1. What is Hadoop?

**Apache Hadoop** is an **open source**, **Scalable**, and **Fault tolerant** framework written in **Java** from the**ASF** – Apache Software Foundation. It efficiently processes large volumes of data on a cluster of commodity hardware. Hadoop is not only a storage system but is a platform for large data storage as well as processing. Most of Hadoop code is written by Yahoo, IBM, Facebook, Cloudera.

Inspiration from **Google** written a paper about the technologies like [**Map-Reduce**](http://data-flair.training/blogs/hadoop-mapreduce-tutorial-comprehensive-guide-beginners/) programming model as well as its file system (**GFS**), Hadoop was originally written for the Nutch search engine project. When Doug Cutting and his team were working on it, very soon Hadoop became a top-level project due to its huge popularity.

1. What is Hadoop Technology?

Apache Hadoop provides parallel processing of data as it works on multiple machines simultaneously. an efficient framework for running jobs on multiple nodes of clusters. Hadoop can be [**setup on a single machine (pseudo-distributed mode)**](http://data-flair.training/blogs/install-cloudera-hadoop-cdh5-ubuntu/)but it shows its real power with a cluster of machines. **Cluster** means a group of systems connected via LAN.

Hadoop consists of three key parts –

* [**Hadoop Distributed File System**](http://data-flair.training/blogs/comprehensive-hdfs-guide-introduction-architecture-data-read-write-tutorial/) **(HDFS) –**It is the storage layer of Hadoop.
* [**Map-Reduce**](http://data-flair.training/blogs/hadoop-mapreduce-introduction-tutorial-comprehensive-guide/)**–**It is the data processing layer of Hadoop.
* [**YARN**](http://data-flair.training/blogs/hadoop-yarn-tutorial/)**–**It is the resource management layer of Hadoop.

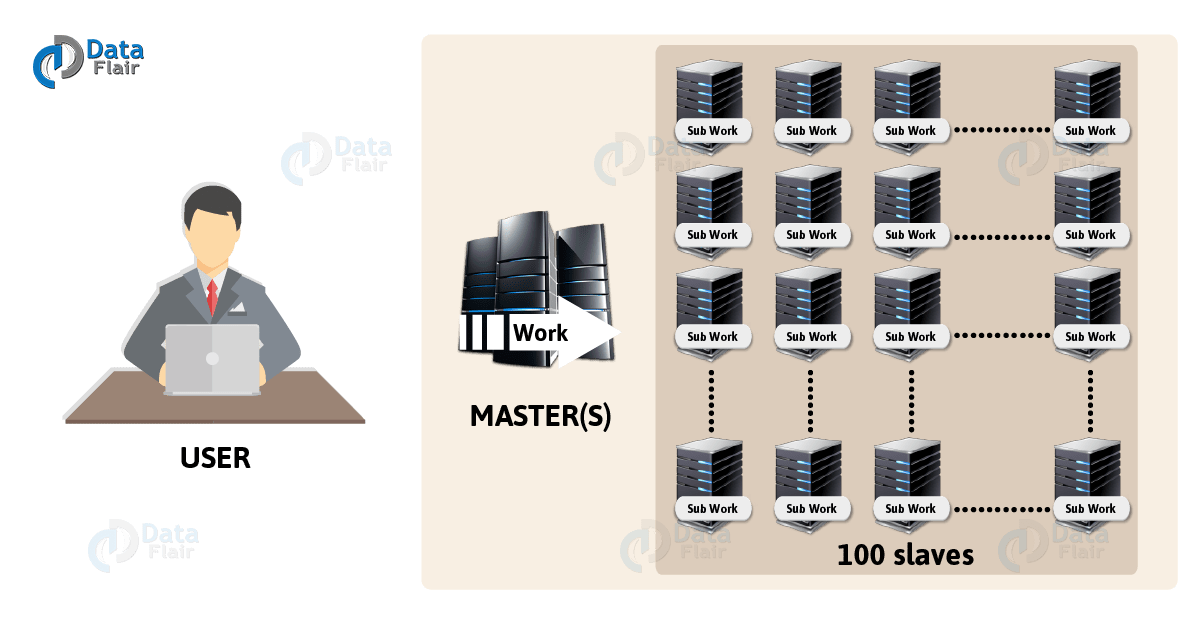
1. Advantage of Hadoop?

Following *characteristics of Hadoop* make it a unique platform:

* Flexibility to store and mine any type of data whether it is structured, semi-structured or unstructured. It is not bounded by a single schema.
* Excels at processing data of complex nature. Its scale-out architecture divides workloads across many nodes. Another added advantage is that its flexible file-system eliminates ETL bottlenecks.
* Scales economically, as discussed it can deploy on commodity hardware. Apart from this its open-source nature guards against vendor lock.

1. What is Hadoop Architecture?

Hadoop works in **master-slave** fashion. There is a master node and there are n numbers of slave nodes. Master manages, maintains and monitors the slaves while slaves are the actual worker nodes. In Hadoop architecture, the Master should deploy on good configuration hardware, not just commodity hardware. Master stores the metadata (data about data) while slaves are the nodes which store the data. The client connects with master node to perform any task.

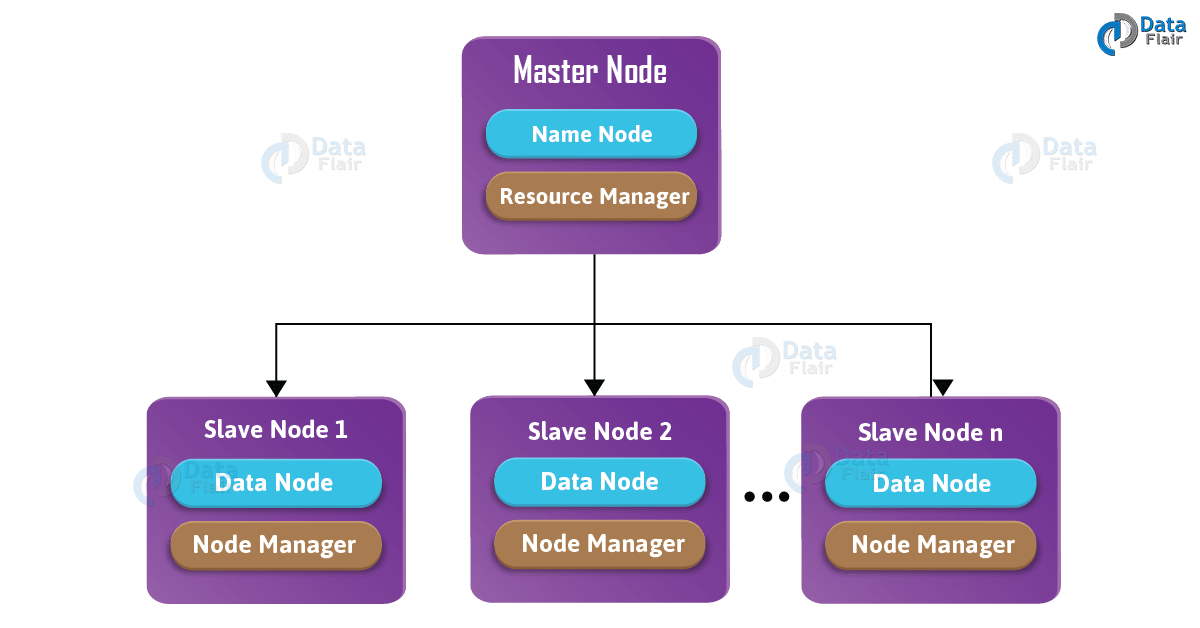


1. What is Hadoop Deamons?

Daemons are the processes that run in the background. There are mainly 4 daemons which run for Hadoop.

* **Namenode** – It runs on master node for HDFS.
* **Datanode** – It runs on slave nodes for HDFS.
* **ResourceManager** – It runs on master node for Yarn.
* **NodeManager** – It runs on slave node for Yarn.

These 4 demons run for Hadoop to be functional. Apart from this, there can be secondary NameNode, standby NameNode, Job HistoryServer, etc.



1. What are Different Flavour of Hadoop?

All flavors are almost same and if you know one, you can easily work on other flavors as well.

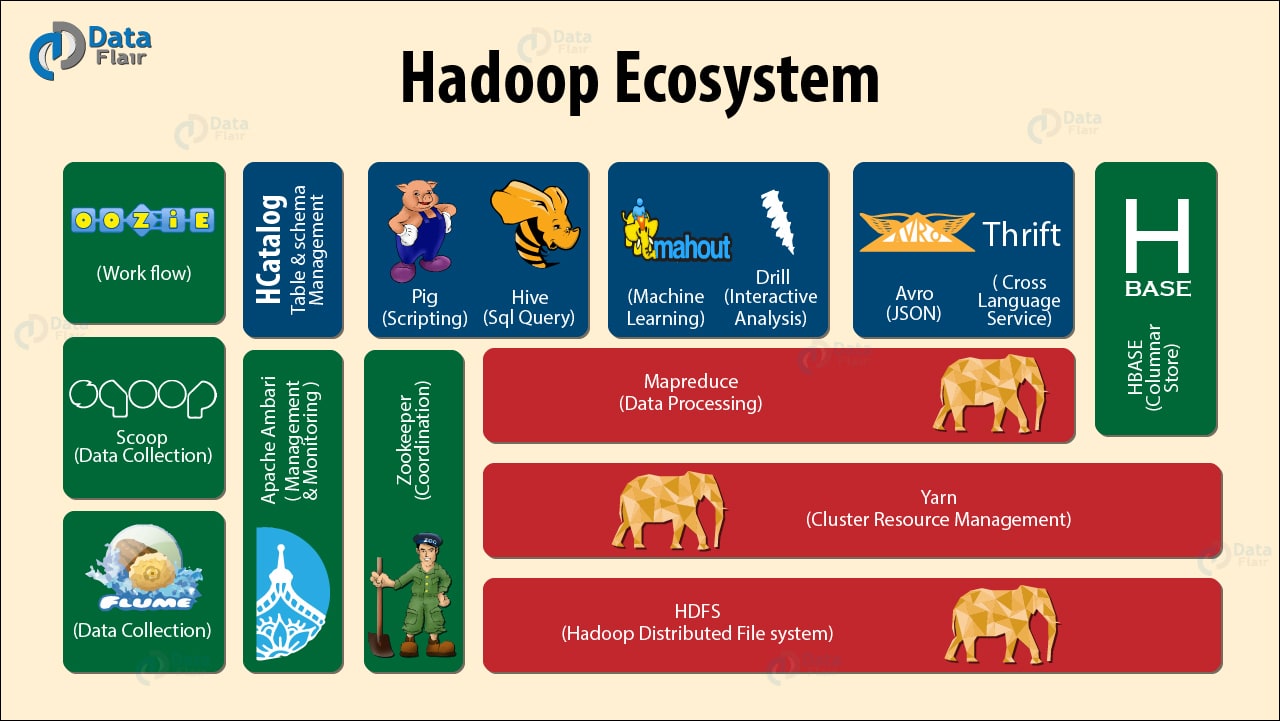
* **Apache** – Vanilla flavor, as the actual code is residing in Apache repositories.
* **Hortonworks** – Popular distribution in the industry.
* **Cloudera** – It is the most popular in the industry.
* **MapR** – It has rewritten HDFS and its HDFS is faster as compared to others.
* **IBM** – Proprietary distribution is known as Big Insights.

All the databases have provided native connectivity with Hadoop for fast data transfer. Because, to transfer data from Oracle to Hadoop, you need a connector.

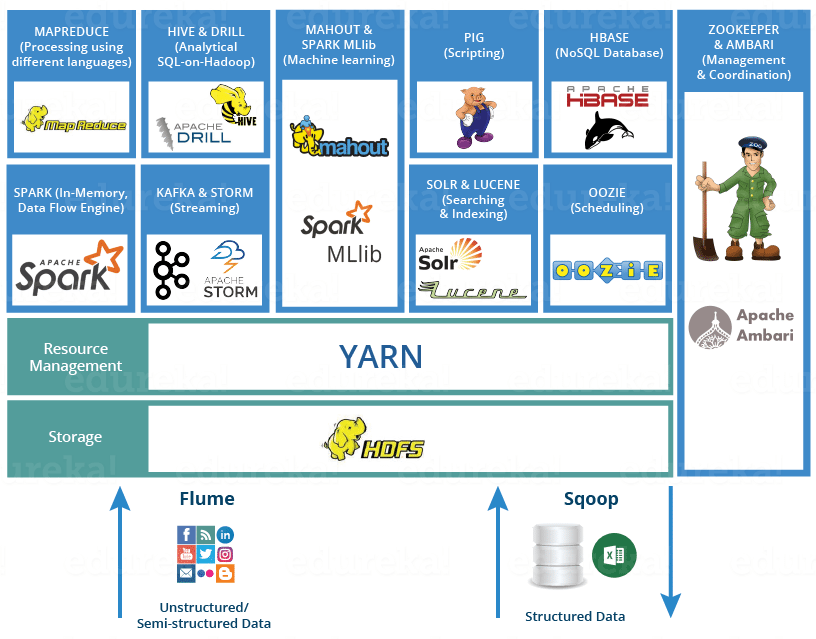
1. What is Hadoop Ecosystem?

Hadoop Ecosystem is neither a programming language nor a service, it is a platform or framework which solves big data problems. It as a suite which encompasses many services (ingesting, storing, analyzing and maintaining) inside it.

1. What is Hadoop Ecosystem?



*There are 15 Hadoop Ecosystem Components:*

* [**Hadoop HDFS**](http://data-flair.training/blogs/introduction-tutorial-hdfs/)**–** Distributed storage layer for Hadoop (*Hadoop Distributed File System*).
* [**Yarn Hadoop**](http://data-flair.training/blogs/list-best-books-learn-apache-yarn/)**–** Resource management layer introduced in Hadoop 2.x (*Yet Another Resource Negotiator*).
* [**Hadoop Map-Reduce**](http://data-flair.training/blogs/hadoop-mapreduce-flow-how-data-flows-in-mapreduce/)**–** Parallel data processing layer for Hadoop.
* [**HBase**](http://data-flair.training/blogs/hbase-tutorial-beginners-guide/)**–** It is a column-oriented database that runs on top of HDFS. It is a NoSQL database which does not understand the structured query. For sparse data set, it suits well (*NoSQL Database*).
* [**Hive**](http://data-flair.training/blogs/hive-tutorial-an-introductory-guide-for-beginners/)**–** Apache Hive is a data warehousing infrastructure based on Hadoop and it enables easy data summarization, using SQL queries.
* [**Pig**](http://data-flair.training/blogs/introduction-tutorial-apache-pig/)**–** It is a top-level scripting language. As we use it with Hadoop. Pig enables writing complex data processing without Java programming.
* [**Flume**](http://data-flair.training/blogs/introduction-apache-flume-tutorial-beginners-guide/)**–** It is a reliable system for efficiently collecting large amounts of log data from many different sources in real-time.
* **Sqoop –** It is a tool design to transport huge volumes of data between Hadoop and RDBMS.
* **Oozie –** It is a Java Web application uses to schedule Apache Hadoop jobs. It combines multiple jobs sequentially into one logical unit of work (*Job Scheduling*).
* **Zookeeper** – A centralized service for maintaining configuration information, naming, providing distributed synchronization, and providing group services (*Managing Cluster*).
* **Mahout/Spark MLlib –** A library of scalable machine-learning algorithms, implemented on top of Apache Hadoop and using the MapReduce paradigm.
* **Spark –** In memory data processing layer.
* [**Apache Drill**](https://www.edureka.co/blog/hadoop-ecosystem#apache_drill) - *SQL on Hadoop*
* [**Solr & Lucene**](https://www.edureka.co/blog/hadoop-ecosystem#apache_solr_and_lucene) - *Searching & Indexing*
* [**Ambari**](https://www.edureka.co/blog/hadoop-ecosystem#apache_ambari)- *Provision, Monitor and Maintain cluster*
* [**HCatalog**](https://www.edureka.co/blog/hadoop-ecosystem#apache_ambari) - *Provision, Monitor and Maintain cluster*
* **Avro** - *Provision, Monitor and Maintain cluster*
* **Thrift** - *Provision, Monitor and Maintain cluster*
* 

1. What is HDFS?

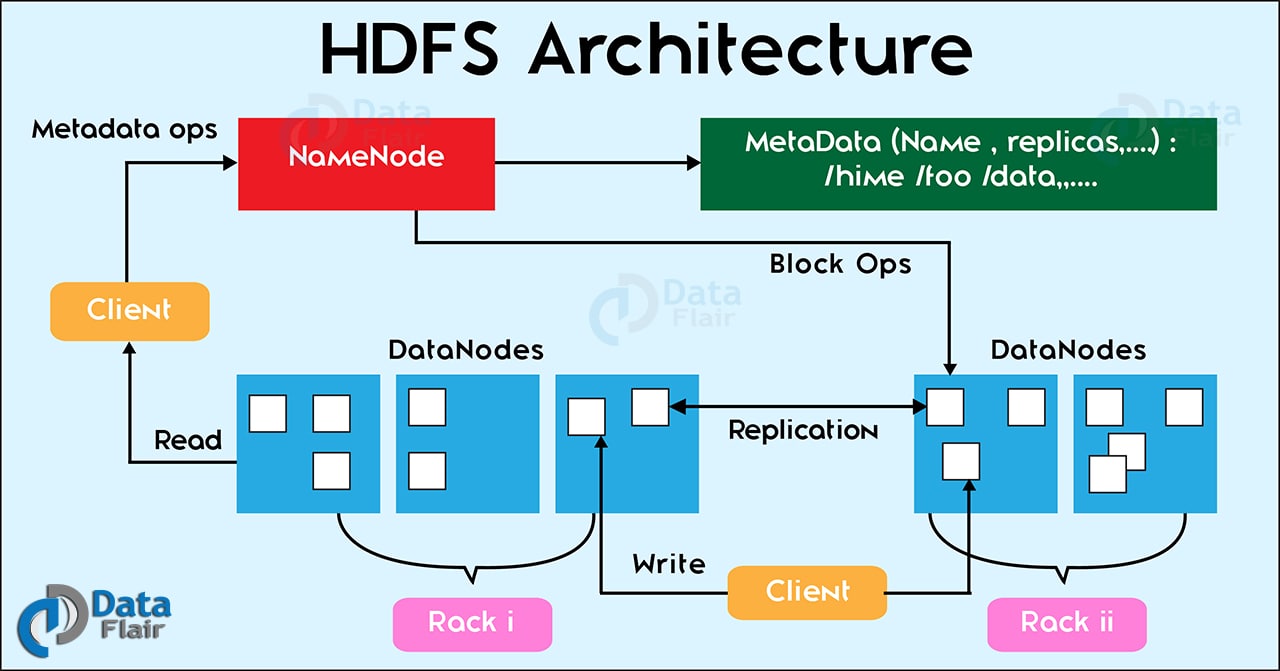
**HDFS** is the primary storage system of Hadoop, the backbone of Hadoop Ecosystem. Hadoop distributed file system (HDFS) is a java based file system that provides scalable, fault tolerance, reliable and cost efficient data storage for [**Big data**](http://data-flair.training/blogs/what-is-big-data/).

* HDFS is the one, which makes it possible to store different types of large data sets (i.e. structured, unstructured and semi structured data).
* HDFS creates a level of abstraction over the resources, from where we can see the whole HDFS as a single unit.
* It helps us in storing our data across various nodes and maintaining the log file about the stored data (metadata).

In Hadoop Architecture on the master node, a daemon called ***namenode*** run for HDFS. On all the slaves a daemon called ***datanode*** run for HDFS

A file is split up into blocks (default 128 MB) and stored distributedly across multiple machines. These blocks replicate as per the replication factor. After replication, it stored at different nodes.

1. What are HDFS Components?



There are two major components of Hadoop HDFS- NameNode and DataNode.

**i. NameNode**

It is also known as*Master* node. NameNode does not store actual data or dataset. NameNode stores Metadata i.e. number of [**blocks**](http://data-flair.training/blogs/data-blocks-hdfs-hadoop-distributed-file-system/), their location, on which Rack, which Datanode the data is stored and other details. It consists of files and directories.

**Tasks of HDFS NameNode**

* Manage file system namespace.
* Regulates client’s access to files.
* Executes file system execution such as naming, closing, opening files and directories.

**ii. DataNode**

It is also known as *Slave*. HDFS Datanode is responsible for storing actual data in HDFS. Datanode performs [**read and write operation**](http://data-flair.training/blogs/hadoop-hdfs-data-read-and-write-operations/) as per the request of the clients. Replica block of Datanode consists of 2 files on the file system. The first file is for data and second file is for recording the block’s metadata. HDFS Metadata includes checksums for data. At startup, each Datanode connects to its corresponding Namenode and does handshaking. Verification of namespace ID and software version of DataNode take place by handshaking. At the time of mismatch found, DataNode goes down automatically.

**Tasks of HDFS DataNode**

* DataNode performs operations like block replica creation, deletion, and replication according to the instruction of NameNode.
* DataNode manages data storage of the system.

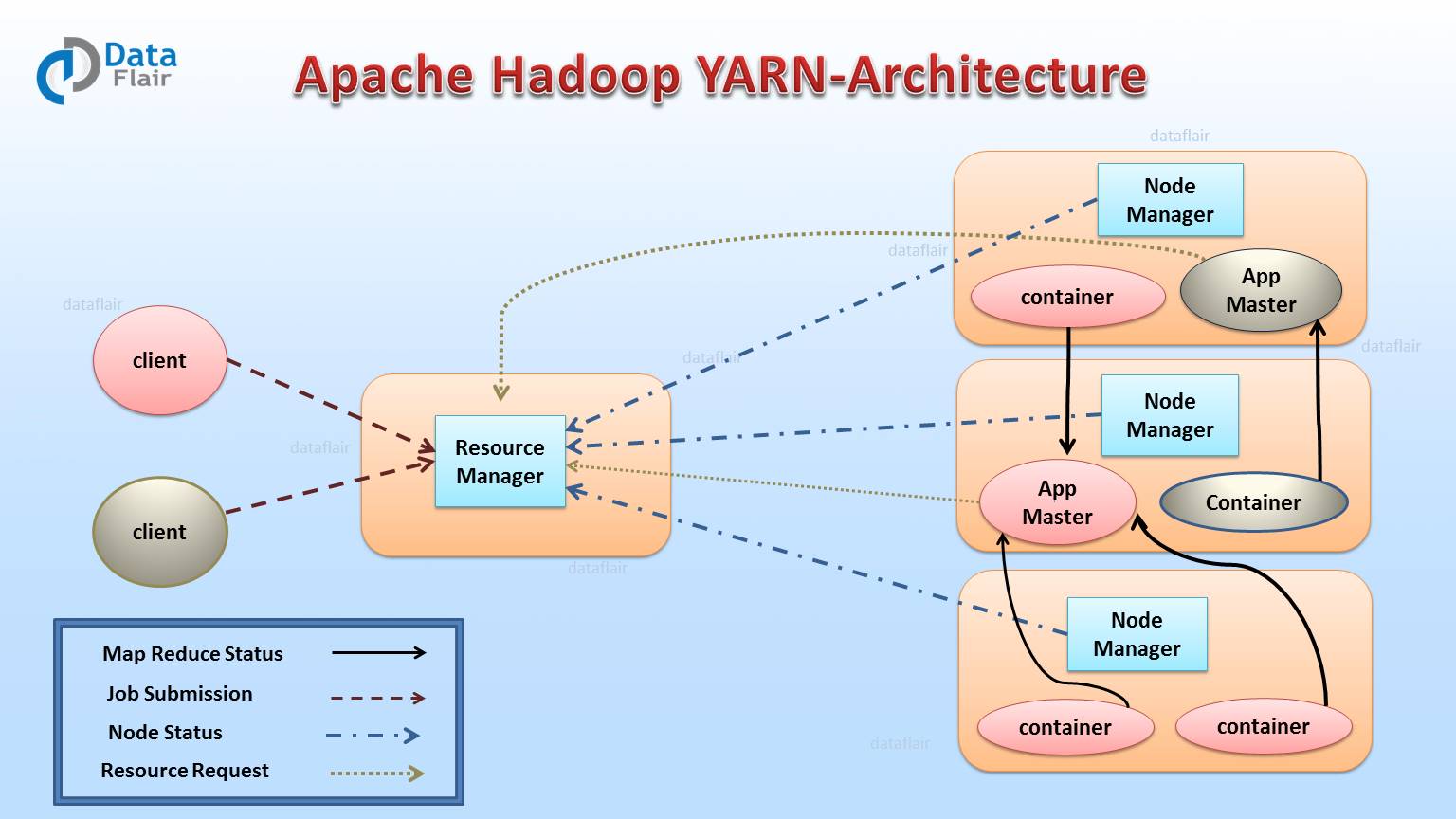
1. What is YARN?

YARN as the brain of your Hadoop Ecosystem. It performs all your processing activities by allocating resources and scheduling tasks. YARN is called as the operating system of Hadoop as it is responsible for managing and monitoring workloads. It allows multiple data processing engines such as real-time streaming and batch processing to handle data stored on a single platform.

It has two major components, i.e. **ResourceManager and NodeManager**.

* **ResourceManager** is main node in the processing department. It receives the processing requests, and then passes the parts of requests to corresponding NodeManagers accordingly, where the actual processing takes place. ResourceManager has two components:
  + **Schedulers:**Based on your application resource requirements, Schedulers perform scheduling algorithms and allocates the resources.
  + **ApplicationsManager:**While ApplicationsManager accepts the job submission, negotiates to containers (i.e. the Data node environment where process executes) for executing the application specific ApplicationMaster and monitoring the progress. ApplicationMasters are the deamons which reside on DataNode and communicates to containers for execution of tasks on each DataNode.
* **NodeManagers** are installed on every DataNode. It is responsible for execution of task on every single DataNode.

On the master node, the [**ResourceManager**](http://data-flair.training/blogs/deep-dive-into-hadoop-yarn-resource-manager/) daemon runs for the YARN then for all the slave nodes [**NodeManager**](http://data-flair.training/blogs/deep-dive-into-hadoop-yarn-node-manager/) daemon runs.

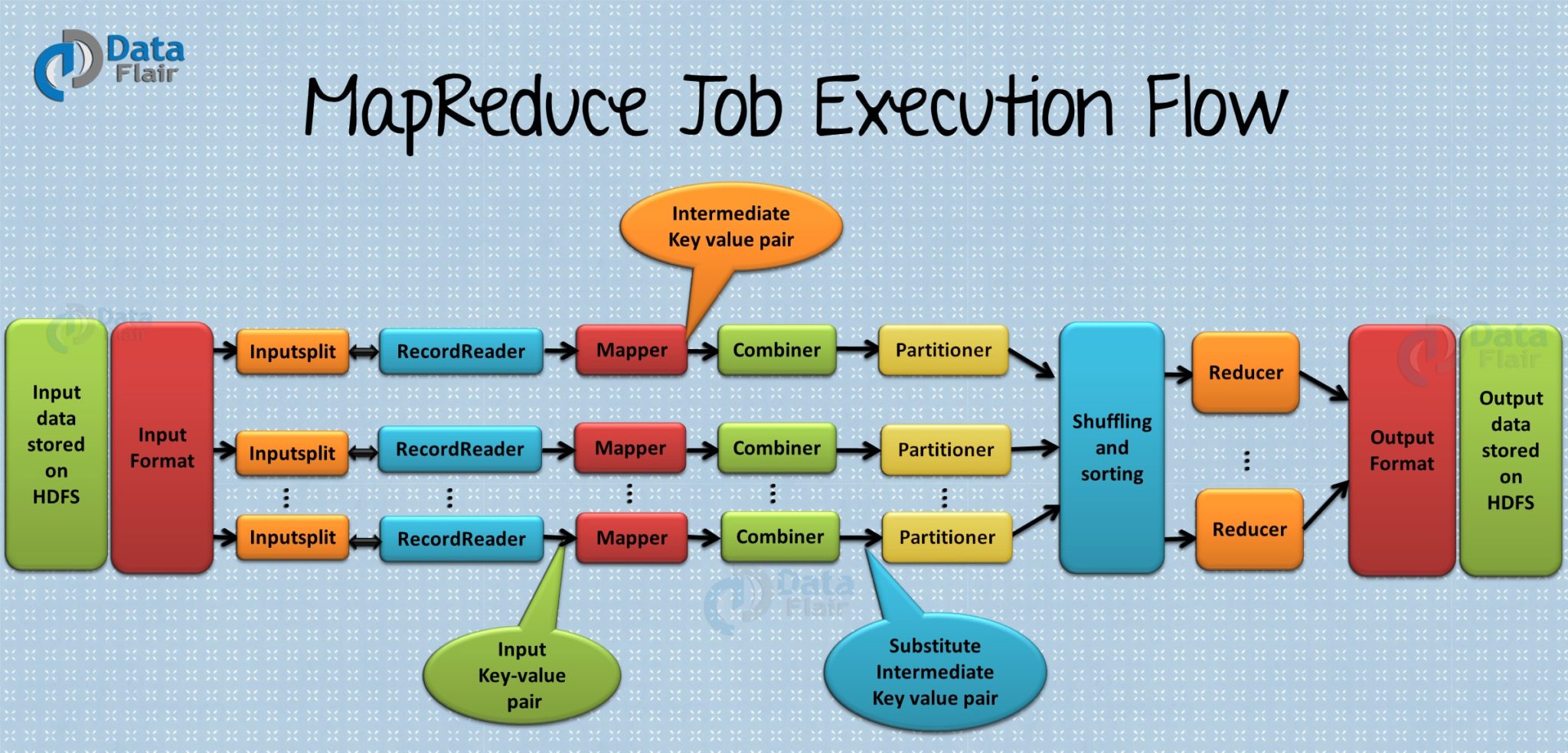
[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/Apache-YARN-architecture-min-1.jpg)

Main features of YARN are:

* **Flexibility –** Enables other purpose-built data processing models beyond MapReduce (batch), such as interactive and streaming. Due to this feature of YARN, other applications can also be run along with Map Reduce programs in Hadoop2.
* **Efficiency –** As many applications run on the same cluster, Hence, efficiency of Hadoop increases without much effect on quality of service.
* **Shared –** Provides a stable, reliable, secure foundation and shared operational services across multiple workloads. Additional programming models such as graph processing and iterative modeling are now possible for data processing.

1. What is Map-Reduce?

MapReduce is the core component of processing in a Hadoop Ecosystem as it provides the logic of processing. It is a software framework for easily writing applications that process the vast amount of structured and unstructured data stored in the Hadoop Distributed File system. MapReduce programs are parallel in nature, thus are very useful for performing large-scale data analysis using multiple machines in the cluster.

[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/mapreduce-job-execution-flow-1-1.jpg)

**Working of MapReduce**

Hadoop Ecosystem component ‘MapReduce’ works by breaking the processing into two phases. Each phase has [**key-value pairs**](http://data-flair.training/blogs/key-value-pairs-hadoop-mapreduce/) as input and output. In addition, programmer also specifies two functions:**map function**and**reduce function**

* Map phase
* Reduce phase

**Map** **function** takes a set of data and converts it into another set of data, where individual elements are broken down into tuples (key/value pairs). **Map function** performs actions like filtering, grouping and sorting. The result generated by the Map function is a key value pair (K, V) which acts as the input for Reduce function.

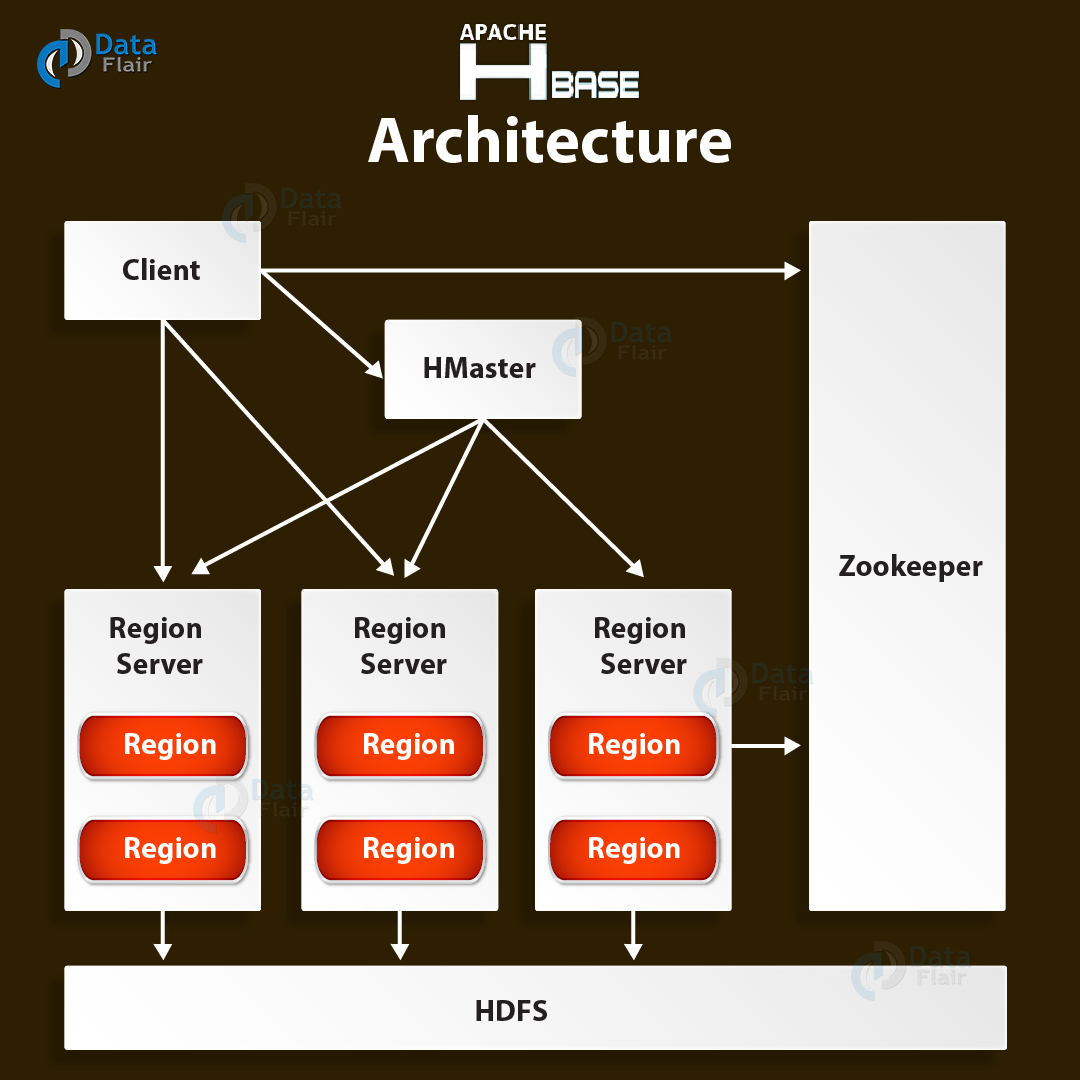
**Reduce** **function**takes the output from the Map as an input and combines those data tuples based on the key and accordingly modifies the value of the key. **Reduce function** aggregates and summarizes the result produced by map function.

**Features of MapReduce**

* **Simplicity –**MapReduce jobs are easy to run. Applications can be written in any language such as **java, C++**, and **python**.
* **Scalability –**MapReduce can process petabytes of data.
* **Speed –**By means of parallel processing problems that take days to solve, it is solved in hours and minutes by MapReduce.
* **Fault Tolerance –**MapReduce takes care of failures. If one copy of data is unavailable, another machine has a copy of the same key pair which can be used for solving the same subtask.

1. What is HBase?

**Apache HBase** is a Hadoop ecosystem component which is distributed database that was designed to store structured data in tables that could have billions of row and millions of columns. HBase is scalable, distributed, and Nosql database that is built on top of HDFS. HBase, provide real time access to read or write data in HDFS.

[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/Apache-HBase-Architecture-01.jpg)**Components of Hbase**

There are two HBase Components namely- HBase Master and RegionServer.

**i. HBase Master**

It is not part of the actual data storage but negotiates load balancing across all RegionServer.

* Maintain and monitor the [**Hadoop cluster.**](http://data-flair.training/blogs/install-hadoop-2-x-ubuntu-hadoop-multi-node-cluster/)
* Performs administration (interface for creating, updating and deleting tables.)
* Controls the failover.
* HMaster handles DDL operation.

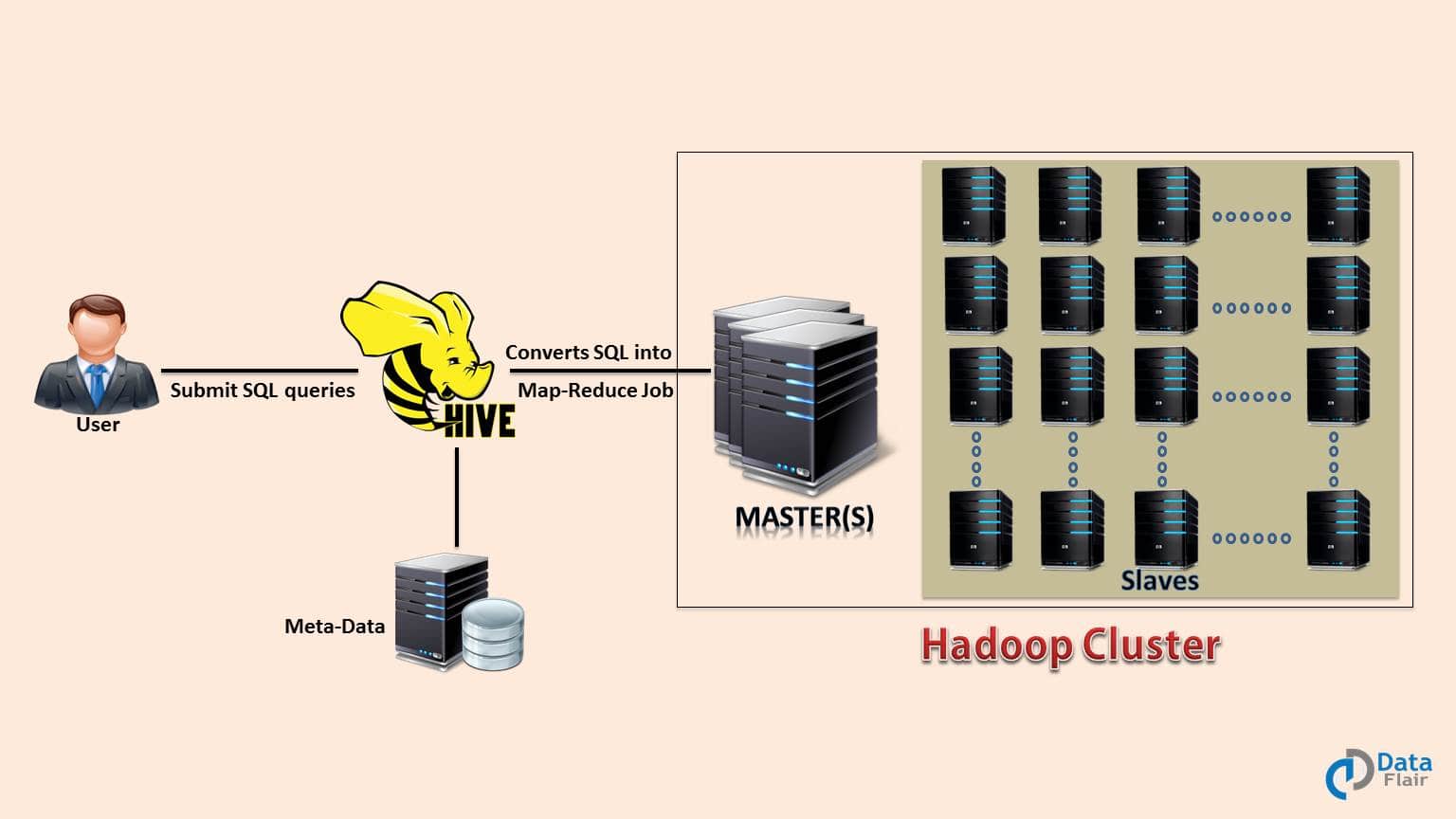
**ii. RegionServer**

It is the worker node which handle read, write, update and delete requests from clients. Region server process runs on every node in Hadoop cluster. Region server runs on HDFS DateNode.

1. What is Hive?

The Hadoop ecosystem component,**Apache Hive,** is an open source data warehouse system for querying and analyzing large datasets stored in Hadoop files. Hive do three main functions: *data summarization, query, and analysis*.

Hive use language called **HiveQL** (HQL), which is similar to SQL. HiveQL automatically translates SQL-like queries into [**MapReduce jobs**](http://data-flair.training/blogs/hadoop-mapreduce-job-execution-flow/) which will execute on Hadoop.

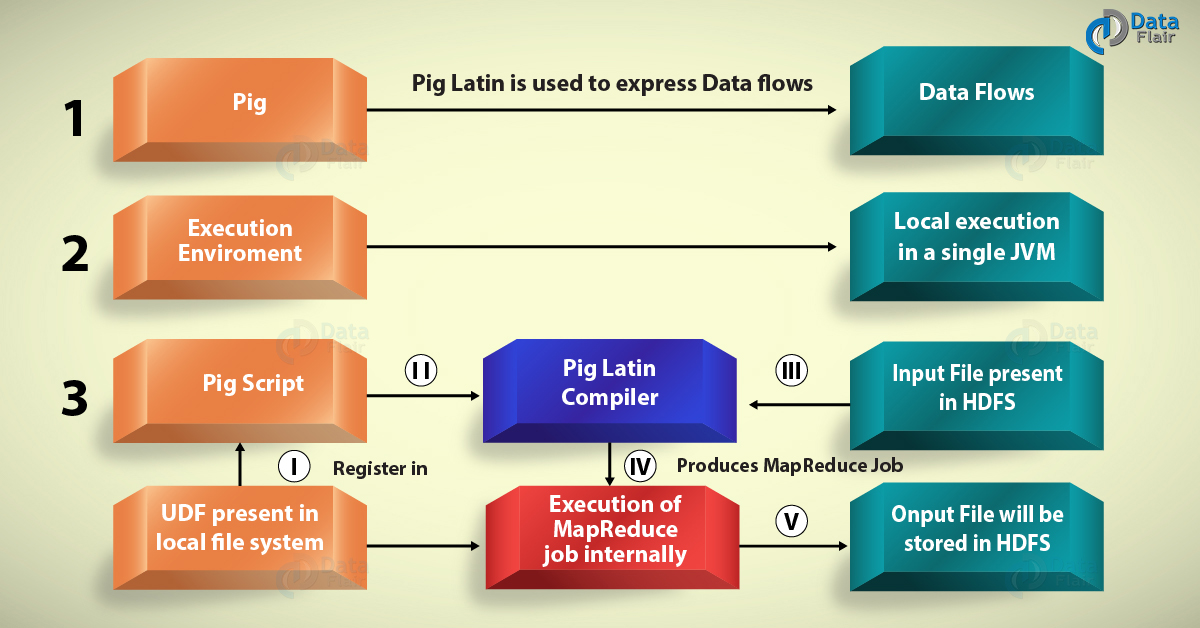
[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/Apache-Hive-Introduction-Tutorial-Training-1.jpg)

**Main parts of Hive are:**

* **Metastore –** It stores the metadata.
* **Driver –** Manage the lifecycle of a HiveQL statement.
* **Query compiler –** Compiles HiveQL into Directed Acyclic Graph(DAG).
* **Hive server –** Provide a thrift interface and JDBC/ODBC server.

1. What is Pig?

**Apache Pig** is a high-level language platform for analyzing and querying huge dataset that are stored in HDFS. Pig as a component of Hadoop Ecosystem uses ***PigLatin***language. It is very similar to SQL. It loads the data, applies the required filters and dumps the data in the required format. For Programs execution, pig requires Java runtime environment.

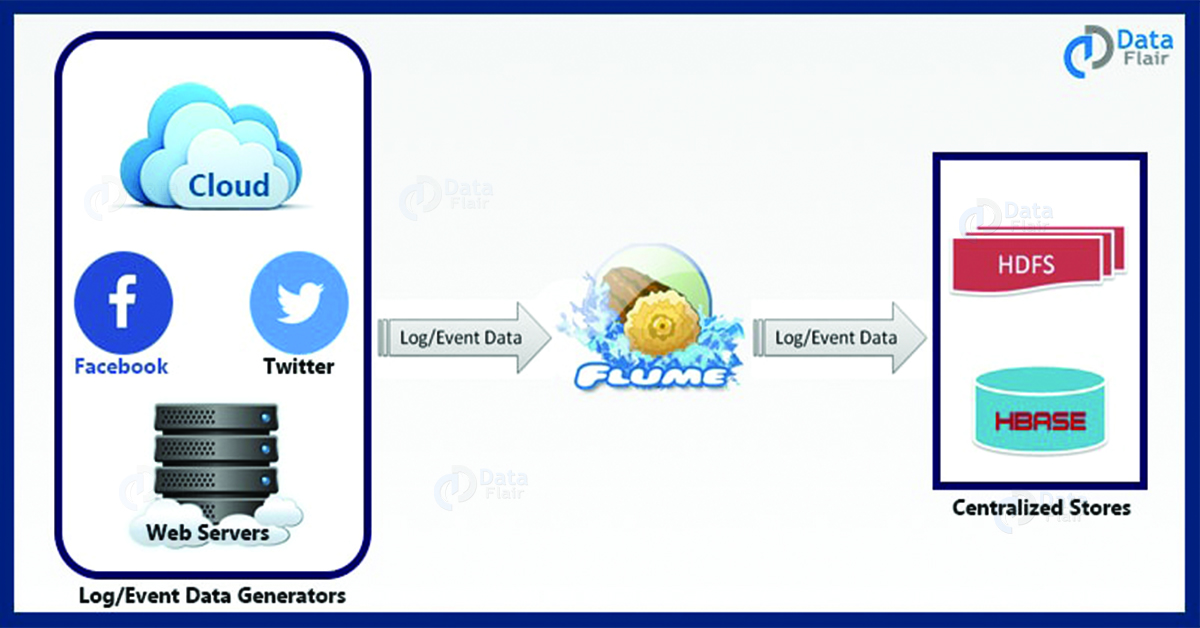
[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/Pig.jpg)

**Features of Apache Pig:**

* **Extensibility –**For carrying out special purpose processing, users can create their own function.
* **Optimization opportunities –**Pig allows the system to optimize automatic execution. This allows the user to pay attention to semantics instead of efficiency.
* **Handles all kinds of data –**Pig analyzes both structured as well as unstructured.

1. What is Flume?

**Flume** efficiently collects, aggregate and moves a large amount of data from its origin and sending it back to HDFS. It is fault tolerant and reliable mechanism. This Hadoop Ecosystem component allows the data flow from the source into Hadoop environment. It uses a simple extensible data model that allows for the online analytic application. Using Flume, we can get the data from multiple servers immediately into hadoop.

[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/Apache-Flume-Introduction-Tutorial-Training-2.jpg)

1. What is Scoop?

**Sqoop** imports data from external sources into related Hadoop ecosystem components like HDFS, Hbase or Hive. It also exports data from Hadoop to other external sources. Sqoop works with relational databases such as teradata, Netezza, oracle, MySQL.

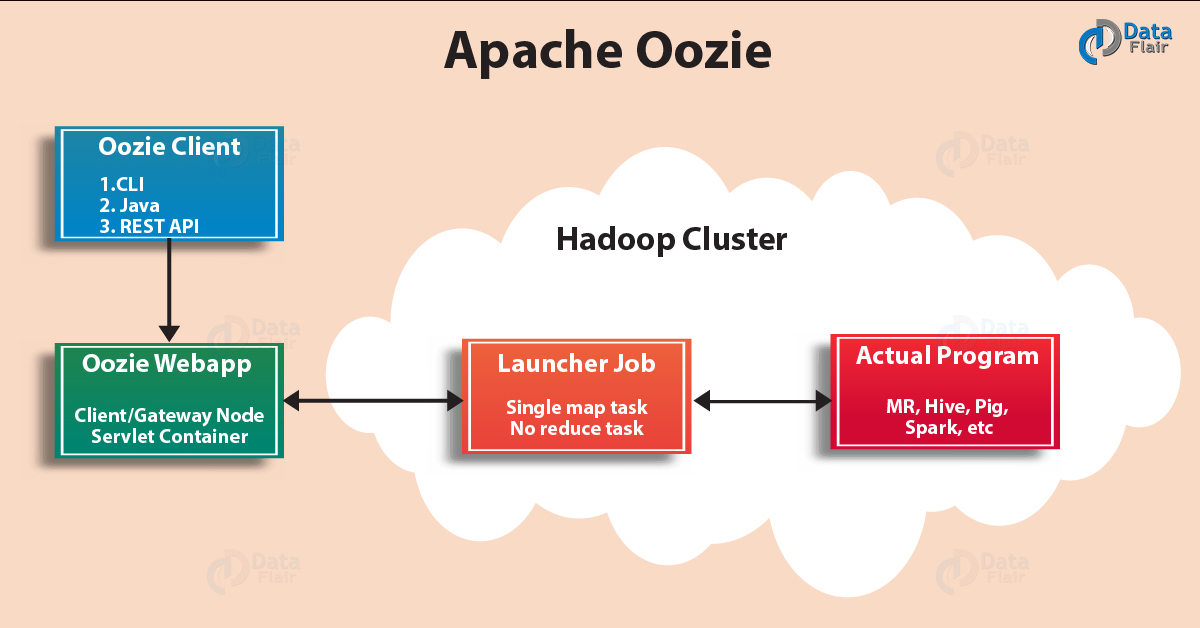
[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/How-Sqoop-Works-01.jpg)

**Features of Apache Sqoop:**

* **Import sequential datasets from mainframe –** Sqoop satisfies the growing need to move data from the mainframe to HDFS.
* **Import direct to ORC files –** Improves compression and light weight indexing and improve query performance.
* **Parallel data transfer –** For faster performance and optimal system utilization.
* **Efficient data analysis –** Improve efficiency of data analysis by combining structured data and unstructured data on a schema on reading data lake.
* **Fast data copies –** from an external system into Hadoop.

1. What is Oozie?

It is a workflow scheduler system for managing apache Hadoop jobs. Oozie combines multiple jobs sequentially into one logical unit of work. Oozie framework is fully integrated with apache Hadoop stack, YARN as an architecture center and supports Hadoop jobs for apache MapReduce, Pig, Hive, and Sqoop.

[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/Apache-Oozie-01.jpg)

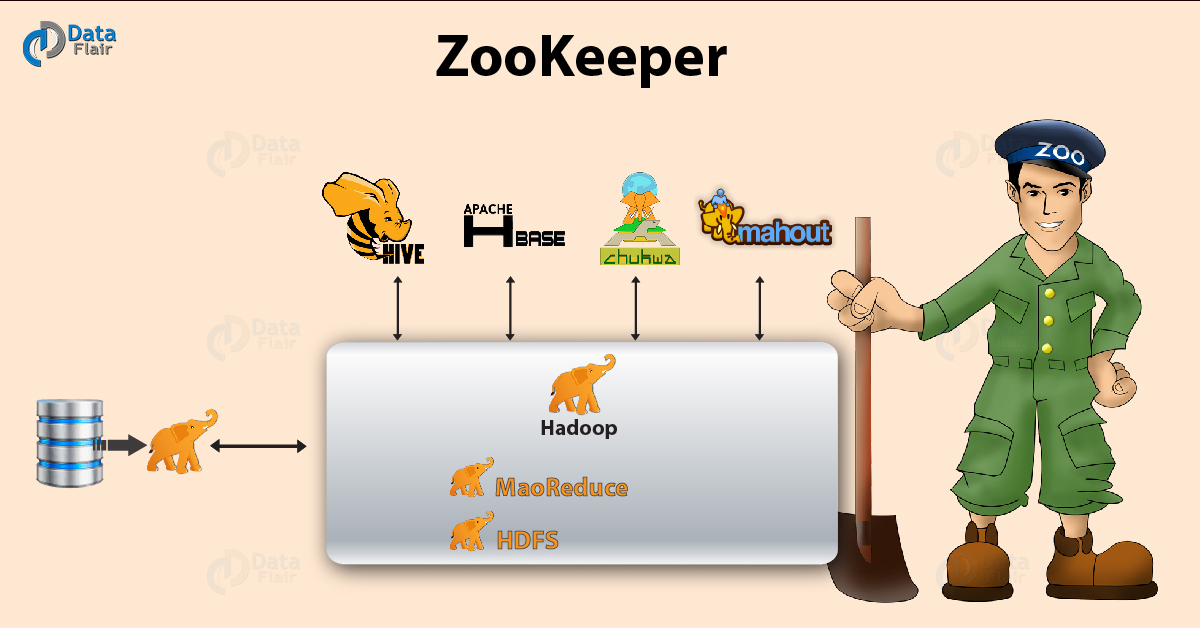
In Oozie, users can create Directed Acyclic Graph of workflow, which can run in parallel and sequentially in Hadoop. Oozie is scalable and can manage timely execution of thousands of workflow in a [**Hadoop cluster**](http://data-flair.training/blogs/installation-hadoop-3-x-ubuntu-pseudo-distributed-mode/). Oozie is very much flexible as well. One can easily start, stop, suspend and rerun jobs. It is even possible to skip a specific failed node or rerun it in Oozie.

There are two basic types of Oozie jobs:

* **Oozie workflow –**It is to store and run workflows composed of Hadoop jobs e.g., MapReduce, pig, Hive.
* **Oozie Coordinator –** It runs workflow jobs based on predefined schedules and availability of data.

1. What is Zookeeper?

**Apache Zookeeper** is a centralized service and a Hadoop Ecosystem component for maintaining configuration information, naming, providing distributed synchronization, and providing group services. Zookeeper manages and coordinates a large cluster of machines.

[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/ZooKeeper-01.jpg)

**Features of Zookeeper:**

* **Fast –** Zookeeper is fast with workloads where reads to data are more common than writes. The ideal read/write ratio is 10:1.
* **Ordered –** Zookeeper maintains a record of all transactions.

1. What is Mahoot/Spark MLlib?

**Mahout** is open source framework for creating scalable [**machine learning**](http://data-flair.training/blogs/machine-learning-tutorial/)algorithm and data mining library. Once data is stored in Hadoop HDFS, mahout provides the data science tools to automatically find meaningful patterns in those big data sets.

**Algorithms of Mahout are:**

* **Clustering –** Here it takes the item in particular class and organizes them into naturally occurring groups, such that item belonging to the same group are similar to each other.
* **Collaborative filtering –** It mines user behavior and makes product recommendations (e.g. Amazon recommendations)
* **Classifications –** It learns from existing categorization and then assigns unclassified items to the best category.
* **Frequent pattern mining –** It analyzes items in a group (e.g. items in a shopping cart or terms in query session) and then identifies which items typically appear together.

1. What is Spark?

Apache Spark is a framework for real time data analytics in a distributed computing environment. The Spark is written in Scala and was originally developed at the University of California, Berkeley. It executes in-memory computations to increase speed of data processing over Map-Reduce. It is 100x faster than Hadoop for large scale data processing by exploiting in-memory computations and other optimizations. Therefore, it requires high processing power than Map-Reduce.



As you can see, Spark comes packed with high-level libraries, including support for R, SQL, Python, Scala, Java etc. These standard libraries increase the seamless integrations in complex workflow. Over this, it also allows various sets of services to integrate with it like MLlib, GraphX, SQL + Data Frames, Streaming services etc. to increase its capabilities.

This is a very common question in everyone’s mind:

*“Apache Spark: A Killer or Saviour of Apache Hadoop?” – O’Reily*

The Answer to this – This is not an apple to apple comparison. Apache Spark best fits for real time processing, whereas Hadoop was designed to store unstructured data and execute batch processing over it. When we combine, Apache Spark’s ability, i.e. high processing speed, advance analytics and multiple integration support with Hadoop’s low-cost operation on commodity hardware, it gives the best results.

That is the reason why, Spark and Hadoop are used together by many companies for processing and analyzing their Big Data stored in HDFS.

1. What is Apache Drill?

The main purpose of the Hadoop Ecosystem Component is large-scale data processing including structured and semi-structured data. It is a low latency distributed query engine that is designed to scale to several thousands of nodes and query petabytes of data. The drill is the first distributed SQL query engine that has a schema-free model.

**Application of Apache drill**

The drill has become an invaluable tool at Cardlytics, a company that provides consumer purchase data for mobile and internet banking. Cardlytics is using a drill to quickly process trillions of records and execute queries.

**Features of Apache Drill**:

The drill has specialized memory management system to eliminates garbage collection and optimize memory allocation and usage. Drill plays well with Hive by allowing developers to reuse their existing Hive deployment.

* **Extensibility –**Drill provides an extensible architecture at all layers, including query layer, query optimization, and client API. We can extend any layer for the specific need of an organization.
* **Flexibility –** Drill provides a hierarchical columnar data model that can represent complex, highly dynamic data and allow efficient processing.
* **Dynamic schema discovery –** Apache drill does not require schema or type specification for data to start the query execution process. Instead, drill starts processing the data in units called record batches and discover schema on the fly during processing.
* **Drill decentralized metadata –**Unlike other SQL Hadoop technologies, the drill does not have centralized metadata requirement. Drill users do not need to create and manage tables in metadata to query data.

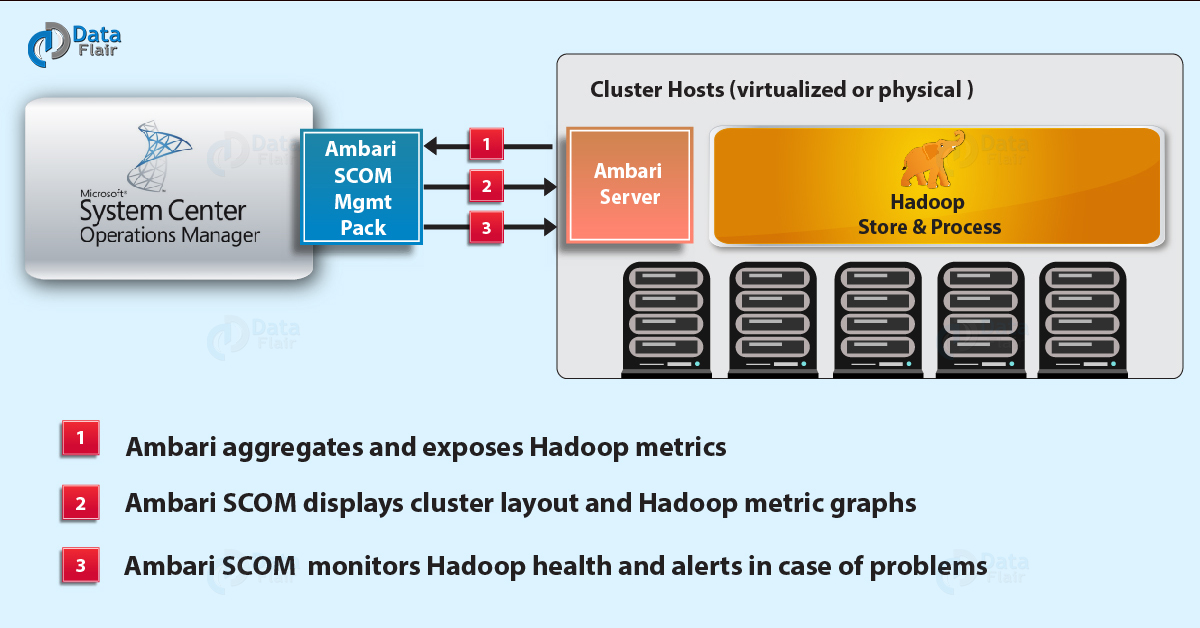
1. What is Solr and Lucene?

Apache Solr and Apache Lucene are the two services which are used for searching and indexing in Hadoop Ecosystem.

* Apache Lucene is based on Java, which also helps in spell checking.
* If Apache Lucene is the engine, Apache Solr is the car built around it. Solr is a complete application built around Lucene.
* It uses the Lucene Java search library as a core for search and full indexing.

1. What is Ambari?

Ambari, another Hadop ecosystem component, is a management platform for provisioning, managing, monitoring and securing apache Hadoop cluster. Hadoop management gets simpler as Ambari provide consistent, secure platform for operational control.

[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/ambari-scom-01.jpg)

**Features of Ambari:**

* **Simplified installation, configuration, and management –** Ambari easily and efficiently create and manage clusters at scale.
* **Centralized security setup –**Ambari reduce the complexity to administer and configure cluster security across the entire platform.
* **Highly extensible and customizable –** Ambari is highly extensible for bringing custom services under management.
* **Full visibility into cluster health –** Ambari ensures that the cluster is healthy and available with a holistic approach to monitoring.

1. What is HCatalog?

It is a table and storage management layer for Hadoop. HCatalog supports different components available in Hadoop ecosystem like MapReduce, Hive, and Pig to easily read and write data from the cluster. HCatalog is a key component of Hive that enables the user to store their data in any format and structure.

By default, HCatalog supports RCFile, CSV, JSON, sequenceFile and ORC file formats.

**Benefits of HCatalog:**

* Enables notifications of data availability.
* With the table abstraction, HCatalog frees the user from overhead of data storage.
* Provide visibility for data cleaning and archiving tools.

1. What is Avro?

**Avro** is an open source project that provides data serialization and data exchange services for Hadoop. These services can be used together or independently. Big data can exchange programs written in different languages using Avro.

Using serialization service programs can serialize data into files or messages. It stores data definition and data together in one message or file making it easy for programs to dynamically understand information stored in Avro file or message.

**Avro schema –** It relies on schemas for serialization/deserialization. Avro requires the schema for data writes/read. When Avro data is stored in a file its schema is stored with it, so that files may be processed later by any program.

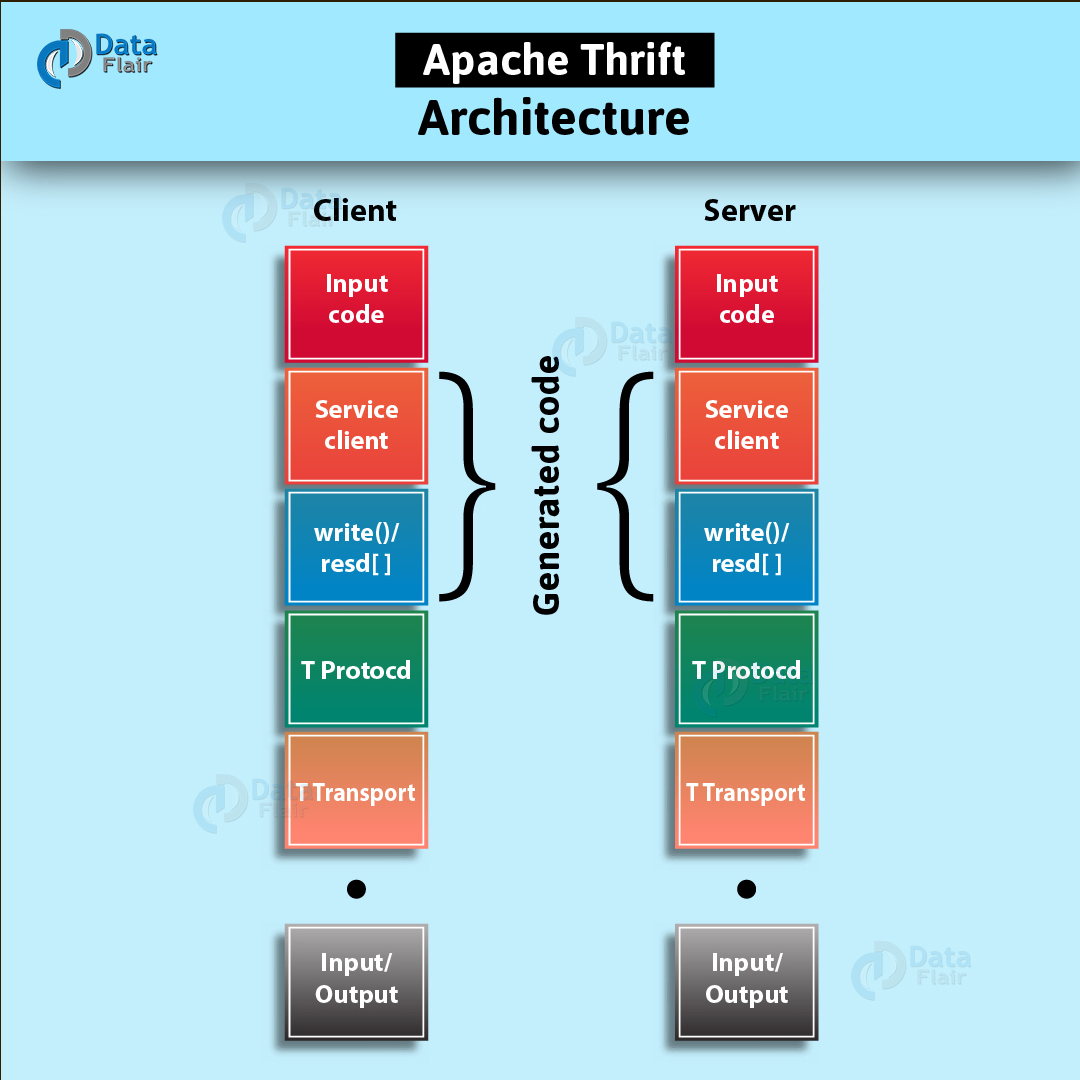
**Dynamic typing –** It refers to serialization and deserialization without code generation. It complements the code generation which is available in Avro for statically typed language as an optional optimization.

**Features provided by Avro:**

* Rich data structures.
* Remote procedure call.
* Compact, fast, binary data format.
* Container file, to store persistent data.

1. What is Thrift?

It is a software framework for scalable cross-language services development. Thrift is an interface definition language for RPC(Remote procedure call) communication. Hadoop does a lot of RPC calls so there is a possibility of using Hadoop Ecosystem componet Apache Thrift for performance or other reasons.

[](https://d2h0cx97tjks2p.cloudfront.net/blogs/wp-content/uploads/Apache-Thrift-Architecture-01.jpg)

1. Reference

<https://data-flair.training/blogs/hadoop-tutorial/>

<https://data-flair.training/blogs/hadoop-ecosystem-components/>

<https://www.edureka.co/blog/hadoop-ecosystem>