it does necessitate keeping data arrival order. It can be thought of as a large data pool that collects all historical data and new data (structured, unstructured, and semi-structured data, as well as binary from sensors, devices, and other sources) in near real time into a single location, with the schema and data requirements not defined until the data is queried («schema-on-read»).

The data lake can be partitioned into three tiers if necessary: one for raw data, another for supplemented daily data sets, and a third for third-party data. Another option is to divide the data lake into three parts based on its lifespan: data that is less than 6 months old, older but still active data, and archived data that is no longer used but must be kept (this stale data can be moved to slower, less expensive media).

As a result, the data lake serves as a cost-effective location for basic data analysis, while flexible and task-oriented data structuring is used just where and when it is required (Stein, 2014). The analyzed data is the data lake output, which is an important part of the extended analytical ecosystem.

The data lake should be integrated with the rest of the enterprise's information technology infrastructure. This necessitates basic data classification and indexing, as well as data security. Support for data in data lakes should have a few key characteristics:

1) A high-availability, scale-out architecture that increases with the data;
2) Governing and enforcing regulations for data retention, destruction, and identification;
3) A centralized cataloguing and indexing of all available data (and metadata), including sources, versioning, veracity, and accuracy;
4) Data cardinality refers to the way data is related to other data.
5) Data transformation lineage (tracking) refers to what was done with it, when and where it came from (assessment of internal, external, and acquired third-party data sources), who and why altered it, what versions are available, and so on;
6) A single, easy-to-manage, fully shareable data storage available to all applications (rather than constructing separate silos for new files, mobile workflows, cloud workflows, and data copies);
7) A shared-access architecture that allows each bit of data to be accessed in numerous formats at the same time, eliminating the need to extract, transform, and load data and allowing data-in-place analytics, expedited workflow assistance between different applications, and so on;
8) Support for mobile workforce from any device (tablet, Smartphone, laptop, or desktop);
9) Agile data lake analytics using several analytical methodologies and data pipelines, as well as single-subject analytics based on extremely particular use cases;
10) Some level of service quality, with consolidated workflows securely isolated in their own zones within the system for security or performance;
11) Efficiency, such as erasure coding, compression, and deduplication;
12) Data is never moved since processing is applied to the data, not the other way around, and so on.

Logs and sensor data (e.g., from the Internet of Things), low-level customer behavior (e.g., Website click streams), social media, document collections (e.g., e-mail and customer files), geo-location trails, images, video, and audio, and other data useful for integrated analysis are among the data that go into a lake. Application framework for capturing and contextualizing data through cataloguing and indexing, as well as advanced metadata management, are all part of the data lake governance. It aids in the collaborative creation of models (views) of this data, as well as gaining additional visibility and managing incremental metadata upgrades. And advanced metadata management combines working with quickly changing data structures with query response times of less than a second on highly organized data. As a single raw-data storage, the data lake's operational availability, integrity, access control, authentication and authorization, monitoring and audit, business continuity, and disaster recovery are all critical.

IV. CONCEPT OF FAST DATA

Enterprise data is expanding at an unsustainable rate in today's dynamic world. Because the stream of data from sensors, actuators, and machine-to-machine communication in the Internet of Things and modern networks is so large, it's become critical for businesses to figure out what data is time-sensitive and needs to be acted on right away versus

ISSN (O) 2321-2004, ISSN (P) 2321-5526

IJIREEICE

International Journal of Innovative Research in
Electrical, Electronics, Instrumentation and Control Engineering

Vol. 9, Issue 11, November 2021

DOI: 10.17148/IJIREEICE.2021.91104

what data can sit in a database or data lake until it's needed (Shalom, 2014). Fast data refers to the use of big data analytics on smaller data sets in real-time or near-real-time to solve a specific problem. They're crucial in applications that require low latency and rely on high input/output capacity for quick updates. Fast data analytics aims to swiftly collect and process structured and unstructured data in order to take action. Fast data is frequently ingested in data systems in streams, and there is a greater emphasis on processing large data streams at high speeds. New flash drives are poised to exceed the current speed limit, which is mostly imposed by the performance of hard drive devices. The use of in-memory databases and a data grid on top of flash devices will allow stream processing capacity to be increased. As a result, fast data necessitates the use of two technologies: a streaming system capable of providing events as they occur, and a data store capable of processing each item as it occurs. Fast data processing is defined as the ability to "ingest" (receive millions of events per second), "decide" (make a data-driven choice on each event), and "analyze in real time" (to enable automated decision-making and provide visibility into operational trends of the events).

Some fast data applications require real-time streams, while others require batch data. Smart surveillance cameras, for example, that can continuously record events and use predictive analytics to identify and flag security anomalies as they occur, or smart grid applications that can analyze real-time electric power usage at tens of thousands of locations and automatically initiate load shedding to balance supply and demand in specific geographical areas are examples of potential use cases for fast data.

As a result, we may conclude that fast data is a complement to big data when it comes to managing vast amounts of "in-flight" data. Interacting with fast data is very different from interacting with massive data at rest, and it necessitates various approaches.

V. CONCLUSION

Let's go over the main points that the notions mentioned use. Big data is defined by volume, velocity, variety, truthfulness, variability, value, and visibility and can be structured, semi-structured, or unstructured. Batch processing in pseudo-real or soft real-time, stream processing in hard real-time, and hybrid processing are the three types of big data processing. A data lake stores a large amount of raw data in its natural format (structured, unstructured, and semi-structured) that is categorized according to reusability criteria until it is needed, as well as processing systems (engines) that can consume data without damaging its structure. It can be thought of as a big data pool that collects all past data and adds new data in near real time, with the schema and data requirements not being established until the data is queried. The data lakes are well-managed and safeguarded, with scale-out architectures that provide high availability, centralized cataloging and indexing, a shared-access approach that allows users to access data from any approved modern device, and improved data lineage (tracking). Fast data refers to time-sensitive structured and unstructured "in-flight" data that should be collected and acted upon as soon as possible (requires low latency and processing of big data streams at speed). It refers to the use of big data analytics on smaller data sets in real-time or near-real-time to solve a specific problem. Fast data necessitates a streaming system that can provide events as quickly as they arrive, as well as a data store that can process each item as quickly as it arrives. Some fast data applications require real-time streams, while others require batch data.

Consequently, we can deduce that not all huge data is quick, and not all rapid data is big. Thus, these two notions share a point of intersection. The conclusion reached after analyzing big data and data lakes is that the second notion evolutionary continues the first on a greater turn of the spiral. Figure 1 depicts the final picture of the three concepts' interrelationship. A suggested future study topic is a detailed comparison of architectures that enable these ideas.

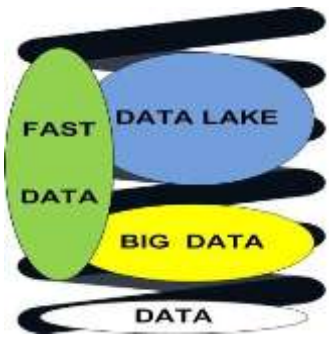

Figure 1: Intrinsic relationship between the concepts of big data, fast data, and data lake.

ISSN (O) 2321-2004, ISSN (P) 2321-5526

IJIREEICE

International Journal of Innovative Research in
Electrical, Electronics, Instrumentation and Control Engineering

Vol. 9, Issue 11, November 2021

DOI: 10.17148/IJIREEICE.2021.91104

REFERENCES

[1]. Dixon, J. (2015). Pentaho, Hadoop, and Data Lakes. URL: https://jamesdixon.wordpress.com/ 2010/10/14/pentaho-hadoop-and-data-lakes/ (access date 28/05/2016).
[2]. Shalom, N. (2014). The next big thing in big data: fast data. URL: http://venturebeat.com/2014/06/25/the-next-big-disruption-in-big-data/ (access date 28/05/2016).
[3]. Hornbeck, R.L. (2013). Batch Versus Streaming: Differentiating Between Tactical and Strategic Big Data Analytics. URL: http://datatactics.blogspot.ru/2013/02/batch-versus-streaming- differentiating.html (access date 28/05/2016).
[4]. Laskowski, N. (2016). Data lake governance: A big data do or die. URL: http://searchcio. techtarget.com/feature/Data-lake-governance-A-big-data-do-or-die (access date 28/05/2016).
[5]. Marz, N., Warren, J. (2013). Big Data: Principles and best practices of scalable real-time data systems. Manning Publication Co.
[6]. McClure, T. (2016). Yesterday's unified storage is today's enterprise data lake. URL: http://searchstorage.techtarget.com/opinion/Yesterdays-unified-storage-is-todays-enterprise-data-lake (access date 28/05/2016).
[7]. Miloslavskaya, N., Senatorov, M., Tolstoy, A, Zapechnikov, S. (2014). Information Security Maintenance Issues for Big Security-Related Data. Proceedings of 2014 International Conference on Future Internet of Things and Cloud FiCloud 2014. Barcelona (Spain). Pp. 361-366.
[8]. Rajaraman, A., Leskovec, J., Ullman, J.D. (2011). “Mining of Massive Datasets”. Cambridge University Press. 326 p.
[9]. Stein, B., Morrison, A. (2014). The enterprise data lake: Better integration and deeper analytics. PricewaterhouseCooper. URL: http://www.pwc.com/us/en/technology-forecast/2014/cloud- computing/assets/pdf/pwc-technology-forecast-data-lakes.pdf (access date 28/05/2016).
[10]. The IDC study (2014). The Digital Universe of Opportunities: Rich Data and the Increasing Value of the Internet of Things. URL: http://www.emc.com/leadership/digital-universe/2014iview/index.htm (access date 28/05/2016).