**DS USING PYTHON LAB**

**EXPERIMENT: 9 A**

**AIM:** A case study on Power BI and Apache Spark

**THEORY:**

**APACHE SPARK**

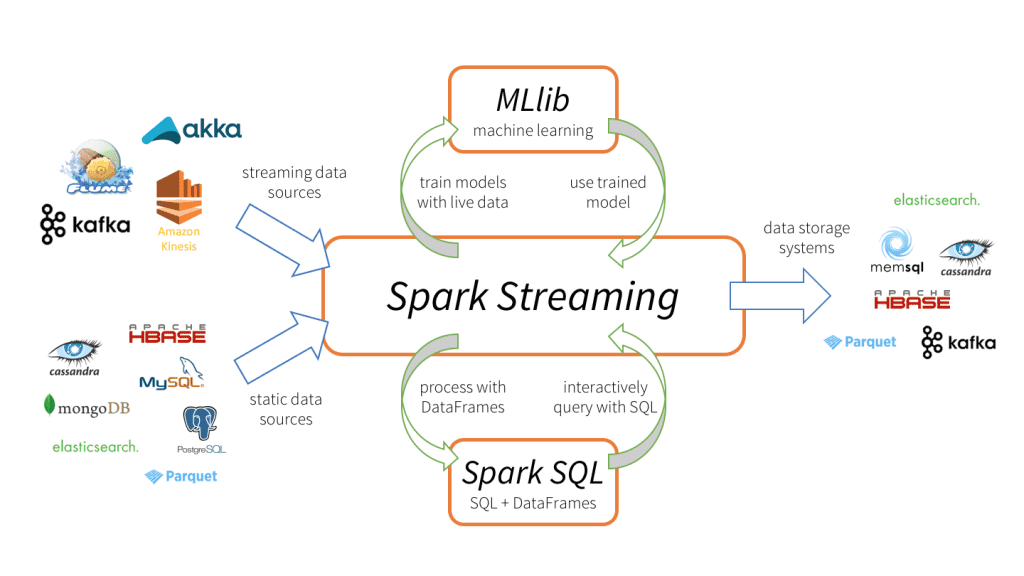
**i. Introduction to Apache Spark**

Apache Spark is an actively developed and unified computing engine and a set of libraries. It is used for parallel data processing on computer clusters and has become a standard tool for any Developer or Data Scientist interested in Big Data.

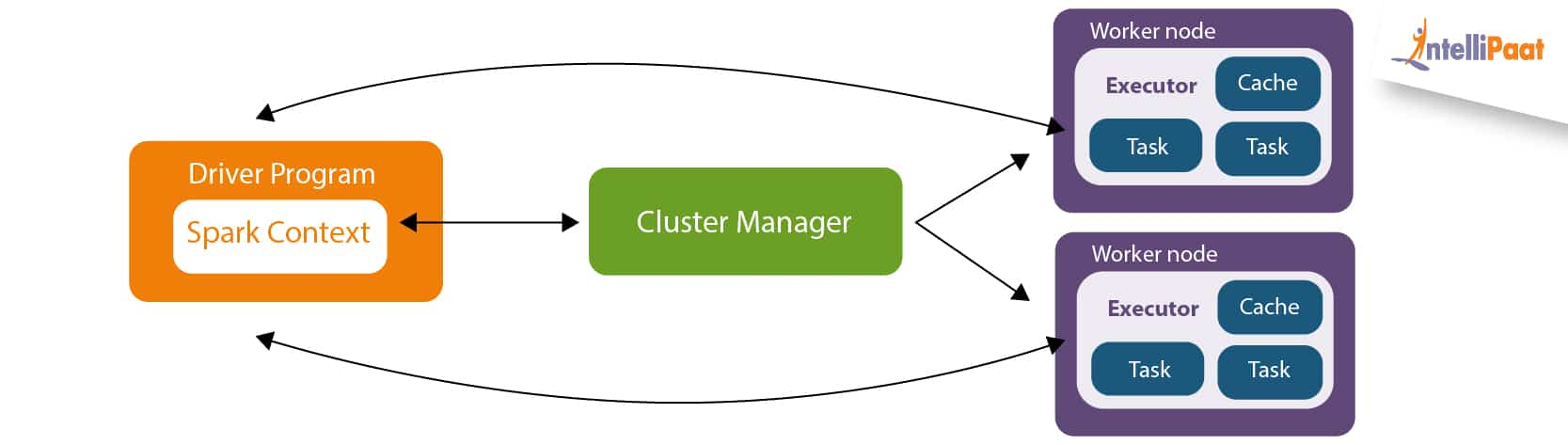
Spark supports multiple widely-used programming languages like Java, Python, R, and Scala. It includes libraries for a diverse range of tasks such as SQL, Streaming, Machine Learning, etc. It runs anywhere from a laptop to a cluster of thousands of servers making it a beginner-friendly system with a steep learning curve and users can scale up to big data processing or to an incredibly large scale.

Apache Spark has following features.

* Speed − Spark helps to run an application in Hadoop cluster, up to 100 times faster in memory, and 10 times faster when running on disk. This is possible by reducing number of read/write operations to disk. It stores the intermediate processing data in memory.
* Supports multiple languages − Spark provides built-in APIs in Java, Scala, or Python. Therefore, you can write applications in different languages. Spark comes up with 80 high-level operators for interactive querying.
* Advanced Analytics − Spark not only supports ‘Map’ and ‘reduce’. It also supports SQL queries, Streaming data, Machine learning (ML), and Graph algorithms.



**ii. Working on the Apache Spark Architecture**

****

Driver Program in the Apache Spark architecture calls the main program of an application and creates SparkContext. A SparkContext consists of all the basic functionalities. Spark Driver contains various other components such as DAG Scheduler, Task Scheduler, Backend Scheduler, and Block Manager, which are responsible for translating the user-written code into jobs that are actually executed on the cluster.

Spark Driver and SparkContext collectively watch over the job execution within the cluster. Spark Driver works with the Cluster Manager to manage various other jobs. The cluster Manager does the resource-allocating work. And then, the job is split into multiple smaller tasks which are further distributed to worker nodes.

Whenever an RDD is created in the SparkContext, it can be distributed across many worker nodes and can also be cached there.

Worker nodes execute the tasks assigned by the Cluster Manager and return it back to the Spark Context.

An executor is responsible for the execution of these tasks. The lifetime of executors is the same as that of the Spark Application. If we want to increase the performance of the system, we can increase the number of workers so that the jobs can be divided into more logical portions.

**iii. Architecture of Apache Spark**

The Spark follows the master-slave architecture. Its cluster consists of a single master and multiple slaves. The Spark architecture depends upon two abstractions:

* Resilient Distributed Dataset (RDD)
* Directed Acyclic Graph (DAG)

Resilient Distributed Datasets (RDD)

The Resilient Distributed Datasets are the data items that can be stored in memory on worker nodes. Here,

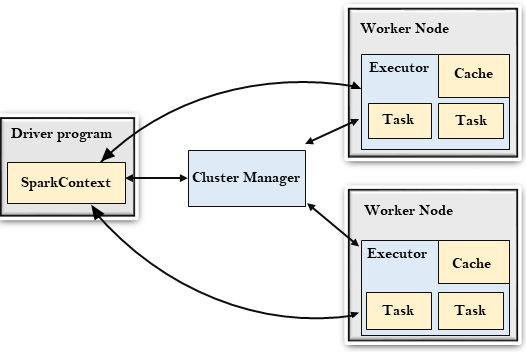
Resilient: Restore the data on failure.

Distributed: Data is distributed among different nodes.

Dataset: Group of data.

Directed Acyclic Graph (DAG)

A directed Acyclic Graph is a finite direct graph that performs a sequence of computations on data. Each node is an RDD partition, and the edge is a transformation on top of data. Here, the graph refers to the navigation whereas directed and acyclic refers to how it is done..



* Driver Program

The Driver Program is a process that runs the main() function of the application and creates the SparkContext object. The purpose of SparkContext is to coordinate the spark applications, running as independent sets of processes on a cluster.

To run on a cluster, the SparkContext connects to a different type of cluster managers and then performs the following tasks: -

* It acquires executors on nodes in the cluster.
* Then, it sends your application code to the executors. Here, the application code can be defined by JAR or Python files passed to SparkContext.
* At last, SparkContext sends tasks to the executors to run.
* Cluster Manager

The role of the cluster manager is to allocate resources across applications. The Spark is capable enough of running on a large number of clusters.

It consists of various types of cluster managers such as Hadoop YARN, Apache Mesos, and Standalone Scheduler.

Here, the Standalone Scheduler is a standalone spark cluster manager that facilitates to the installation of Spark on an empty set of machines.

* Worker Node

The worker node is a slave node whose role is to run the application code in the cluster.

* Executor

An executor is a process launched for an application on a worker node. It runs tasks and keeps data in memory or disk storage across them. It read and writes data to external sources. Every application contains its executor.

* Task

A unit of work will be sent to one executor.

**v. Use cases of Apache Spark**

Spark is a general-purpose distributed processing system used for big data workloads. It has been deployed in every type of big data use case to detect patterns, and provide real-time insight. Example use cases include:

* Financial Services:Spark is used in banking to predict customer churn, and recommend new financial products. In investment banking, Spark is used to analyze stock prices to predict future trends.
* Healthcare:Spark is used to build comprehensive patient care, by making data available to front-line health workers for every patient interaction. Spark can also be used to predict/recommend patient treatment.
* Manufacturing:Spark is used to eliminate downtime of internet-connected equipment, by recommending when to do preventive maintenance.
* Retail:Spark is used to attract, and keep customers through personalized services and offers.

**Power BI**

**Introduction:**

Power BI is a Data Visualization and Business Intelligence tool that converts data from different data sources to interactive dashboards and BI reports. Power BI suite provides multiple software, connector, and services - Power BI desktop, Power BI service based on Saas, and mobile Power BI apps available for different platforms. This set of services is used by business users to consume data and build BI reports.

The power BI desktop app is used to create reports, while Power BI Services (Software as a Service - SaaS) is used to publish the reports, and Power BI mobile app is used to view the reports and dashboards.



**Architecture:**

* Power BI includes the following components −
* Power BI Desktop − This is used to create reports and data visualizations on the dataset.
* Power BI Gateway − You can use Power BI on-premises gateway to keep your data fresh by connecting to your on-premises data sources without the need to move the data. It allows you to query large datasets and benefit from existing investments.
* Power BI Mobile Apps − Using Power BI mobile apps, you can stay connected to their data from anywhere. Power BI apps are available for Windows, iOS, and Android platforms.
* Power BI Service − This is a cloud service and is used to publish Power BI reports and data visualizations.

**Supported Data Sources**

Power BI supports a large range of data sources. You can click Get data and it shows you all the available data connections. It allows you to connect to different flat files, SQL databases, and Azure cloud, or even web platforms such as Facebook, Google Analytics, and Salesforce objects. It also includes an ODBC connection to connect to other ODBC data sources, which are not listed.Following are the available data sources in Power BI −

* Flat Files
* SQL Database
* OData Feed
* Blank Query
* Azure Cloud platform
* Online Services
* Blank Query
* Other data sources such as Hadoop, Exchange, or Active Directory

**PowerBI Dashboard**

Power Bl Desktop is an enlargement tool used to generate dashboards and reports. Power Bl applications will be accessed by other users through desktop and mobile devices.

There are Power BI components are 5 main components released in the market.

* Power Query: It can be used to search, access, and transform public and/ or internal data sources.
* Power Pivot: It is used for data modeling for in-memory analytics.
* Power View: This component allows you to visualize, analyze, and display data.
* Power Map: It brings data to life with interactive geographical visualization.
* Power BI Service: It allows you to share data views and workbooks, which you can refresh from on-premises and cloud-based data sources.
* Power BI Q&A: It allows you to ask questions and get immediate answers using a natural language query.
* Data Management Gateway: It offers periodic data refreshers, view data feeds, expose tables.
* Data Catalog: This component allows the user to discover and reuse queries using the Data Catalog. Metadata can be facilitated for search functionality.

**PowerBI chart types and visual lists**

* Line Charts
* Bar Charts
* Area Charts
* Columns Charts
* Cards
* Combo Charts
* Pie Charts
* Doughnut Charts
* Decomposition Tree
* Funnel Charts
* KPIs
* Gauge Charts
* Waterfall Charts
* Maps
* Matrix

**CONCLUSION:** In this experiment, we have performed a case study on Apache Spark and Power BI