Procedia Computer Science

Volume 88, 2016, Pages 300–305

7th Annual International Conference on Biologically Inspired Cognitive Architectures, BICA 2016

Big Data, Fast Data and Data Lake Concepts

Natalia Miloslavskaya and Alexander Tolstoy
National Research Nuclear University MEPhI (Moscow Engineering Physics Institute)
{NGMiloslavskaya, AITolstoj}@mephi.ru

Abstract
Today we witness the appearance of two additional to Big Data concepts: data lakes and fast data. Are they simply the new marketing labels for the old Big Data IT or really new ones? Thus the key goal of the paper is to identify the relationship between these three concepts.

Keywords: big data, fast data, data lake

1 Introduction

In the last decades the enterprises' data used for better decision-making and more efficient operations is growing tremendously. Almost all the modern enterprises get a huge amount of data about the current state of their IT infrastructure (ITI). These data need to be processed promptly and correctly to identify information useful for business needs. The majority of this data is unstructured. According to the IDC study "The Digital Universe of Opportunities: Rich Data and the Increasing Value of the Internet of Things", the amount of unstructured data in 2020 is expected to be around 44 ZB (IDC, 2014). Among many other big data application areas there are two, where amalgamation of big data plus real-world insight are working together: 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 the extra-large arrays of documents without their direct reading by a human, etc.; and 2) Analytical processing of data about the ITI's state to identify anomalies in the system functioning, IS incidents and intrusion prevention, etc.

All these data should not be considered as a combination of separate data elements. It is a must to maintain the recorded relationships of every file execution and modification, registry modification, network connection, executed binary in your environment, etc. Moreover, it is a data stream with the following unique features: huge or possibly infinite volume, dynamically changing, flowing in and out in a fixed order, demanding fast (often real-time) response time, etc. Typical examples of data streams include various kinds of time-series data and data produced in dynamic ITI environment such as network traffic, telecommunications, video surveillance, Website click streams, sensor networks, etc.

The standard terminology in the field of big data has not yet been developed at present. First of all we had data. Now we witness the appearance of another two concepts: data lakes and fast data. Are

 Selection and peer-review under responsibility of the Scientific Programme Committee of BICA 2016

doi:10.1016/j.procs.2016.07.439

they simply the new marketing labels for the old Big Data IT or really new ones? Thus the key goal of the paper is to identify the relationship between these three concepts. It is organized as follows. Three concepts, namely big data, data lakes and fast data, are consistently described in Sections 2-4 correspondently. Their interrelation and future research area conclude the paper.

2 Big Data Concept

The following interpretation of the big data concept can be offered. That is the datasets of such size and structure that exceed the capabilities of traditional programming tools (databases, software, etc.) for data collection, storage and processing in a reasonable time and a-fortiori exceed the capacity of their perception by a human. Data can be structured, semi-structured and unstructured that makes it impossible to manage and process them effectively in a traditional way (Miloslavskaya, 2014). The criteria for determining the difference between big data IT and traditional IT are three "V": volume – very large volumes of data; velocity – very high data transfer rate; variety – weak structured data, which is primarily understand as data structure irregularity and difficulty of extracting homogeneous data from a stream and identifying some correlations. Later four additional "V" – veracity, variability, value and visibility – were added to them.

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

1) *Batch processing in pseudo real or soft real-time*, where data already stored in the non-volatile memory are processed (only the stored data are processed) and probability and time characteristics of data conversion process are mainly determined by the requirements of the applied problems. This model provides performance benefits since it can use more data and, for example, perform better training of predictive models;
2) *Stream processing in hard real-time*, where collected data without storing to non-volatile media are processed (only the processing operations results are stored) and probability and time characteristics of data conversion process are mainly determined by incoming data rate, since the appearance of the queues at the processing nodes leads to irreversible loss of data. This model is suitable for domains where a low response time is critical;
3) *Hybrid processing* using hybrid model (also known as Lambda Architecture (Marz, 2013)) with three architectural principles: robustness (the system has to be able to manage hardware, software and human errors); data immutability (raw data is stored forever and it is never modified) and recomputation (results always can be obtained by (re)-computing the stored raw data) and implemented by a four-layer architecture: batch layer (contains the immutable, constantly growing master dataset stored on a distributed file system and computes the batch views from this raw data); serving layer (loads and exposes the batch views in a data store for further querying), speed layer (deals only with new data and compensates for the high latency updates of the serving layer and computes the real-time views) and combination layer (for synchronization, results composition and other non-trivial issues).

Big Data IT fundamentally differ from traditional IT so that they become data-centric or data-driven. If for traditional IT a processing device or medium (computer, cluster, Cloud), which processes various requests (orders, etc.), is put at the center of the data processing process, the big data IT is considered primarily as continuous flowing substance, processing mechanisms for which must be built in the streams themselves. Wherein a downstream rate for data incoming for processing and a rate of results delivery should be no lower that the stream rate, as otherwise this would lead to an infinite growth or queues or useless storage of infinitely increasing volumes of raw data.

Theoretical basis for big data IT is a section of computing, known as the data science, including the following (Rajaraman, 2011): Development of methodology for distributed file systems and converting datasets to create procedures for parallel and distributed processing of very large data

amounts; Similarity search, including key minhashing techniques and locality-sensitive hashing; Data-stream processing and specialized algorithms for fast arriving data that must be processed immediately; Search engine technology for large-scale datasets and ranking search results, link-spam detection, and the hubs-and-authorities approach; Frequent-itemset data mining, including associative rules, market-baskets, the a-priori algorithm and its improvements; Very large, high-dimensional datasets clustering algorithms; Web applications problems: managing advertising and recommendation systems; Algorithms for analyzing and mining the structure of very large graphs (like social-networks); Techniques for obtaining the important properties of a large dataset by dimensionality reduction, including singular-value decomposition and latent semantic indexing; Machine-learning algorithms that can be applied to very large-scale data, such as perceptrons, support-vector machines, and gradient descent.

Let us formulate some important characteristics of big data: *Be accurate*: data needs to be correct and get from a reliable (trusted) source; *Be timely*: data must be current and reflect up-to-date ITI's status, and, if necessary, the historic data should be added in due course; *Be comprehensive*: data needs to be collected into a model that paints a full picture, is flexible integrated and easily distilled into useful information; *Be tailored*: data should be tailored towards a specific business purpose; *Be relevant*: data must be applicable to and actual for the organization using it.

In general, big data processing is aimed at data mining refers to extracting or «mining» (discover) knowledge from large amounts of data. Data mining integrates various techniques from multiple disciplines such as 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.

3 Data Lake Concept

A few years ago (in 2010) a new concept of «data lakes» or «data hubs» has been appears. The term itself was introduced by James Dixon (Dixon, 2010), but sometimes it is disparaged as being simply a marketing label for a product that supports Hadoop. Or we know also another vision: yesterday's unified storage is today's enterprise data lake (McClure, 2016).

A data lake refers to a massively scalable storage repository that holds a vast amount of raw data in its native format («as is») until it is needed plus processing systems (engine) that can ingest data without compromising the data structure (Laskowski, 2016). The data lakes are typically built to handle large and quickly arriving volumes of unstructured data (in contrast to data warehouses' highly structured data) from which further insights are derived. Thus the lakes use dynamic (not pre-build static like in data warehouses) analytical applications. The data in the lake becomes accessible as soon as it is created (again in contrast to data warehouses designed for slowly changing data).

The data lakes often include a semantic database, a conceptual model that leverages the same standards and technologies used to create Internet hyperlinks, and add a layer of context over the data that defines the meaning of the data and its interrelationships with other data. The data lake strategies can combine SQL and NoSQL database approaches and online analytics processing (OLAP) and online transaction processing (OLTP) capabilities.

In contrast to a hierarchical data warehouse with files or folders data storage, the data lake uses a flat architecture, where each data element has a unique identifier and a set of extended metadata tags. The data lake does not require a rigid schema or manipulation of the data of all shapes and sizes, but it requires maintaining the order of the data arrival. It can be imagined as a large data pool to bring in all of the historical data accumulated and new data (structured, unstructured and semi-structured plus binary from sensors, devices and so on) in near real time into one single place, in which the schema and data requirements are not defined until the data is queried («schema-on-read» is used).

If required the data lake can be divided into three separate tiers: one for raw data, a second for augmented daily data sets and another for third-party information. Another possible approach is to split the data lake into three partitions according to their lifetime: data that is less than 6 months old; older but still active data and archived data no longer used but needs to be retained (this stale data can be moved to slower, less expensive media).

Hence, the data lake serves as a cost-effective place to conduct preliminary analysis of data, while flexible and task-oriented data structuring is implemented only where and for what it is necessary (Stein, 2014). The data lake outflow is the analyzed data and it forms a key component of the extended analytical ecosystem.

The data lake should be integrated with the rest of the enterprise's ITI. This requires the initial cataloguing and indexing of the data as well as data security. A few very import characteristics should be support for data in the data lakes:

1) A scale-out architecture with high availability that grows with the data;
2) Governance and enforcing policies for retention, disposition, identification of data to be tired;
3) A centralized cataloging and indexing of the inventory of data (and metadata) that is available, including sources, versioning, veracity and accuracy;
4) Data cardinality means how it relates to other data;
5) Data transformation lineage (tracking) means what was done with it, when and where it came from (the evaluation of internal, external, and acquired third party's data sources), who and why changed it, what versions are exist, how long it will be useful or relevant, etc.;
6) A single easy to manage and fully shareable data store being accessible to all the applications (instead of creating silos for new file, mobile, cloud workflows, and copies of data);
7) A shared-access model so that each bit of data would be simultaneously accessible in multiple formats to eliminate the extract, transform and load process and allow data-in-place analytics, accelerated workflow support between disparate applications, etc.;
8) Access from any device (a tablet, smartphone, laptop, desktop) to support mobile workforce;
9) Agile analytics into and from the data lake using multiple analytical approaches and data workflows as well as single subject analytics based on very specific use cases;
10) Some level of quality of service with securely isolate consolidated workflows in their own zones within the system for safeguarding or performance;
11) Efficiency including erasure coding, compression, deduplication;
12) You never move the data as the processing goes to the data, not the other way round, etc.

The data going into a lake contain logs and sensor data (e.g., from the Internet of Things), low-level customer behavior (e.g., Website click streams), social media, document collections of (e.g., e-mail and customer files), geo-location trails, images, video and audio and another data useful for integrated analysis. The data lake governance includes application framework to capture and contextualize data by cataloging and indexing and further advanced metadata management. It helps to collaboratively create models (views) of this data and then gain more visibility and manage incremental improvements to the metadata. And the advanced metadata management combines working with rapidly changing data structures, as well as sub-second query response on highly structured data. And for the data lake itself as it is a single raw-data store ensuring its operational availability, integrity, access control, authentication and authorization, monitoring and audit, business continuity and disaster recovery is of great importance.

4 Fast Data Concept

In current dynamic world the enterprises' data is growing too fast. As the stream of data from sensors, actuators and machine-to-machine communication in the Internet of Things and modern

networks is very large, it has become vital for enterprises to identify what data is time-sensitive and should be acted upon right away and, vice versa, what data can sit in a database or data lake until there is a reason to mine it (Shalom, 2014). Fast data corresponds to the application of big data analytics to smaller data sets in near-real or real-time in order to solve a particular problem. They play an important role in applications that require low latency and depend upon the high input/output capability for rapid updates. The goal of fast data analytics is to quickly gather and mine structured and unstructured data so that action can be taken. Fast data often comes into data systems in streams and it is more emphasized on processing big data streams at speed, and new flash drives are ready for breaking the current speed limit which is bounded mostly by the performance of hard drive devices. The combination of in-memory databases and data grid on top of flash devices will allow an increase in the capacity of stream processing. Thus, fast data requires two technologies: a streaming system capable of delivering events as fast as they come in and a data store capable of processing each item as fast as it arrives. On their basis fast data processing can be described as «ingest» (get millions of events per second), «decide» (make a data-driven decision 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 rely on rapid batch data while others require real-time streams. Potential use cases for fast data include, for example, smart surveillance cameras 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 with demand in specific geographical areas.

Thus, we can conclude that fast data is a complimentary approach to big data for managing large quantities of «in-flight» data. Interacting with fast data radically differs from interacting with big data at rest and requires systems that are architected differently.

5 Conclusion

Let us repeat the main ideas which are used by the concepts discussed. Big data can be structured, semi-structured and unstructured and is characterized by volume, velocity, variety, veracity, variability, value and visibility. There are three types of big data processing: batch in pseudo real or soft real-time, stream in hard real-time and hybrid. A data lake holds a vast amount of raw data in its native format (structured, unstructured and semi-structured) regarded according to the requirements of reusability until it is needed plus processing systems (engine) that can ingest data without compromising the data structure. It can be imagined as a large data pool to bring in all of the historical data accumulated and new data in near real time into one single place, in which the schema and data requirements are not defined until the data is queried. The data lakes are well-managed and protected, have scale-out architectures with high availability, centralized cataloging and indexing, shared-access model from any permitted modern device, use agile analytics and advanced data lineage (tracking). Fast data is time-sensitive structured and unstructured “in-flight” data and should be gathered and acted upon right away (requires low latency and processing of big data streams at speed). It corresponds to the application of big data analytics to smaller data sets in near-real or real-time in order to solve a particular problem. Fast data requires a streaming system capable of delivering events as fast as they come in and a data store capable of processing each item as fast as it arrives. Some fast data applications rely on rapid batch data while others require real-time streams.

Thus we can conclude that not all big data is fast as well as not all fast data is big. Consequently, these two concepts have the intersection. While analyzing the big data and the data lakes the conclusion is that the second concept evolutionary continues the first one on a higher turn of the spiral. The final picture of the three concepts interrelation in shown in Figure 1. The possible further research area is comparison in detail of architectures supporting these concepts.

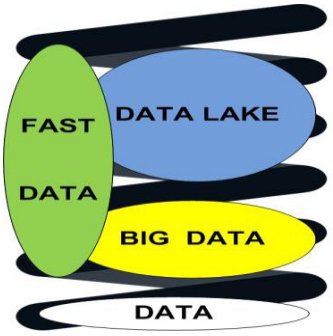

Figure 1: Interrelation between big data, fast data and data lake concepts

6 Acknowledgement

This work was supported by the Competitiveness Growth Program of the Federal Autonomous Educational Institution of Higher Education National Research Nuclear University MEPhI (Moscow Engineering Physics Institute).

References

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).

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).

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).

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).

Marz, N., Warren, J. (2013). Big Data: Principles and best practices of scalable real-time data systems. Manning Publication Co.

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).

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.

Rajaraman, A., Leskovec, J., Ullman, J.D. (2011). “Mining of Massive Datasets”. Cambridge University Press. 326 p.

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).

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).