**IPL: Indian Premier Data League**

Code Components:

1. Data Storage: Amazon s3
2. Databricks
   * Transformation (Spark)
   * SQL Analytics: SQL
   * Visualization

**Ecosystem of Spark Core**

* Apache Spark Core: Executing every code that is passed to spark. It is heart of Spark
* Spark SQL: For SQL query
* Spark Streaming: For processing real time data
* MLlib (Machine learning): ML on large data
* GraphX (Graph): Graphical db

**Apache Spark**

Apache spark is unified computing engine and a set of libraries for parallel data processing on computer clusters Basically framework which partitions data in distributed way on machine and combines output at the end

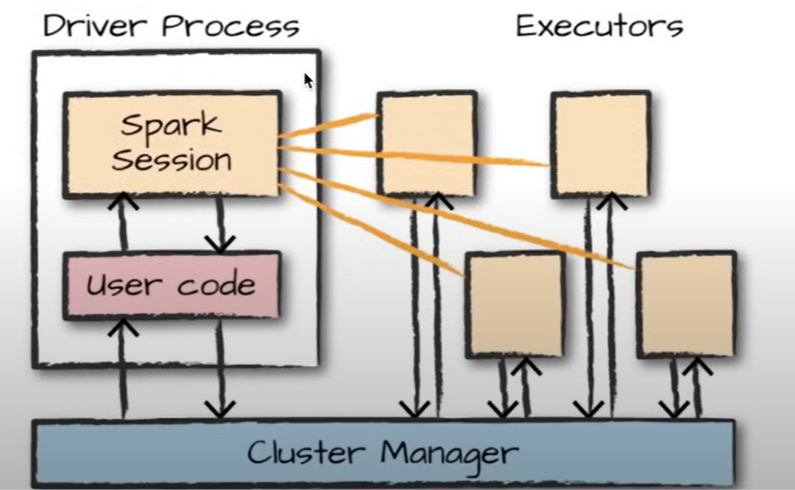
1. Structured Streaming, Advanced Analytics, Libraries and Ecosystem
2. Structured APIs: Datasets, DataFrames, SQL
3. Low level APIs: RDDs, Distributed Variables

**Architecture of Apache Spark**

Single machines do not have enough power and resources as if they were single computer. A cluster is needed.

A cluster, is a group of computers, pools the resources of many machines together, giving us the ability to use all cumulative resources as if they were a single computer A group of machines alone is not powerful, you need framework to coordinate work across them. Spark does just that, managing and coordinating the execution of tasks on data across a cluster of computers.

A group of machines alone is not powerful, you need framework to coordinate work across them. Spark does just that, managing and coordinating the execution of tasks on data across a cluster of computers



Spark applications consist of 2 processes:

1. Driver Process:

Runs the main() function

* 1. Managing information about spark application
  2. Responding to users program or input
  3. Analysing, distributing, and scheduling work across the executors

1. Executor Process
   1. Executing code assigned to it by driver
   2. Reporting the state of computation on that executor back to driver node

SparkSession You control your spark application through a driver process called SparkSession Default variable created when executed code, entry point

Spark O/P: <pyspark.sql.session.SparkSession>

**Spark DataFrame:**

A dataframe is most common Structured API and simply represents a table of data with rows and columns This is also distributed

**Transformations**

Business logic on data Creates logical plan and physical plan for execution Only executed after Action is made

divsBy2 = myRange.where("number % 2 == 0")

divsBy2.count() -- Action

**Databricks**

Software that supports Apache spark environment No need to install JVM, packages, configure env path on 1000nds of machines, scaling and networking. Databricks takes care of all these. Only focus on writing code.