Big Data Analytics using Real-Time Architecture

Ashwini D.Meshram
Assistant.Professor, Department of CSE
N.B Navale Sinhgad College of Engg,Solapur,Maharashtra

Anuja S.Kulkarni
Assistant.Professor, Department of CSE
N.B Navale Sinhgad College of Engg,Solapur,Maharashtra

Saanavi S.Hippargi
Assistant.Professor, Department of CSE
N.B Navale Sinhgad College of Engg,Solapur,Maharashtra

Abstract- Real-time big data analytics has become a frequent buzzword among big data discussions. The report on "Real-time big data analytics: Emerging architecture by Mark Barlow revealed that RTBDA is an essential aspect and value proposition of big data analytics, specifically the ability to make decisions in real time based on the analysis of available information. The capability to store data quickly isn't new. What's new is the capability to do something meaningful with that data, quickly and cost effectively. For some, real-time big data analytics (RTBDA) is a ticket to improved sales, higher profits and lower marketing costs. To others, it signals the dawn of a new era in which machines begin to think and respond more like humans. The goal of this paper is sketching out a practical RTBDA stack that will serve a variety of stakeholders including users, vendors, investors, and corporate executives who make or influence purchasing decisions around information technology.

Keywords- Big Data, RTBDA stack, Hadoop.

I. INTRODUCTION

'Big Data' is a term used to describe collection of data that is huge in size and yet growing exponentially with time. In short, such a data is so large and complex that none of the traditional data management tools are able to store it or process it efficiently.

Following are some the examples of 'Big Data'-

1. The New York Stock Exchange (NYSE) generates about one terabyte of new trade data per day.
2. Social media impact:-Statistic shows that 500+terabytes of new data gets 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.
3. Single Jet engine can generate 10+terabytes of data in 30 minutes of a flight time. With many thousand flights per day, generation of data reaches up to many Petabytes.

Analysis of Big data

Batch and real time data processing are the two types of analysis of big data and both have advantages and disadvantages as given in fig 1. The decision to select the best data processing system for the specific job at hand depends on the types and sources of data and processing time needed to get the job done and create the ability to take immediate action if needed.

Batch data processing is an efficient way of processing high volumes of data is where a group of transactions is collected over a period of time. Data is collected, entered, processed and then the batch results are produced (Hadoop is focused on batch data processing). Batch processing requires separate programs for input, process and output. An example is payroll and billing systems.

In contrast, real time data processing involves a continual input, process and output of data. Data must be processed in a small time period (or near real time). Radar systems, customer services and bank ATMs are examples. Real time data processing and analytics allows an organization the ability to take immediate action for those times when acting within seconds or minutes is significant. The goal is to obtain the insight required to act prudently at the right time - which increasingly means immediately.

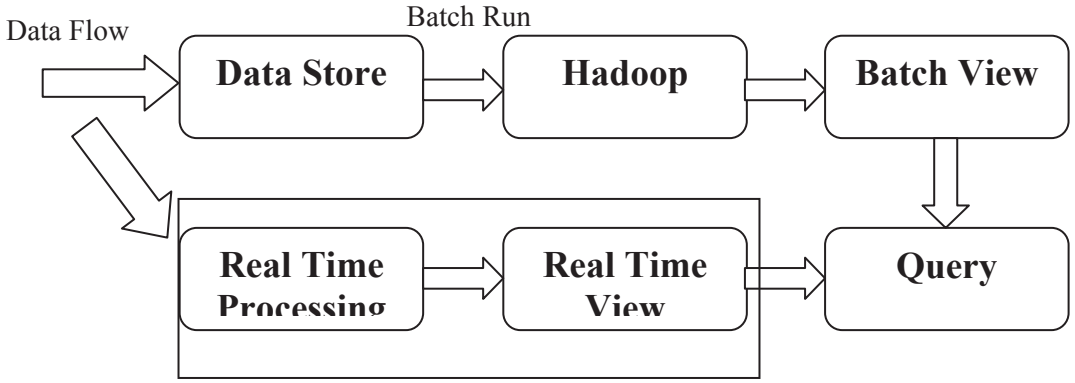


Fig 1.Batch and real time data processing

Batch Processing Disadvantages

If you are currently using batch processing, you understand that you need to run the process at a time when your systems aren't as busy with other functions, especially if you have large amounts of data to process and enter. This often means completing the processing overnight when your business is closed. However, batch processing requires some level of supervision to ensure it is running correctly, as well as an increased risk of creating downtime for your business. If a problem arises and no one is monitoring the process, your business will experience costly delays that can negatively impact the customer experience and your ability to make money. You also can't access the data until the following day after it has been processed. This can cause problems if customers are requesting information about an order they placed that day.

REAL-TIME PROCESSING ADVANTAGES

Making the switch to real-time processing can provide your business with a number of advantages. Real-time processing means the data will be available to everyone in real-time, your business will require fewer resources to sync the system, reduce the amount of paper used and improve the amount of uptime for your system. Because all the data enters into the system immediately, you will be able to monitor what is happening within your business instead of waiting until the following day to discover a problem that could have been an easy fix if it had been identified immediately. Your team will be able to see errors as they happen and take care of them right away to improve the customer experience with immediate billing and help your business run more smoothly by increasing productivity and keeping closer track of inventory.

II. DEFINING RTBDA

In a very simplified way, one could say that big data analytics is composed of two parts that distinguish it from business intelligence or data warehousing and mining: distributed, parallel processing and the ability to act in real time.

One of the challenges that big data analytics addresses is the need to process large disparate data sets that normally cannot be accommodated by a single database or server. One of the solutions to address this is the use of distributed, parallel processing where large data sets are distributed to multiple servers that each process a part of the data set in parallel. Big data analytics does not require a specific structure for the data, but can work with both structured and unstructured data. Using Hadoop with MapReduce is an example of such an approach and can be credited with being a driving force behind the current interest in big data.

There are now a multitude of complementary solutions to various aspects of distributed, parallel processing, including Cassandra, Impala, Hive, HBase, Storm and Spark, and many more. New tools are emerging every day to make it easier to program algorithms or analyze data. One of the driving principles of many of these solutions is time limitation. Solutions can be found for processing large amounts of data, but what is important in a big data perspective is that processing should be completed within a defined time frame. That time frame is now increasingly being associated with real time.

Real time supports predictive analytics. Predictive analytics enables organizations to move to a future-oriented view of what's ahead and offers organizations some of the most exciting opportunities for driving value from big data. Real-time data provides the prospect for fast, accurate, and flexible predictive analytics that quickly adapt to changing business conditions. The faster you analyze your data, the more timely the results, and the greater its predictive value.

RTBDA stack

David Smith proposes a four-layer RTBDA technology stack as shown in fig 2. Although his stack is geared for predictive analytics, it serves as a good general model. He proposes a Revolution Analytics on open source R,a programming language designed specifically for data analytics.

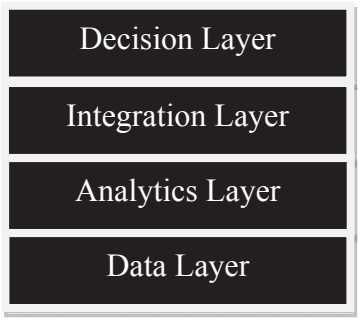
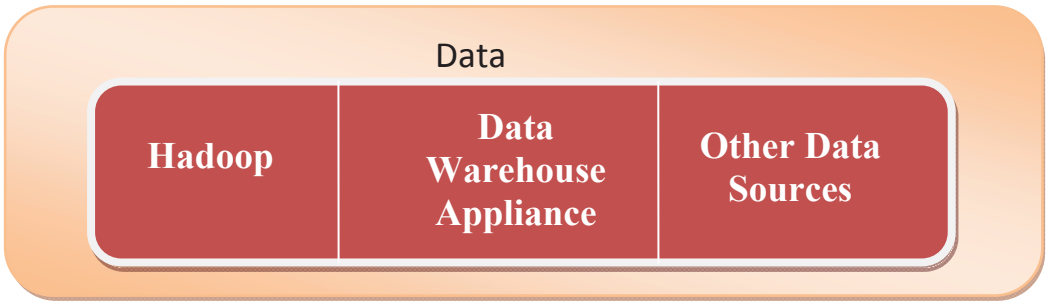


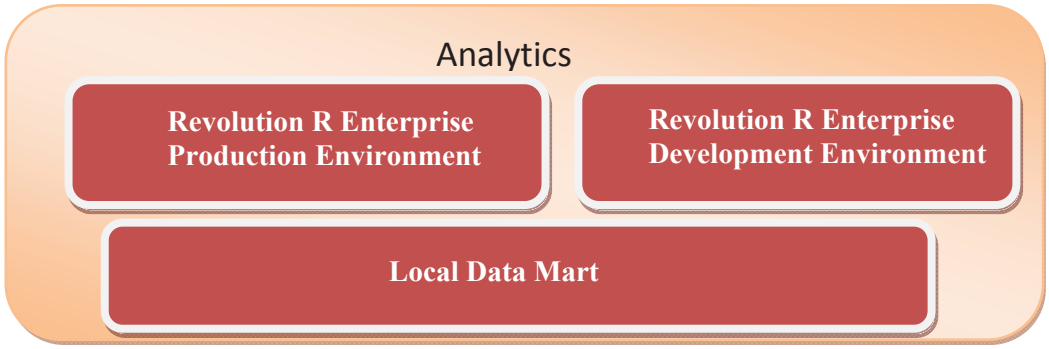
Fig 2.Real Time Big Data Predictive Analytics Stack

Data layer :- At this level one can structured data in an RDBMS, NoSQL, Hbase, or Impala; unstructured data in Hadoop MapReduce; streaming data from the web, social media, sensors and operational systems; and limited capabilities for performing descriptive analytics. Tools such as Hive, HBase, Storm and Spark also sit at this layer.



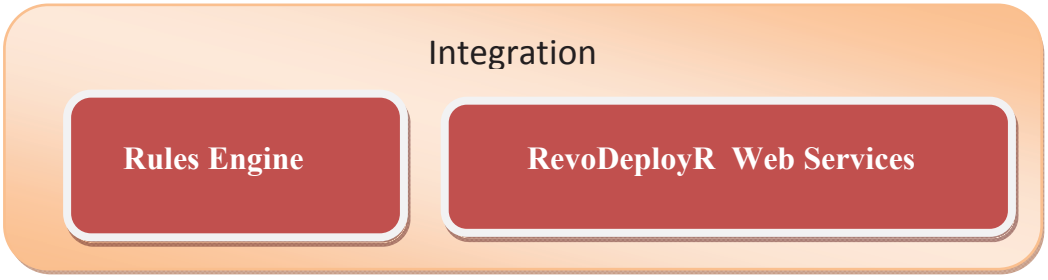
Data Layer

Analytics layer:- The analytics layer lies above the data layer. It includes a production environment for deploying real-time scoring and dynamic analytics; a development environment for building models; and a local data mart that is updated periodically from the data layer, situated near the analytics engine to improve performance.



Analytics Layer

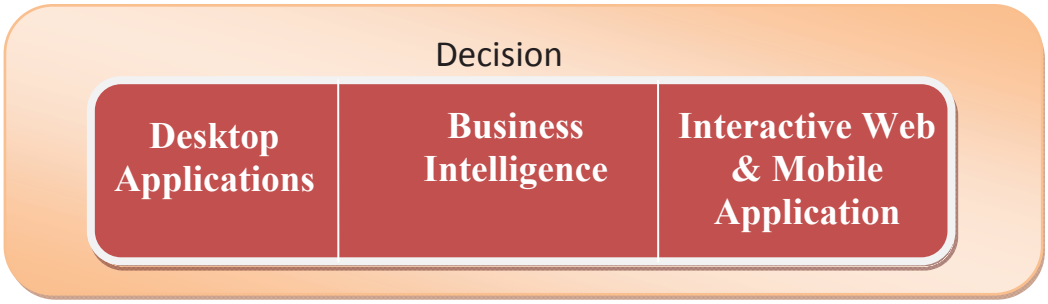
Integration layer: - Integration layer lies at the top of analytics layer. It is the "glue" that holds the end-user applications and analytics engines together, and it usually includes a rules engine or CEP engine, and an API for dynamic analytics that "brokers" communication between app developers and data scientists.



Integration Layer

Decision layer: - This is the topmost layer in RTBDA stack. , it can include end-user applications such as desktop, mobile, and interactive web apps, as well as business intelligence software. This is the layer that most people "see." It's the layer at which business analysts, c-suite executives, and customers interact with the real-time big data analytics system. It's important to note that each layer is associated with different sets of users, and that different sets of users will define "real time" differently. Moreover, the four layers aren't passive lumps of technologies.

— each layer enables a critical phase of real-time analytics deployment.



Decision Layer

III. Real Time Deployment Model

The five phases of deploying real time predictive analysis with big data to production are

1. Data distillation
2. Model development
3. Validation and deployment
4. Real-time scoring
5. Model refresh

1. *Data distillation* — Data in the data layer is not ready for predictive modeling. It lacks the structure required for building models or performing analysis. As given in fig 3,the data distillation phase includes extracting features for unstructured text, combining disparate data sources, filtering for populations of interest, selecting relevant features and outcomes for modeling, and exporting sets of distilled data to a local data mart.

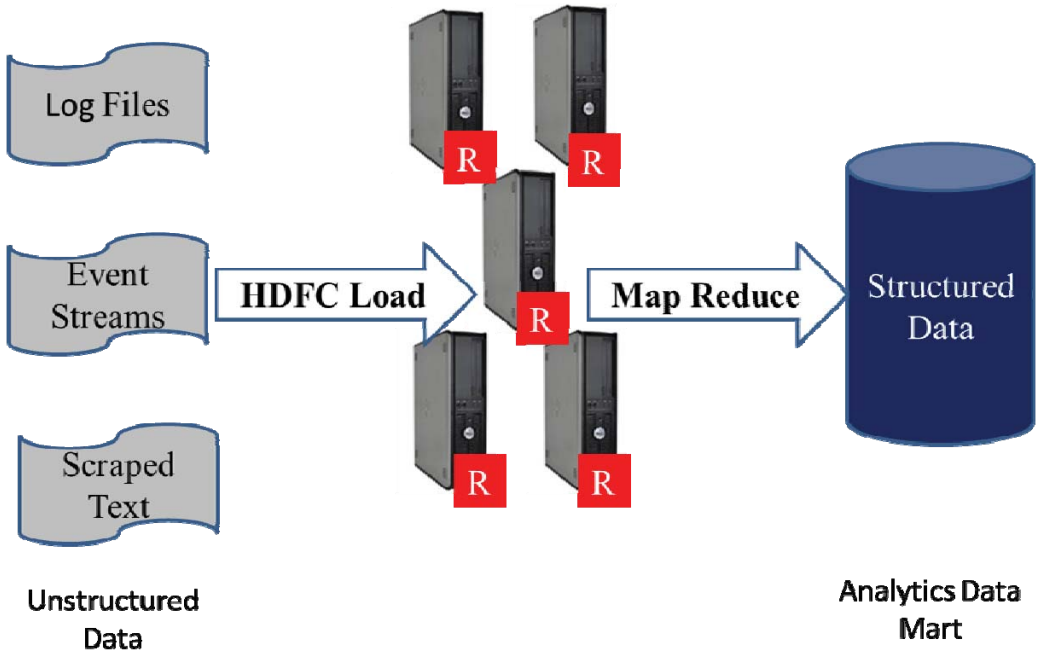


Fig 3. Example of Data Distillation in Hadoop

2. *Model development* — Processes in this phase include feature selection, sampling and aggregation; variable transformation; model estimation; model refinement; and model enchmarking as shown in fig 4.The goal at this phase is creating a predictive model that is powerful, robust, comprehensible and implementable. The key requirements for data scientists at this phase are speed, flexibility, productivity, and reproducibility. These requirements are critical in the context of big data: a data scientist will typically construct, refine and compare dozens of models in the search for a powerful and robust real-time algorithm.

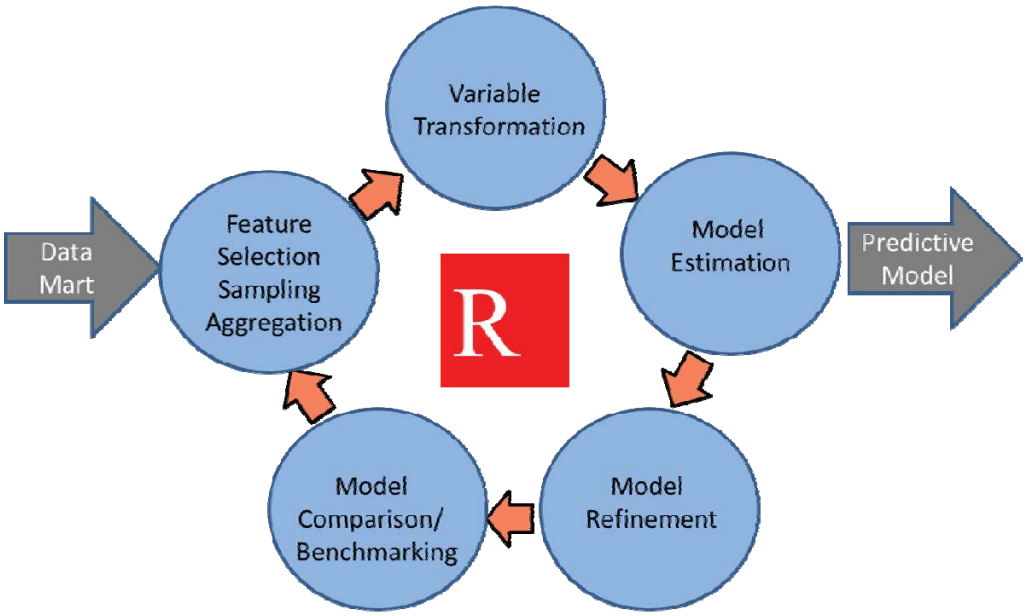


Fig 4. Model Development cycle

3. Validation and deployment — The goal at this phase is testing the model to make sure that it works in the real world. The validation process involves re-extracting fresh data, running it against the model, and comparing results with outcomes run on data that's been withheld as a validation set. If the model works, it can be deployed into a production environment.

4. Real-time scoring — In real-time systems, scoring is triggered by actions at the decision layer (by consumers at a website or by an operational system through an API), and the actual communications are brokered by the integration layer. In the scoring phase, some real-time systems will use the same hardware that's used in the data layer, but they will not use the same data. At this phase of the process, the deployed scoring rules are "divorced" from the data in the data layer or data mart. Note also that at this phase, the limitations of Hadoop become apparent. Hadoop today is not particularly well-suited for real-time scoring, although it can be used for "near real-time" applications such as populating large tables or pre-computing scores. Newer technologies such as Cloudera's Impala are designed to improve Hadoop's real-time capabilities.

5. Model refresh — Data is always changing, so there needs to be a way to refresh the data and refresh the model built on the original data. The existing scripts or programs used to run the data and build the models can be re-used to refresh the models. Simple exploratory data analysis is also recommended, along with periodic (weekly, daily, or hourly) model refreshes. The refresh process, as well as validation and deployment, can be automated using web-based services such as RevoDeployR.

IV. CONCLUSION

Today, most of our technology infrastructure is not designed for real time. We all know the Hadoop like Batch processing system had evolved and matured over past few years for excellent offline data processing platform for Big Data. Hadoop is a high-throughput system which can crunch a huge volume of data using a distributed parallel processing paradigm called MapReduce. But there are many use cases across various domains which require real-time / near real-time response on Big Data for faster decision making. Hadoop is not suitable for those use cases. Real-time systems perform analytics on short time windows, i.e. correlating and predicting events streams generated for the last few minutes. Now, for better prediction capabilities, real-time systems often leverage batch processing systems such as Hadoop.

In other words, real-time denotes the ability to process data as it arrives, rather than storing the data and retrieving it at some point in the future. Real time is a step toward building machines that respond to problems the way people do. As information technology systems become less monolithic and more distributed, real-time big data analytics will become less exotic and more commonplace.

REFERENCES

[1] "Real-Time Big Data Analytics: Emerging Architecture,"by Mike Barlow O'Reilly Media, 2013.
[2] "Big Data computing and clouds: Trends and future directions" ,Marcos D. Assunção a,, Rodrigo N. Calheiros b, Silvia Bianchi c, Marco A.S. Netto c, Rajkumar Buyyab, J. Parallel Distrib. Comput.2014.
[3] David Smith ,Revolution Analytics Consulting Services, www.revolutionanalytics.com/services
[4] G. Alex, "Interactive SQL in Apache Hadoop with Impala and Hive", available at http://www.infoq.com/news/2014/02/SQL-Apache-Hadoop-Impala-Hive, 2014