# Big Data Integrations

## Part 1

< 20 Pages

This Book consist of list of 8 Big Data Integrations with

- Architecture diagram
- Use case (why?)
- Descriptions
- Modules.

(Gowtham SB and Saravana Kumar P)

PosterMyWall.com

## Authors

**Author 1**
Gowtham SB
Big Data Engineer / Data Science enthusiast

Email – sbgowtham.sb@gmail.com
LinkedIn - https://www.linkedin.com/in/sbgowtham/

**Author 2**
Saravana Kumar
Big Data Engineer

LinkedIn - https://www.linkedin.com/in/saravanasaro/

## Disclaimer

1. **The content in this book is based on my personal experiments plus internet reference.**
2. **"A Question can have many answers and justifications". The answer and justification in the book is based on my personal experiments.**
3. **This book will have basic architecture and justifications in simple explanations.**

## About Book

This Book consist of list of 8 Big Data Integrations with

- Architecture diagram
- Use case (why?)
- Descriptions
- Modules.

## Audience

People who have intermediate/Advance knowledge on BIG DATA

## Table of content

## Introduction

This Book consist of list of 8 Big Data Integrations.

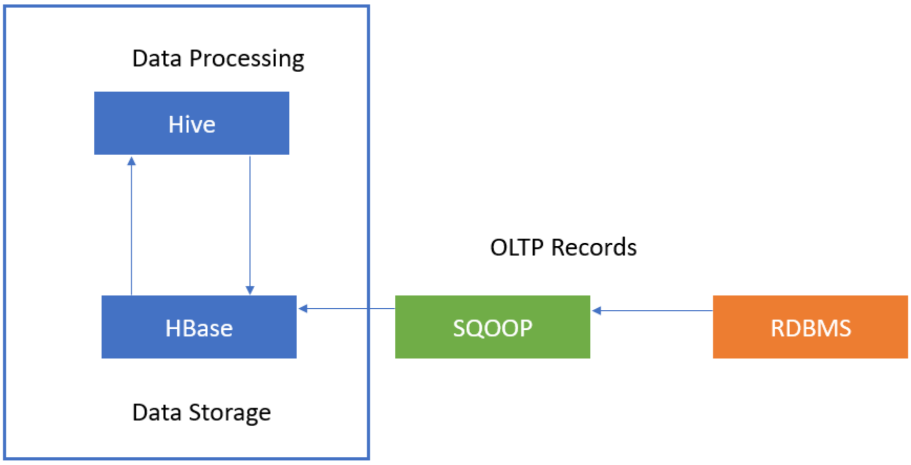
Integrations are always challenging and especially in Big Data world. In this book I have few integrations techniques with proper justifications in shorts.

This book has all the explanations and justifications in theory.

Soon 2nd Version of the book will be released.

## Hive HBase Integration

### Architecture



### What is?

- **HBase** - Apache HBase is an open source NoSQL database that provides real-time read/write access to those large datasets.
- **SQOOP** - Apache Sqoop efficiently transfers bulk data between Apache Hadoop and structured datastores such as relational databases.
- **HDFS** - The Hadoop Distributed File System (HDFS) is the primary data storage system used by Hadoop applications.
- **Hive** - Apache Hive is a data warehouse system for data summarization, analysis and querying of large data systems in open source Hadoop platform.

### Description

HBase tables from Hive that can be accessed by both Hive and HBase. This allows you to run Hive queries on HBase tables. You can also convert existing HBase tables into Hive-HBase tables and run Hive queries on those tables as well.

### Why

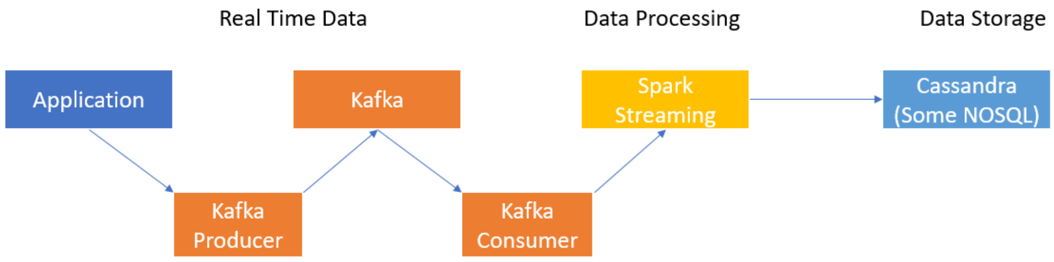
HBase allows update(versioning) where as in Hive from 0.14 version we can able to perform INSERT UPDATE and DELETE but enabling ACID settings and Create table with ORC and Bucketing since Bucket in hive is not recommended in many cases. So, we can’t use update directly in hive instead load the data in HBase and Query it in Hive (HBase has less processing queries so we go Hive for processing and HBase for storage).

### Modules

| | |
|---|---|
| Data Pipeline | - RDBMS to HBase through SQOOP. |
| Data Storage | - Store the Data in HBase. |
| Data Processing | - Process the Data with Hive Query Engine. |

## Kafka Spark Cassandra Integrations

### Architecture



### What is?

- **Spark** - Apache Spark is a unified analytics engine for large-scale data processing.
- **Kafka** - Apache Kafka provides low-latency, high-throughput, fault-tolerant publish and subscribe pipelines and can process streams of events.
- **Cassandra** - Apache Cassandra is a free and open-source distributed wide column store NoSQL database management system designed to handle large amounts of data across many commodity servers, providing high availability with no single point of failure.

### Description

Process the Realtime data for customer analytics. Retrieve the data using Kafka and process it with Spark and store in Cassandra(NOSQL).

### Why

Kafka is a leading Messaging queue system, Spark streaming is highly scalable and leading framework for Stream processing and Cassandra is leading NOSQL comes under with columnar store, AP (Availability and partition tolerance).

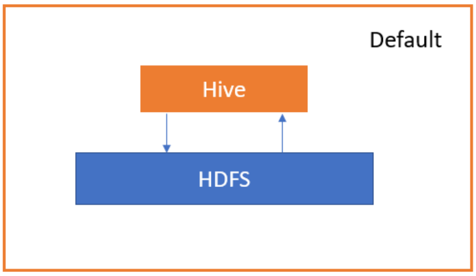
### Modules

Data Pipeline - Application to Spark through Kafka.
Data Processing - Process the Data with Spark streaming.
Data Storage - Store the Data in Cassandra.

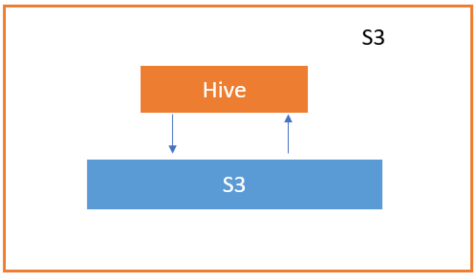
## Hive, AWS Storage Integrations

### Architecture

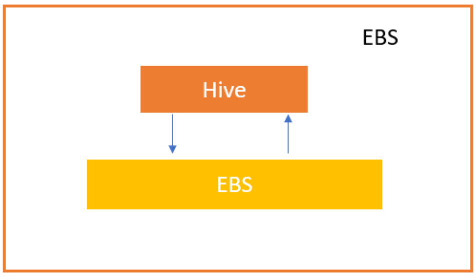
Hive with HDFS as storage



Hive with AWS (S3) storage



Hive with AWS (EBS) storage



### What is?

- **Hive** - Apache Hive is a data warehouse system for data summarization, analysis and querying of large data systems in open source Hadoop platform.
- **S3** - Amazon Simple Storage Service is storage and a Filesystem in AWS (Amazon web service).
- **EBS** - Amazon Elastic Block Store (Amazon EBS) provides persistent block storage volumes for use with Amazon EC2 instances in the AWS Cloud.
- **HDFS** - The Hadoop Distributed File System (HDFS) is the primary data storage system used by Hadoop applications.

### Description

Split the Storage and compute. Process the data in hive by separating the data from the cluster, so we decided to choose some external storage (in AWS) to place the file in S3 and EBS.

### Why

To split compute and storage. Store the data outside to Hadoop cluster so that you can have a on demand clusters like delete the cluster when it is unused.

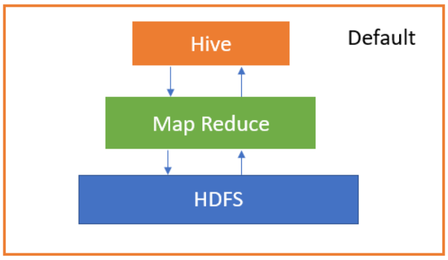
### Modules

Data Storage - Store the Data in HDFS | S3 | EBS or any other external storage system.
Data Processing - Process the Data with Hive Query Engine.

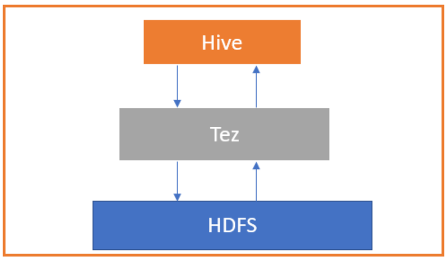
## Hive, Processing Engines Integrations

### Architecture

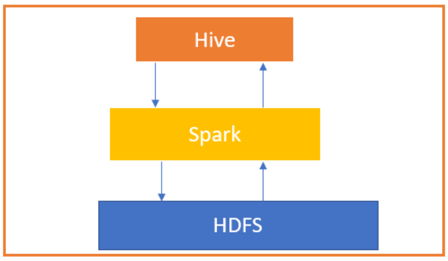
Hive with MR Engine



Hive with Tez Engine



Hive with Spark Engine



### What is?

- **Hive** - Apache Hive is a data warehouse system for data summarization, analysis and querying of large data systems in open source Hadoop platform.
- **HDFS** - The Hadoop Distributed File System (HDFS) is the primary data storage system used by Hadoop applications.
- **Spark** - Apache Spark is a unified analytics engine for large-scale data processing.
- **TEZ** - Apache Tez is an extensible framework for building high performance batch and interactive data processing applications, coordinated by YARN in Apache Hadoop.

### Description

Execute Hive with various processing engines like MR, SPARK and Tez.

### Why

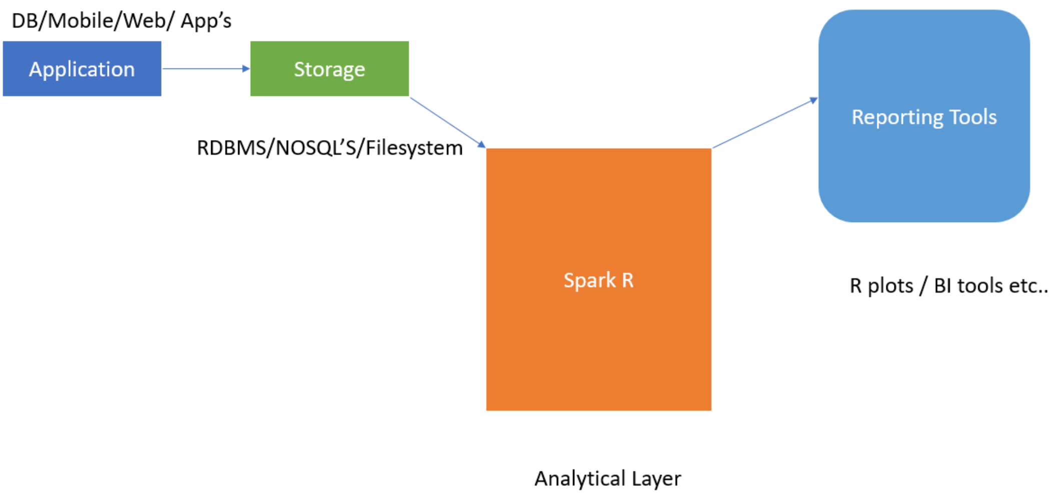
Performance optimization – Hive with native Map Reduce engine is slow compare to TEZ (improves the MapReduce paradigm by dramatically improving its speed, while maintaining MapReduce's ability to scale to petabytes of data) and Spark (In-memory data processing engine).

### Modules

Data Storage - Store the Data in Hive.
Data Processing- Process the Data with Hive and Engine as (MR | SPARK | TEZ).

## Spark R Integrations

### Architecture



### What is?

- **Spark** - Apache Spark is a unified analytics engine for large-scale data processing.
- **HDFS** - The Hadoop Distributed File System (HDFS) is the primary data storage system used by Hadoop applications.

### Description

Retrieve the data from the application and perform business analytics using Spark R.

### Why

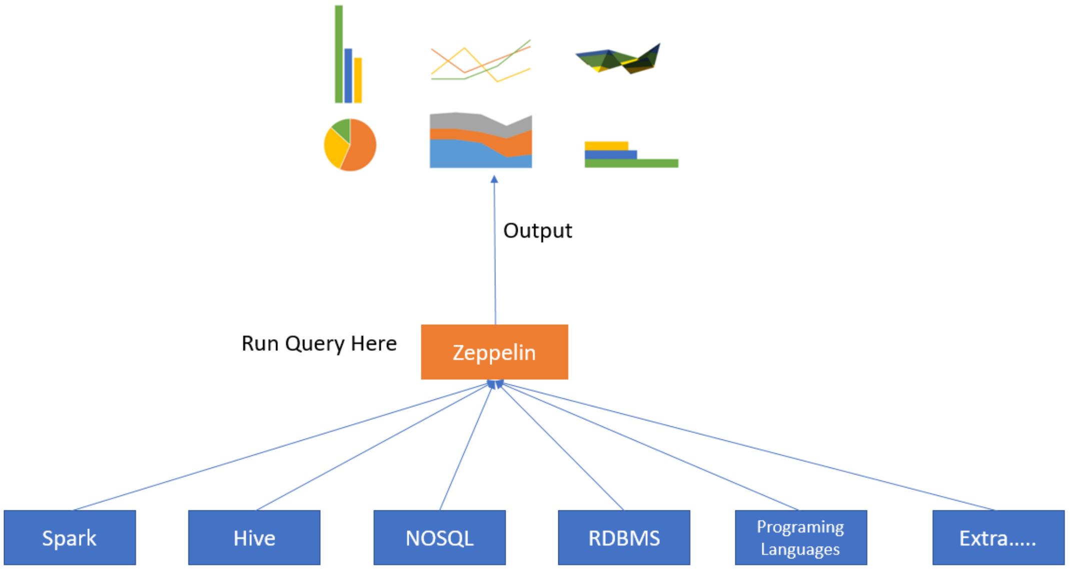
This requirement is about customer analytics so we need R programming with the best framework in Big Data since spark supports R we can use Spark R because Spark is an in-memory processing unit and visualize it by using some reporting tools / plots.

### Modules

Data Storage - Store the Data in HDFS (or any other spark supported storage system).
Data Processing - Process the Data with Spark R.

## Zeppelin Big Data Integrations

### Architecture



### What is?

- **Zeppelin** - Apache Zeppelin is a new and incubating multi-purposed web-based notebook which brings data ingestion, data exploration, visualization, sharing and collaboration features to Hadoop, Spark, etc...

### Description

Run all the interactive queries and REPL languages / Technologies in Web-based notebook to generate charts

### Why

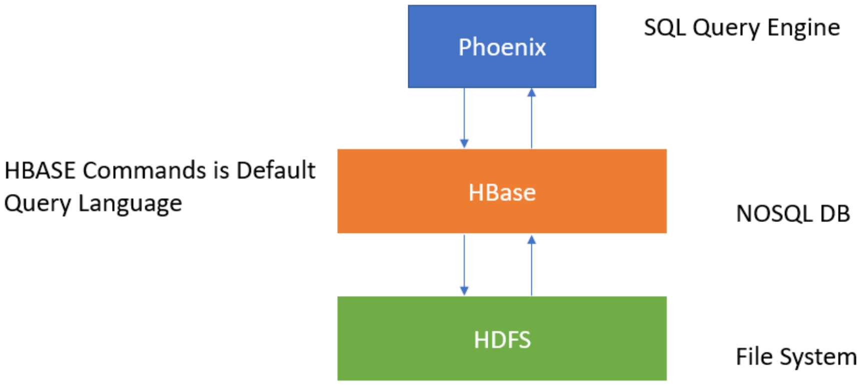
It's boring to use the Command line interface for all the REPL technologies / languages like (Hive, HBase, Cassandra, Spark, Scala, RDBMS and many other) Zeppelin supports so many technologies for integrations

### Modules

Data Visualization – Zeppelin as a web-based note book

## HBase Phoenix Integrations

### Architecture



### What is?

- **HBase** - Apache HBase is an open source NoSQL database that provides real-time read/write access to those large datasets.
- **HDFS** - The Hadoop Distributed File System (HDFS) is the primary data storage system used by Hadoop applications.
- **Phoenix** - Apache Phoenix enables OLTP and operational analytics in Hadoop for low latency applications (an SQL engine).

### Description

Integrating SQL on HBase so we use Apache **Phoenix**.

### Why

Apache Phoenix is fully integrated with other Hadoop products such as Spark, Hive, Pig, Flume, and Map Reduce. And we can use SQL on top of HBase.

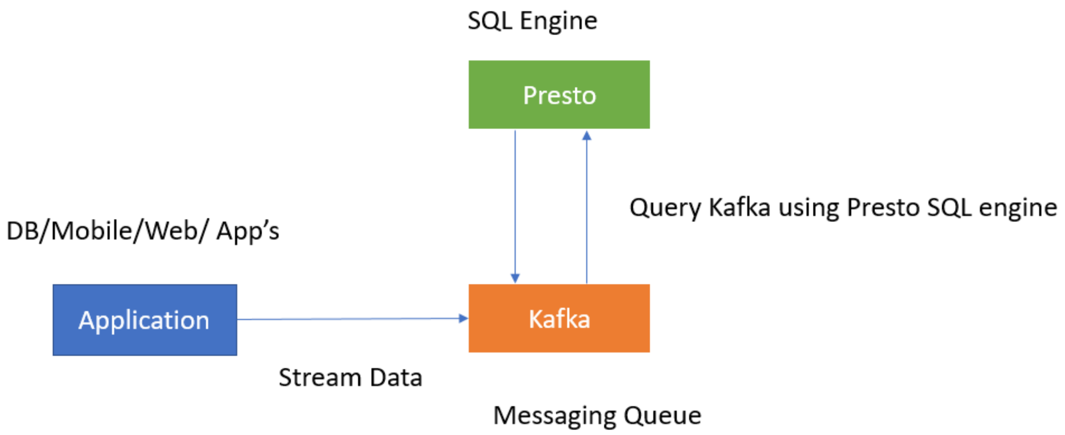
Apache Phoenix takes your SQL query, compiles it into a series of HBase scans, and orchestrates the running of those scans to produce regular JDBC result sets. Direct use of the HBase API, along with coprocessors and custom filters, results in performance on the order of milliseconds for small queries, or seconds for tens of millions of rows.

### Modules

Data Storage - Store the Data in HBase.
Data Processing - Process the Data with **Phoenix Engine**.

## Kafka Presto Integration

### Architecture



### What is?

- **Kafka** - Apache Kafka provides low-latency, high-throughput, fault-tolerant publish and subscribe pipelines and can process streams of events.
- **Presto** - Presto is an open source distributed SQL query engine for running interactive analytic queries against data sources of all sizes ranging from gigabytes.

### Description

Kafka with Presto integration enables SQL on top of Kafka.

### Why

Kafka is a leading Messaging queue system. The Kafka Connector for Presto allows access to live topic data from Apache Kafka using Presto. (SQL engine). It saved lots of time debugging data issues in data pipelines.

### Modules

Data Pipeline - Use Kafka to retrieve Realtime data from applications.
Data Query Engine - Presto for querying and debugging Kafka.

**Thank You See You Soon**