Communications on Applied Electronics (CAE) – ISSN : 2394-4714
Foundation of Computer Science FCS, New York, USA
Volume 2 – No.1, June 2015 – www.caeaccess.org

# A Reference Architecture and Road map for Enabling E-commerce on Apache Spark

Mohit Sewak
Advanced Analytics Division
BNY Mellon Corporation
Pune, India - 411 028

Sachchidanand Singh
Business Analytics Division
IBM Software Labs
Pune, India - 411 057

## ABSTRACT
Apache Spark is an execution engine that besides working as an isolated distributed, in-memory computing engine also offers close integration with Hadoop's distributed file system (HDFS). Apache Spark's underlying appeal is in providing a unified framework to create sophisticated applications involving workloads. It unifies multiple workloads, handles unstructured data very well and has easy-to-use APIs.

Apache Spark also offers a streaming component called Spark Streaming, which can write the streamed data in the same data structures, also resides in-memory and can also be read by the Spark's Spark SQL component running on top of core Spark framework.

Apache Spark has the ability to provide online machine learning, through its MLlib, and SparkR sub projects. With these, besides streaming data it can also execute machine-learning libraries, functions or algorithms.

This paper analyzes Apache Spark and highlights the role of Apache Spark (and eco-system) in the architecture of a modern E-commerce platform. This paper also aims to propose horizontally and vertically scalable reference architectures for both small and medium (SME) & large E-commerce enterprises.

## General Terms
Apache Hadoop, Big Data, Distributed Computing, Analytics & Parallel Data Processing.

## Keywords
Apache Spark, E-commerce, Spark Streaming, Spark SQL, Shark, MLlib, Mahout, SPork, SparkR, GraphX, In-Memory Computing, Distributed Architecture, Big Data, Streaming Engine, Parallel Computing.

## 1. INTRODUCTION
As per a Gartner report, Indian E-commerce market is expected to reach (USD) $6 billion in value of goods sold in 2015, a 70% growth in revenue over 2014, making India the fastest growing E-commerce market in Asia-Pacific region [1].

Forrester expects a strong compound annual growth rate (CAGR) of 9.5% between 2013 and 2018 for US E-commerce, yielding approximately (USD) $414 billion in online sales by 2018. It also expects that online sales will account for 11% of total US retail sales by 2018 [2].

Big Data and Apache Spark has a potential to command a sizeable portion in this space. Apache Spark is interesting for a lot of reasons even beyond the efficiency improvements it brings and its in-memory large-scale data processing capabilities. It has rich APIs to offer & resilient in-memory storage options, is compatible with Hadoop through Yarn, allows easier and more holistic data manipulation and analytical functions that can be developed and implemented in a more optimal way than conventional map-reduce, offers greater processing and querying speed as compared to native Hadoop ecosystem's and offers solution that are backward compatible with existing queries written in Hive [3].

Yarn came into existence with Hadoop 2.0+ & it provided a framework on which any computing engine can be deployed and hence provided a perfect platform for integration of Apache Spark with Hadoop.

In terms of commercial support also Spark is attracting good attraction. Besides Databricks[4], enterprise Hadoop providers ranging from Hortonworks[5] to MapR[4][6], Cloudera[7], Pivotal[8], and IBM[8] are offering varying modes of integration, support, and allied services offerings for Spark on Hadoop platform. Some companies are also willing to provide allied stand alone or integrated products like in-memory file systems [9] and databases [10] leveraging Apache Spark.

Apache Spark can be used to mine e-commerce graph databases and computation frameworks optimized for fast in-memory computation for big graph applications that can work well with gigabytes to even terabytes of data.

## 2. WHAT IS APACHE SPARK?
Founded by University of California, Berkeley at its Algorithms, Machines and People (AMP) Lab, and adopted by Apache as an incubator project in June 2013 [8], Spark is an open source parallel data processing framework that complements Apache Hadoop to enable and simplify the development of fast, unified Big Data applications that combines batch, streaming, and interactive analytics on almost all types of enterprise data [11].

Spark in now Apache's top level project, and has achieved the distinction to be the fastest to be promoted to a top level project from incubation stage.

Apache Spark is built upon and offers native support for Scala [13], thus providing a more succinct programming structure over java based map-reduce, and is also simpler and faster to develop as compared to traditional Map Reduce[14]. Spark's most popular usage so far has been to enable faster batch and stream processing (in that order) on big data, residing mostly in Hadoop clusters, using its own (standalone) resource manager [12].

### 2.1 Fast, Powerful Data Processing
For analysts and data scientists who rely on iterative algorithms (e.g. clustering/classification), Apache Spark is 10

to 100 times faster than MapReduce, enabling improved time to insight (time from raw data to actionable analysis) on large data, resulting in better and faster business decisions outcomes [11] [15]. Most of this capabilities are achieved owing to lower memory latencies that Apache Spark enjoys over disk latencies which the MapReduce framework is restricted to. But even on disk, Apache Spark is reported to be up to 5-10 times faster than conventional MapReduce [16].

## 2.2 Easy, Real-Time Stream Processing

A component of the Apache Spark framework called Spark Streaming extends Spark with APIs for working with (data) streams. It provides full fault tolerance for mission-critical environments. With common code across all batch and streaming applications, one can build sophisticated unified analytic applications quickly and easily using Spark Streaming [11].

# 3. APACHE SPARK IN E-COMMERCE

Every day, hundreds of millions of users and merchants interact on E-commerce sites & terabytes to petabytes of customer and transaction data is collected. The customer interactions can be expressed as complex, large scale graphs. Fast Data Mining on such data requires a distributed and parallel data processing engine that can support fast interactive queries as well as sophisticated batch and real time algorithms. Spark is the best fit for analyzing such enormous amounts (Volume) of data, especially streaming data (Velocity) with the given requirements (Veracity).

As a validation of this, Alibaba Taobao which effectively operates one of the world’s largest E-commerce platforms runs some of the largest Spark jobs in the world. A few of these Spark jobs are even made to run for weeks to perform feature extraction on petabytes of image data [17], which would have otherwise required months to run on conventional Big Data platforms and years (if not failed) on conventional platforms.

# 4. REFERENCE ARCHITECTURE CONSIDERATIONS

Today, almost 81% of online retailers use either a homegrown E-commerce solution or a licensed on-premise solution from a software vendor [18]. While such solutions have served these enterprises well over the years, E-commerce leaders report concerns that such traditional solutions now present substantial operational and opportunity risks. For some customers, their on-premise solutions is so complex that even the thought of replacing them sends shivers down the spine of IT support teams. These solutions are mission critical and tightly integrated into legacy back-end systems and corporate processes. But there is a growing concern that although these solutions may be cost-effective at present, they will be prohibitive for the business to innovate and transform at the same pace that is required to fulfill the growing and changing customer expectations.

Around 46% of E-commerce retailers report difficulties managing their platforms, keeping up with market demands and running such underlying infrastructure [19], [20].

## 4.1 Functional Requirements

While designing the reference architecture for an E-commerce platform, the following functional parameters should be considered:

### *4.1.1 Current and Forecasted Average Load Assessment*

- Size of the Catalog (No. of categories and SKU per category)
- Average browsing time per user (i.e. 05 min, 20 min or 30 min)
- Geographical distribution of users
- Daily and Hourly Average Load (orders and visits)
- Daily and Hourly Peak Load (orders and visits)

### *4.1.2 Peak and Seasonal Load Assessment*

- Quantum and Type of seasonal variations in load for different categories in the product/ service portfolio
- Type of non-seasonal, ad-hoc or promotion based transient peak load requirements

### *4.1.3 Additional Features and Requirements*

- Analytics on sales patterns and trends
- CRM and other sales campaigns and analytics
- Offline and batch analytics, visualization, reporting and MIS requirements
- Online real time analytics requirements (e.g. recommender engines)
- Multi-Currency and FX provider integrations
- Supply side vendor, payment systems and partner tie-ups and real time integration

## 4.2 Non-Functional Attributes and Requirements

While designing the reference architecture for an E-commerce platform, the following non-functional parameters should be considered:

- Regulatory requirements for data server locations and payment processing related routings
- Flexibility in Availability and performance requirements
- Security of the provider’s and customer's data
- Scalability of the platform
- Responsiveness and usability of the platform
- Cost effectiveness

For any E-commerce architecture, the non-functional attributes may play a very significant role in determining a particular architecture. But based on the budget, need and requirements, each of the non-functional attributes may be adopted in varying degrees.

Achieving a good balance of these requirements cost effectively will require an integration with some (public) cloud infrastructure, either in the core architecture or as a cloud bursting option to cater for peak load and seasonal variations (hybrid cloud). But the very moment public cloud comes in any enterprise architecture, provisioning of adequate safety measures also takes high importance.

# 5. ROADMAP, EXTENDIBILITY AND SCALABILITY OF ARCHITECTURE

Ideally a reference architecture should provide inherent scalability, readily catering to the needs of a startup to that of a large scale enterprise alike. Therefore we recommend a reference architecture that can be scaled in two dimensions:

- Strategic enhancement of system capabilities and components (extendibility)
- Dynamically scaling the cluster size of each of the required component in the system independently and flexibly (scalability)

## 5.1 Extendibility

As per the road map the system can be extended with more components to enhance functional and analytical requirements within the same architecture. Any architecture which requires the replacement/ relegation of components as the organization and its aspirations grow, might lead to wastage and additional migration efforts and resources.

The organization can start with a basic architecture, and additional components may be added/ plugged into this architecture as per the growing requirements of the organization. We therefore present two reference architecture diagrams, first depicting the state with minimal components, as adopted when the organization was a small or medium size startup/ enterprise (SME), and the next with extensions more appropriate to a large scale organization that requires the entire range of capabilities.

The following components may be required for a small or medium enterprise:

- Application and web servers for E-commerce application and flow/ logic handling
- Application servers for shopping cart logic and flow handling
- Database Servers (preferably No SQL distributed document databases) for product catalogs, features, stocks, vendor information, and product related images, videos, and documents.
- Database Servers (preferably No SQL distributed columnar store databases) for user activities, clicks, purchase and wish-list related data.
- Business Intelligence (BI) Servers for batch reporting, visualization, dash-boarding and analysis.
- Spark cluster for Spark Streaming and Spark SQL using stand-alone resource manager.
- Distributed integration bus/ messaging/ web service servers.

Additionally, as the organization's needs and aspirations grow, the following additional components may be added to the initial stage architecture.

- Another dedicated Spark Cluster for Online real time analytics, machine learning (e.g. recommender systems).
- A dedicated Hadoop (HDFS) cluster for all transaction & historical product data for offline analytical batch processing.
- A CRM suite and dedicated messaging and mail servers for managing customer and sales campaign at larger scale.

## 5.2 Scalability

The decision to add more components in the existing architecture may be a strategic decision catering to newer and enhanced strategic requirements, but the organization may literally grow every day thus requiring a similar incremental scaling of the platform itself. With more products, categories and SKU's adding to its catalog, new customers coming to the site, additional user activities and transactions being logged, increase in count of real time events, and archive sizes, the current infrastructure of the existing components/ capabilities may outgrow its initial estimate any day, even any moment without much prior notice [21].

Also the peak and seasonal load demand may be several times higher than the average load/ demand, is also highly variable, and difficult to predict in advance. Moreover, the different components in the architecture may grow at different rates, thus making optimal and cost effective scalability planning even more challenging.

The following features in the architecture ensures highly dynamic scalability to cater for highly volatile and un-synchronized demand growth and changes in the patterns across various components of the system:

- Each of the application servers has their own dedicated load balancer, automatic provisioning and scaling pieces.
- Both the product and order database servers are NO SQL distributed databases, ensuring freedom against changing schema, and ensuring high availability and scalability.
- As the product and user activity data storage requirements and growth patterns might be highly dissimilar, both have their dedicated clusters, and resource managers.
- A dedicated Spark cluster for Spark streaming and Spark SQL with its standalone resource manager, which can be independently scaled from the database or file system servers (Hadoop cluster as in Fig - 2).
- Another dedicated Spark cluster for large enterprise (shown in Fig -2) for online real time analytics with MLlib for analytics and machine learning, as the memory and load requirements might be highly different from the small and medium enterprise (shown in Fig -1) Spark cluster, which will be required for Streaming.

## 5.3 Reference Architectures for Small and Medium E-commerce Enterprise

Shown below is the proposed architecture for small & medium E-commerce enterprise:

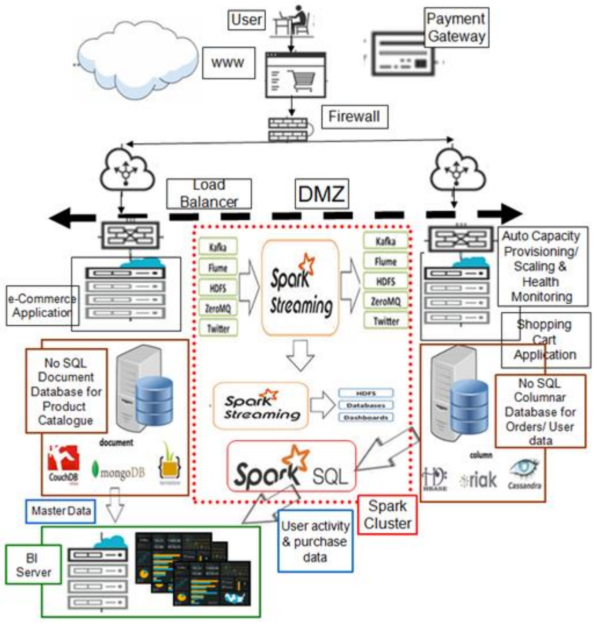


**Fig 1: Reference Architecture for Small and Medium E-commerce Enterprise**

The above architecture consists of an E-commerce application servers, Spark Cluster, shopping cart application & Business Intelligence (BI) server. User requests are routed to different application specific application servers placed in the DMZ (De-Militarized Zone) through load balancers.

The E-commerce application stores online product catalog data in No SQL document database like CouchDB, MongoDB, Terrastore etc. Customer orders, browsing data and transactions entries are stored in the No SQL columnar database like HBASE, Riak & Cassandra etc.

The Apache Spark cluster powers Spark Streaming and Spark SQL. The Spark streaming connects different applications, and Spark SQL acts as a query interface to the data stores containing catalog, user activity, and purchase data.

The architecture has provisions for Load Balancers for applications with flexibility for auto capacity provisioning & health monitoring. The Business Intelligence (BI) server is connected to reference and transaction data stores and can be utilized for both batch and real time analysis.

## 5.4 Reference Architectures for Large E-commerce Enterprise

Shown below is the proposed architecture for large E-commerce enterprise:

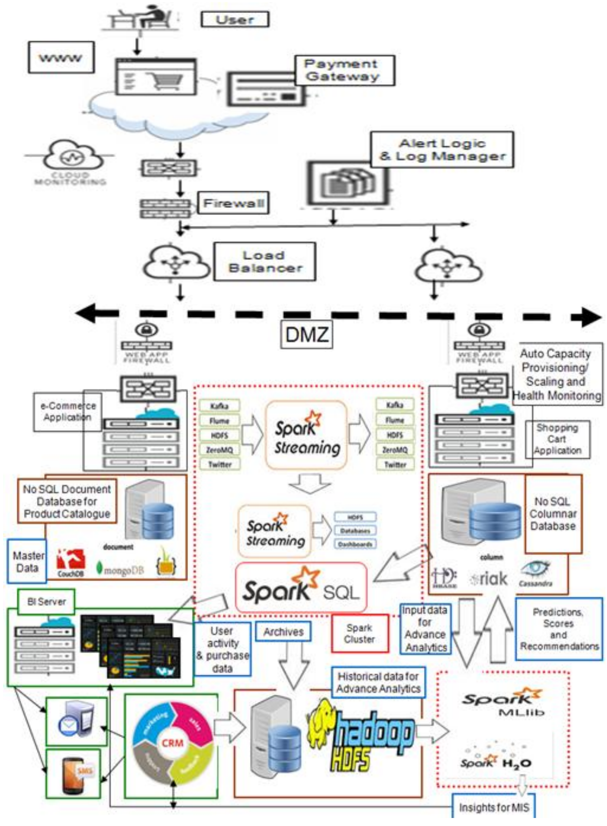


**Fig 2: Reference Architecture for Large E-commerce Enterprise**

The reference architectures for large E-commerce enterprise have additional provisions of Cloud Monitoring, Alert Logic and Log Manager, Spark MLlib/ Spark H2O/ SparkR, Hadoop Clusters, CRM Suite, Messaging servers and other components.

Here Hadoop cluster is used to store historical data archived from different data stores for analysis and analytics requiring historical data or utilizing lambda architecture (combination of archived and real-time data in a single analysis/ algorithm) and can be utilized for analysis ranging across different time periods if required.

Apache Spark's MLlib/ Spark H2O/ SparkR can be utilized for distributed/ parallel in-memory machine learning requirements. Spark's MLlib provides scalable machine learning library consisting of common learning algorithms and utilities, including classification, clustering and regression etc. The Spark H2O besides providing similar Machine Learning algorithms coupled with Spark's SQL and Scala data munging.

The Customer relationship management (CRM) component/ suite is used to organize, automate and synchronize sales, marketing, customer service, and automatic campaign management and digital marketing delivery.

## 6. COMPETITION

Apache Flink, another similar, Apache eco system project is following Apache Spark's footprints to become one of the fastest framework to mature out of incubation stage and be promoted to a top-level project. Moreover, recently Flink, has also added streaming support, which takes it close to be a real competitor to Spark [22].

Oryx, another open source project for real time and deployable, machine learning on distributed data can be a good competitor to Spark's MLlib, Spark H2O & SparkR projects. Cloudera has also recently adopted Oryx into its Hadoop distribution (CDH) [23].

Besides open source frameworks, that works mostly on commodity hardware, appliances like Teradata, and Netezza, which currently are the de-facto choice for running analytical queries on large data sets, are Spark's competitors as well in the large scale, fast data processing area owing to their existing base, available support, available skill sets, and clientele. Though cost and scalability are the deterring factors such appliance oriented solutions are likely to face while competing with Apache Spark.

HP's Vertica is a platform very similar to Apache Cassandra, and also boasts of capability to provide fast analytical queries, while also supporting SQL like query constructs, and high data compression. HP Vertica pose some challenges to Spark especially to Spark SQL platform.

HP's Haven [24] or DistributedR [25] claims to possess the capabilities of being compatible with most of the existing R codes and packages, while also utilizing parallel computation framework on its Vertica platform. So far, this product is closely knit with HPs own distributed columnar database Vertica, but can potentially have a good appeal to the customer base of SparkR

Hortonwork's Tez, IBM's BigSQL, and Cloudera's Impala [23] projects, although not exactly in-memory systems, but claims to have significantly fine-tuned the improved upon Hive's performance, reaching speeds up to 100 times faster as compared to that of HiveQL's. These platforms might also pose some challenge to selected Spark's eco-system project like Spark SQL and Shark [26].

# 7. SUMMARY AND CONCLUSION

Running on commodity hardware, supporting good integration with other open source platforms, and incorporating a large variety of sub-projects that covers most of the technological requirements of an analytically savvy enterprise, that thrives on real time, scalable and cost effective solutions, Apache Spark is surely going to be a favorite for both startups and matured enterprises alike in the field of E-commerce and others. With so many things happening around the platform, and with the level of interest it has attracted from the developers and consumers, Apache Spark's fast paced entry into the elite top level Apache project club comes as no surprise.

Apache Spark surely is disruptive but with this quantum of growth and rapid evolution, also comes an imminent ask for the companies adopting it to be equally agile in technology adoption. This also means that the architecture supporting the business platform, should be scalable and extendible without major business disruption and huge investments and wastes.

Moreover with the growing trend of startups in the E-commerce space, the architecture should be capable of supporting a functional business with minimal investments into technology, yet have the ability to be relevant with the growing requirement of the changing business landscape, with modular design enabled by easy plug ability of advanced features, tools and platforms to support challenges of a large scale enterprise.

The reference architecture proposed in this paper endeavors to answer the challenges as highlighted above. From here on, individual adopters may customize the fine details of these for their specific requirements to offer a robust, scalable and contemporary platform for a modern E-commerce platform.

# 8. REFERENCES

[1] Gartner Says India eCommerce Market To Reach $6 Billion in 2015, http://www.gartner.com/newsroom/id/2876517

[2] US eCommerce Forecast: 2013 To 2018, https://www.forrester.com/US+eCommerce+Forecast+2013+To+2018/fulltext/-/E-RES115513

[3] Finding a Spark at Yahoo! http://blogs.gartner.com/nick-heudecker/finding-a-spark-at-yahoo/

[4] MapR announces Apache Drill and Apache Spark integration, http://www.itwire.com/it-industry-news/development/65714-mapr-announces-apache-drill-and-apache-spark-integration

[5] Hortonworks Invests In Spark On Hadoop, http://www.informationweek.com/big-data/big-data-analytics/hortonworks-invests-in-spark-on-hadoop/d/d-id/1316035

[6] Let Spark Fly: Advantages and Use Cases for Spark on Hadoop, https://www.mapr.com/blog/let-spark-fly-advantages-and-use-cases-spark-hadoop-webinar-follow#.VX7eBGMpldF

[7] Cloudera Offers Apache Spark For Hadoop Big Data, http://google.com/newsstand/s/CBIwxYu3iRE

[8] Apache lights a fire under Hadoop with Spark, http://www.pcworld.com/article/2336380/apache-lights-a-fire-under-hadoop-with-spark.html#tk.rss_all

[9] Pivotal and EMC are betting on Spark cousin Tachyon as in-memoryfilesystem, http://google.com/newsstand/s/CBIwxJba3x8

[10] MemSQL extends in-memory database with Apache Spark connector, http://siliconangle.com/blog/2015/02/10/memsql-extends-in-memory-database-with-apache-spark-connector/

[11] Apache Spark, http://www.cloudera.com/content/cloudera/en/products-and-services/cdh/spark.html

[12] Survey reveals a few interesting numbers about Apache Spark, https://gigaom.com/2015/01/27/a-few-interesting-numbers-about-apache-spark/

[13] Here's why Python and Scala aren't old news in the world of data science, http://google.com/newsstand/s/CBIwqpGhgA8

[14] Apache Spark: Hadoop friend or foe?, http://siliconangle.com/blog/2015/02/05/apache-spark-hadoop-friend-or-foe/

[15] Databricks demolishes big data benchmark to prove Spark is fast on disk, too, http://google.com/newsstand/s/CBIwj7e31ho

[16] 4 reasons why Spark could jolt Hadoop into hyperdrive, http://google.com/newsstand/s/CBIw-Nyvnh8

[17] Mining Ecommerce Graph Data with Spark at Alibaba Taobao, https://databricks.com/blog/2014/08/14/mining-graph-data-with-spark-at-alibaba-taobao.html

[18] The New Retail Reality Calls for the Death of Traditional POS, http://blog.demandware.com/tag/ecommerce/page/12

[19] 4 Reference Architectures To Optimize Your Ecommerce, http://www.rackspace.com/blog/4-reference-architectures-to-optimize-your-ecommerce/

[20] Three New AWS Reference Architectures for E-Commerce, https://aws.amazon.com/blogs/aws/three-new-aws-reference-architectures-for-e-commerce/

[21] Flipkart sends apology mail to customers after its botched 'Big Billion Day Sale', http://ibnlive.in.com/news/flipkart-sends-apology-mail-to-customers-after-its-botched-big-billion-day-sale/504504-7.html

[22] Apache Flink 0.8.0 Released, Roadmap for 2015 Published, http://www.infoq.com/news/2015/01/apache-flink-0.8.0-released

[23] Cloudera is rebuilding machine learning for Hadoop with Oryx,http://google.com/newsstand/s/CBIwqaqThAk

[24] Haven Big Data Platform, http://www8.hp.com/in/en/software-solutions/big-data-platform-haven

[25] HP Distributed R, http://www.vertica.com/hp-vertica-products/hp-vertica-distributed-r/

[26] Spark fires up near-real-time big data, http://gcn.com/articles/2015/02/09/apache-spark.aspx