



**DataBricks Spark – Azure DataBricks**

# Contents

1. Course Introduction
2. Why Apache Spark?
3. Spark Cluster Managers
4. Introduction to DataBricks
5. DataBricks Components
6. RDD
7. Transformations and Actions in RDD
8. DataFrame
9. Transformation and Actions in Dataframe
10. Working with DataFrames
11. SparkSQL
12. File systems and sources supported by Spark
13. DeltaLake
14. Spark Applications
15. Batch ETL using Spark
16. Introduction to Kafka
17. Real-Time ETL and Event partition using Kafka and Spark
18. Spark MLLib and Machine Learning using Spark



# Course Introduction

# Course Introduction

- About this Course
- Course Logistics
- Course Agreements
- About you !
- About the Instructor
- General Instructions for Exercises

## In this course you will learn,

- What is Apache Spark?
- Starting the Spark Shell
- Using the Spark Shell
- Getting Started with RDD
- Getting Started with Datasets and DataFrames
- DataFrame Operations
- Introduction to Data Bricks
- Create your own Databricks workspace
- Auto scaling and auto-terminating behaviour
- Create a notebook inside your home folder in Databricks
- Understand the fundamentals of Apache Spark notebook
- Create and attach to a Spark cluster
- Adding Libraries in DataBricks
- How Data Bricks is different than tradition Apache Spark clusters
- Benefits of using Data Bricks
- Integration of Databricks with other Azure tools
- Security in Data Bricks
- Users and Active directory association
- Identify the types of tasks that are well suited to the unified analytics engine Apache Spark.

## In this course you will learn, [contd]

- HDFS
- DBFS
- Object Stores
- RDBMS
- Hive
- Kafka
- RDD Overview
- RDD Data Sources
- Creating and Saving RDDs
- RDD Operations
- Writing and Passing
- Transformation Functions
- Transformation Execution
- RDD lazy Evaluation
- RDD Partitions and Coalesce

## In this course you will learn, [contd]

- Creating DataFrames from Data Sources
- Saving DataFrames to Data Sources
- DataFrame Schemas
- Eager and Lazy Execution
- Analysing Data with DataFrame Queries
- Querying DataFrames Using
- Column Expressions
- Grouping and Aggregation Queries
- Joining DataFrames
- Catalyst Execution Plan
- Querying Tables in Spark Using SQL
- Querying Files and Views
- The Catalog API
- Comparing Spark SQL, Apache Impala, and Apache Hive-on-Spark
- Apache Spark Applications
- Writing a Spark Application in Cell
- Running an Application
- DataBricks Spark Application Web UI
- Configuring Application Properties
- Log aggregations in Spark

## In this course you will learn, [contd]

- Connecting RDBMS to Spark
- Ingesting Data to Spark DataFrames
- Business flow using a use case
- Writing Data to Object Store
- Connecting Spark with Kafka
- Writing Spark Streaming Application
- Spark Structured Streaming
- Aggregations on Spark Streaming
- Loading the results in ObjectStore
- Data Processing
- Common Apache Spark Use Cases
- Iterative Algorithms in Apache Spark
- Introduction to Machine Learning
- Spark MLlib
- Clustering using KMeans in Spark
- Classification using Regression



### Course Logistics

- Course start and end times
- Break
- Lunch
- Ask Questions

## Course Introduction

### Course Agreements

- Mobile should be on silent mode
- Raise hand and ask question as soon as you have it
- Only one conversation in room at a time during the training
- Keep us informed if getting late
- Have proper sleep during training days so that you don't sleep during training hours.

### About You

- Name
- Company Name
- Role
- Total Experience
- Any Experience with SQL
- Any Experience with Big Data Technologies

### About the Instructor

- Industry Experience
- Experience with various Technologies
- Projects Experience
- Trainings Imparted
- Certifications

## World's Leading Digital Talent Transformation Company



130,000+ Professionals  
Trained world-wide



Delivered Training in  
over 45 Countries



Over 7000 Industry  
Veterans as Instructors



450+ Customers

## Authorized Training Partner For



**Red Hat**



Silver  
**Microsoft  
Partner**



WORLD  
HRD  
CONGRESS

Cognixia is awarded as  
Training Company of the Year, 2018



Asian Training &  
Leadership Award, Dubai



ISO 9001:2015 Certified  
Quality Management System

ISO 9001:2015 Certified  
Quality Management System



ISO/IEC 27001:2013 Certified  
Information Security Management System

## Centres of Excellence



**Machine  
Learning & AI**



**Microservices**



**Robotic Process  
Automation**



**Internet of  
Things**



**Big Data  
Analytics**



**DevOps**



**Cloud  
Computing**



**Cyber  
Security**



**BI  
Technology**



**Professional  
Development**

### **General Instruction for Exercises**

- Every participants have to sign up free databrick cloud environment for the practise
- Instructor will provide a Kafka and Hadoop Environment for the practice whenever required
- Instructor will be providing S3 and AZDL access principals while training.

# Contents

1. **Course Introduction**
2. Why Apache Spark?
3. Spark Cluster Managers
4. Introduction to DataBricks
5. DataBricks Components
6. RDD
7. Transformations and Actions in RDD
8. DataFrame
9. Transformation and Actions in Dataframe
10. Working with DataFrames
11. SparkSQL
12. File systems and sources supported by Spark
13. DeltaLake
14. Spark Applications
15. Batch ETL using Spark
16. Introduction to Kafka
17. Real-Time ETL and Event partition using Kafka and Spark
18. Spark MLLib and Machine Learning using Spark





# Why Apache Spark

# Why Apache Spark?

Data are getting created without bounds :

- Financial Transactions
- Sensor Networks
- Server Logs
- e-Mails and Text Messages
- Social Media
- Machine Feeds

And we are generating data faster than ever

- Automation
- Faster Internet Connectivity
- User-Generated Contents
- IoT
- Bots

## Why Apache Spark?

Twitter processes more than 500+ million tweets per day

Facebook users generate 7 billion comments and “Likes”

YouTube gets 30 Million users per day

5 Billion videos are watched in Youtube a day

Every Minute Amazon makes \$283,000 sales

Google gets 3.5 Million searches every minute

400,000 new tweets every minute

# Why Apache Spark?

The data has many valuable applications from which we can extract values

- ☐ Marketing Analysis
- ☐ Product Recommendations
- ☐ Demand Forecasting
- ☐ Fraud Detection
- ☐ Predictive Models and many more

# Why Apache Spark?

## **Data Processing - Scalability**

How can we process all that information?

There are actually two problems

- Large/scale data storage
- Large/scale data analysis

Reading 3TB data from a single Disk almost takes four hours. We cannot process the data until we read it and limited by the speed of the disk.

# Why Apache Spark?

## Spark Goal

Provide distributed memory abstractions for clusters to support apps with working sets

Retain the attractive properties of MapReduce:

- Fault tolerance (for crashes & stragglers)

- Data locality

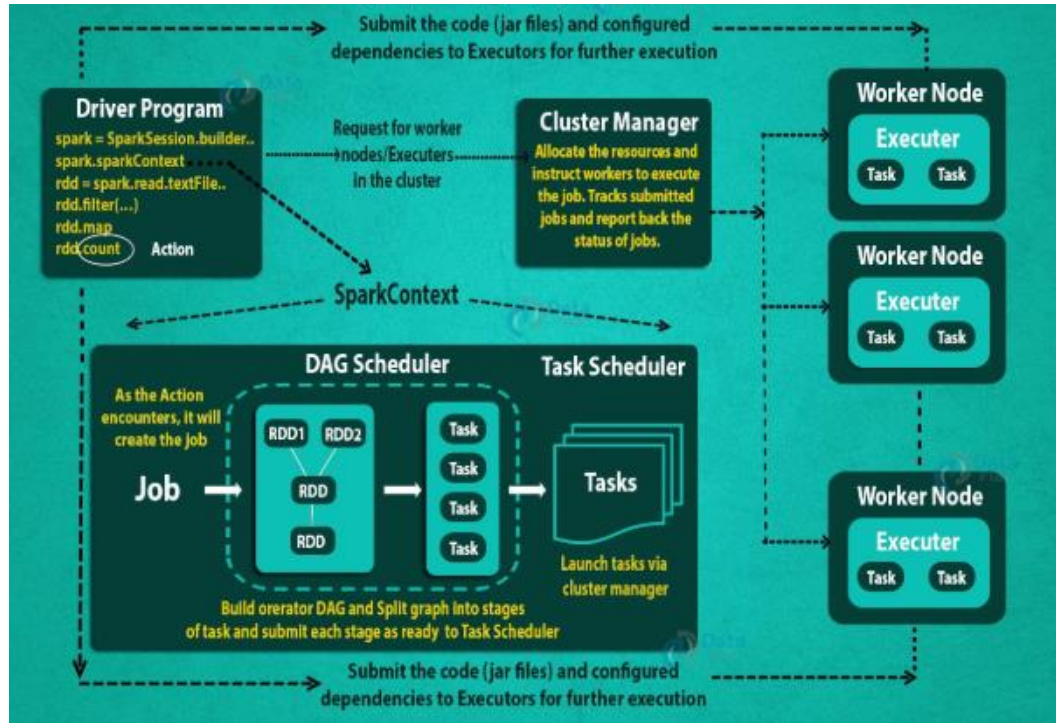
- Scalability

## Apache Spark

- Apache Spark is a fast and general purpose engine for large scale data processing framework.
- Spark is Written in Scala
  - Functional Programming language that runs in JVM
  - Spark runs on Hadoop, Apache Mesos, Kubernetes, standalone, or in the cloud. It can access diverse data sources
  - Expressive computing system, not limited to map-reduce model
  - Facilitate system memory
    - avoid saving intermediate results to disk
    - cache data for repetitive queries (e.g. for machine learning)

# Why Apache Spark?

## Spark Execution Mode





## Why Apache Spark?

### Spark - In-Memory Computing

- In in-memory computation, the data is kept in random access memory(RAM) instead of some slow disk drives and is processed in parallel.
- The Data will be partitioned and processed across the different servers in the cluster.
- The Lightning speed term of spark got derived because spark takes the data to In-Memory for Processing

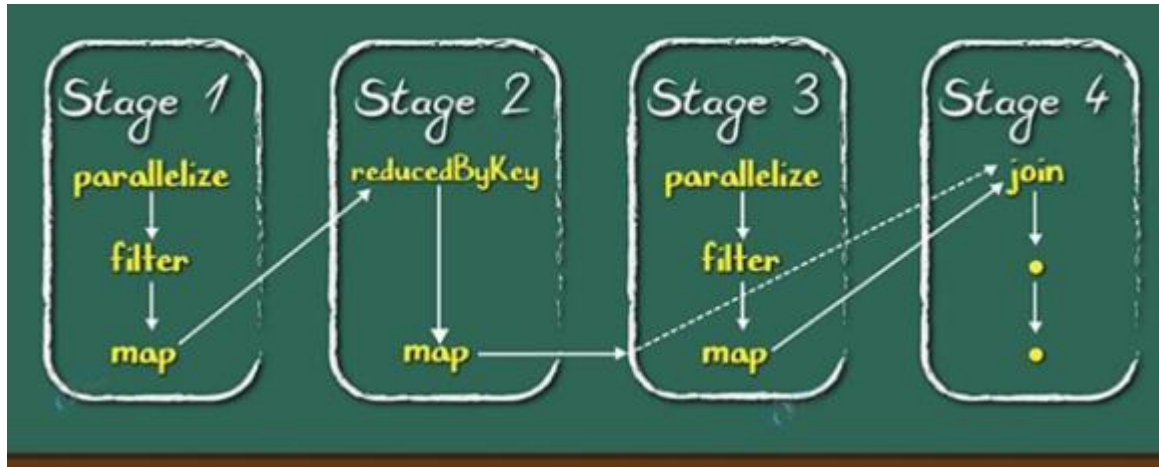
### Spark – Lazy Evaluation

- As the name itself indicates its definition, **lazy evaluation** in Spark means that the execution will not start until an action is triggered.
- Actions are nothing but, the final result. Consider we have a set of data from the source which is coming with sales details from multiple city.
  - First step would be filtering the data which belongs to city New Delhi. This is considered as Transformation.
  - Next the company wants to understand the overall count of sales from New Delhi. This is considered as Action because it gets an aggregated results.
- **Transformations** are lazy in nature meaning when we call some operation it will start its execution with the help of DAG scheduler

# Why Apache Spark?

## Spark – DAG – Directed Acyclic Graph

- **DAG** a finite direct graph with no directed cycles. There are finitely many *vertices* and *edges*, where each edge directed from one vertex to another.



# Why Apache Spark?

## Spark Components

- ☐ RDD - Resilient Distributed Datasets
- ☐ Spark Dataframes
- ☐ Spark Datasets
- ☐ Spark MLIB
- ☐ Spark Streaming

# Why Apache Spark?

## Spark RDD

- Resilient Distributed Datasets
- Partitioned collection of records
- Spread across the cluster
- Read-only -Immutable
- Caching dataset in memory
  - different storage levels available
  - fallback to disk possible

## Spark DataFrames

- DataFrames and Datasets are the primary representation of data in Spark
- DataFrames represent structured data in a tabular form
  - DataFrames model data similar to tables in an RDBMS
  - DataFrames consist of a collection of loosely typed Row objects
  - Rows are organized into columns described by a schema

## Spark Datasets

- Datasets represent data as a collection of objects of a specified type  
Datasets are strongly-typed—type checking is enforced at compile time rather than run time.

An associated schema maps object properties to a table-like structure of rows and columns

Datasets are only defined in Scala and Java

DataFrame is an alias for Dataset[Row]—Datasets containing Row Objects

# Contents

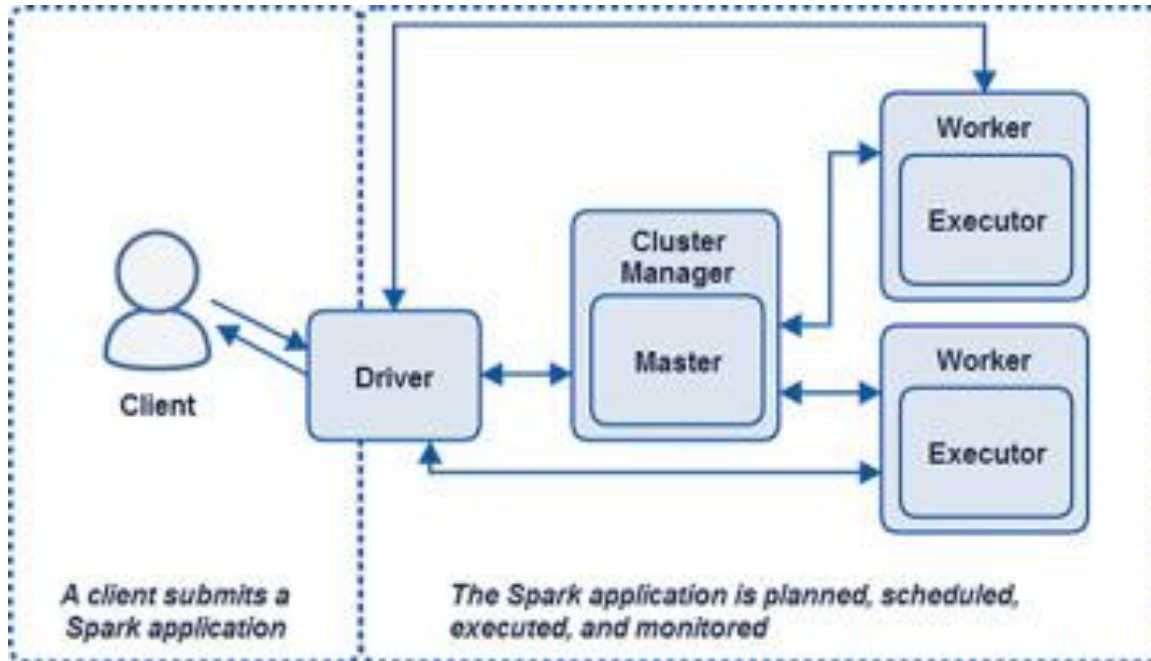
1. **Course Introduction**
2. **Why Apache Spark?**
3. Spark Cluster Managers
4. Introduction to DataBricks
5. DataBricks Components
6. RDD
7. Transformations and Actions in RDD
8. DataFrame
9. Transformation and Actions in Dataframe
10. Working with DataFrames
11. SparkSQL
12. File systems and sources supported by Spark
13. DeltaLake
14. Spark Applications
15. Batch ETL using Spark
16. Introduction to Kafka
17. Real-Time ETL and Event partition using Kafka and Spark
18. Spark MLLib and Machine Learning using Spark



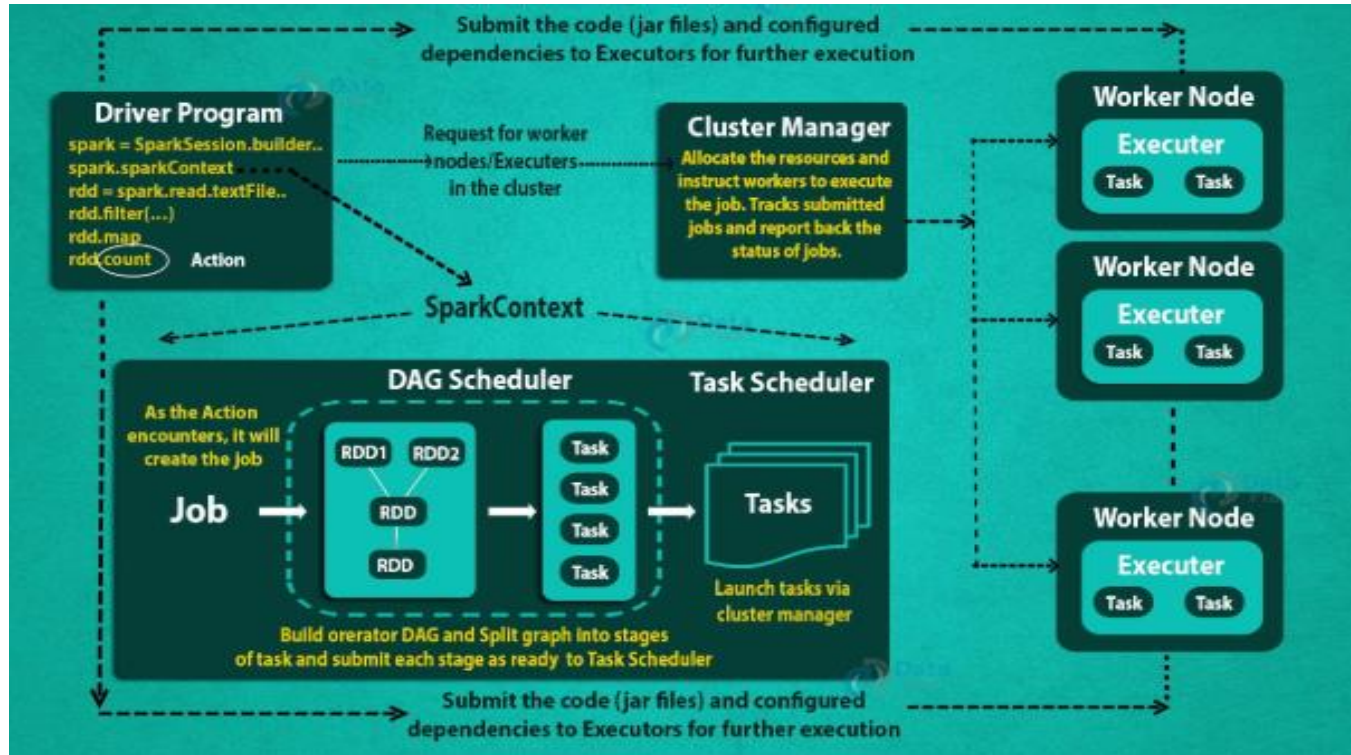


# Spark Cluster Managers

## Standalone Cluster Manager



## Standalone Cluster Manager



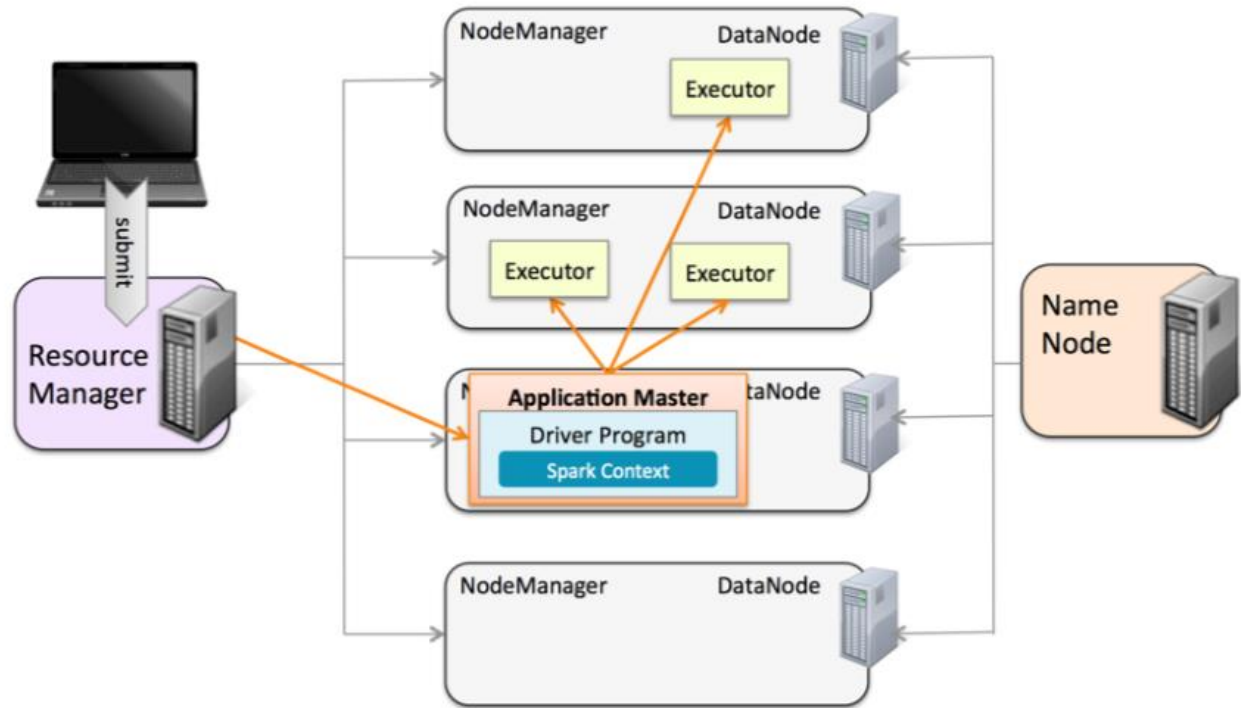
### **Standalone Cluster Manager**

- 1. Client submits the Spark job to the Spark Driver**
- 2. Spark Driver initiates the Spark Context object for the submitted job.**
- 3. Spark Context will frame the spark tasks based on the DAG task scheduler.**
- 4. Spark Driver will request the Cluster Manager for the resource(executors) for the job execution**
- 5. Spark Driver Configure the required extra JAR files for the worker nodes to perform the tasks**

### Standalone Cluster Manager

6. Tasks will be launched on the Executors provided by the cluster Manager
7. The result set from each executor will be send to the Spark Driver
8. Once the job is done, cluster manager will the executors.
9. Spark executors will not be killed until the life time of application.
10. Finally driver will send the result dataset to the Spark client.

## YARN – Yet Another Resource Negotiator



### **YARN – Yet Another Resource Negotiator**

MRv2-MapReduce Version 2/ YARN-Yet Another Resource Negotiator  
Contains three daemons- Resource Manager, Node Manager and Application Master.

#### **Resource Manager-**

- Responsible for allocation of resources in a cluster.
- Runs on Master Node.
- Monitors Node Managers and Application Masters

#### **Node Manager-**

- Communicates with the Resource Manager.
- Runs on worker node.
- Per-machine framework agent who is responsible for containers, monitoring their resource usage and reporting the same to the ResourceManager.

## **YARN – Yet Another Resource Negotiator**

### **Container-**

- Created by Resource Manager upon request.
- Allocates certain amount of resources (CPU,memory) on worker Node.
- Applications can run on one or more containers.

### **Application Master-**

- One Application Master per application.
- Coordinates the running task scheduled and managed by RM.
- Runs in a container.



## Spark Cluster Managers

- Spark will also run in
  - Apache Mesos which is based on DC OS
  - Docker
- Databricks works based on the Standalone cluster manager based
- Whereas Hadoop based spark works on YARN.

# Contents

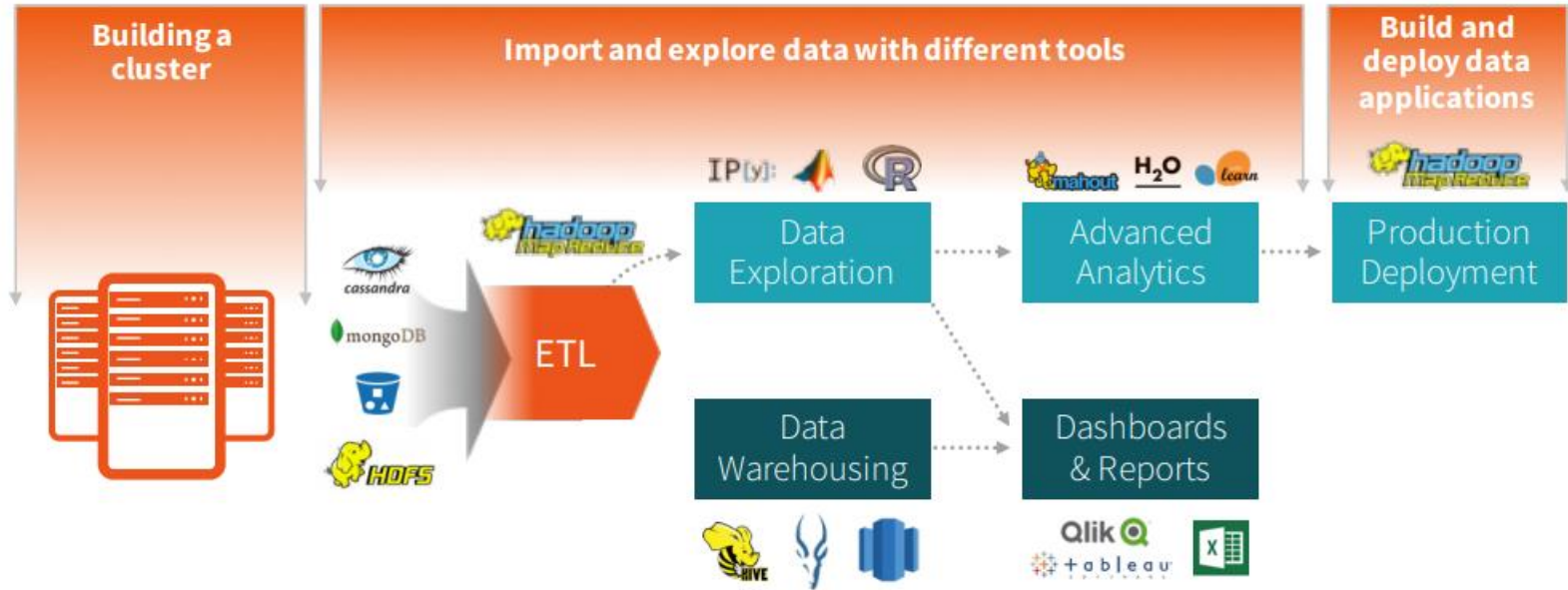
1. **Course Introduction**
2. **Why Apache Spark?**
3. **Spark Cluster Managers**
4. Introduction to DataBricks
5. DataBricks Components
6. File systems and sources supported by Spark
7. DeltaLake
8. RDD
9. Transformations and Actions in RDD
10. DataFrame
11. Transformation and Actions in Dataframe
12. Working with DataFrames
13. SparkSQL
14. Spark Applications
15. Batch ETL using Spark
16. Introduction to Kafka
17. Real-Time ETL and Event partition using Kafka and Spark
18. Spark MLLib and Machine Learning using Spark



# Introduction to DataBricks

# Introduction to DataBricks

## Challenges of Modern scale Data Engineering and Data Science

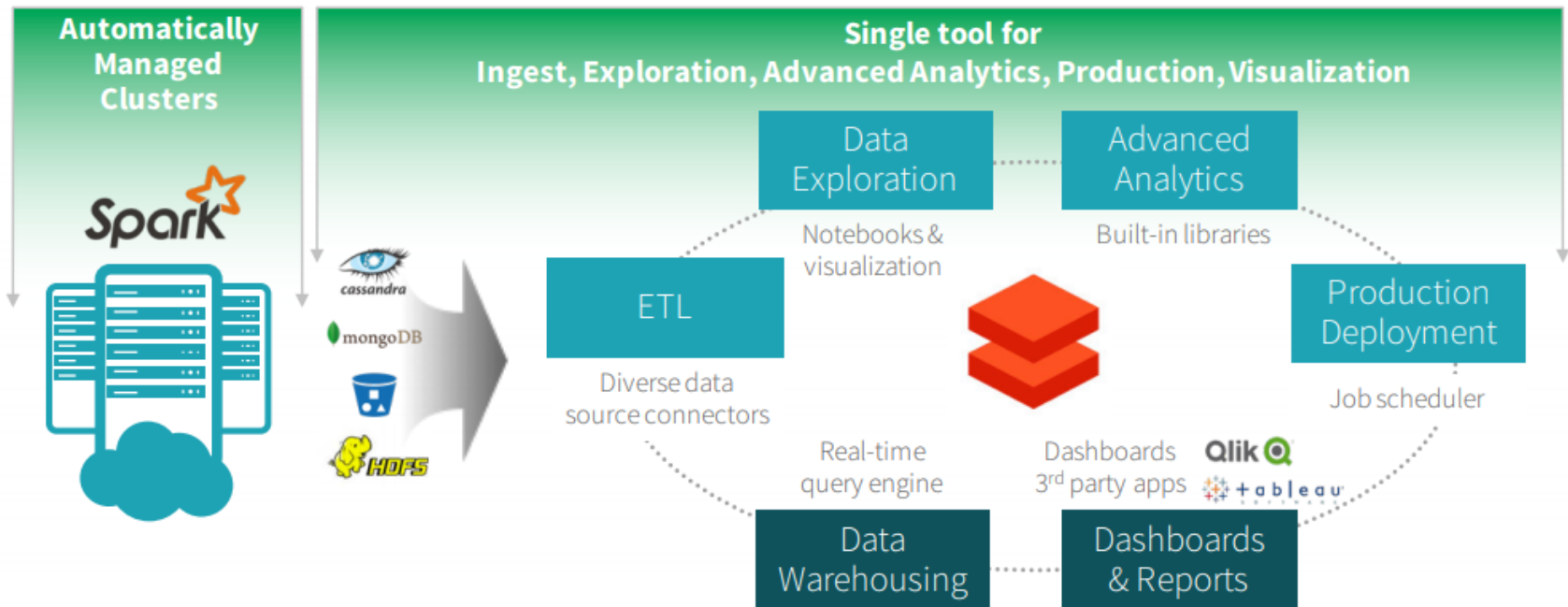


## Challenges of Modern scale Data Engineering and Data Science

1. Setting up the infrastructure for Distributed Data Storage, Distributed Data Processing, Data Visualization components, Data Science Development and Data Pipelines
2. Highly Skilled resources for all the above because all are separate components
3. Complex integrations
4. Managing the pipelines
5. Handling the failures

# Introduction to DataBricks

## Unified End to End Data Bricks Solution

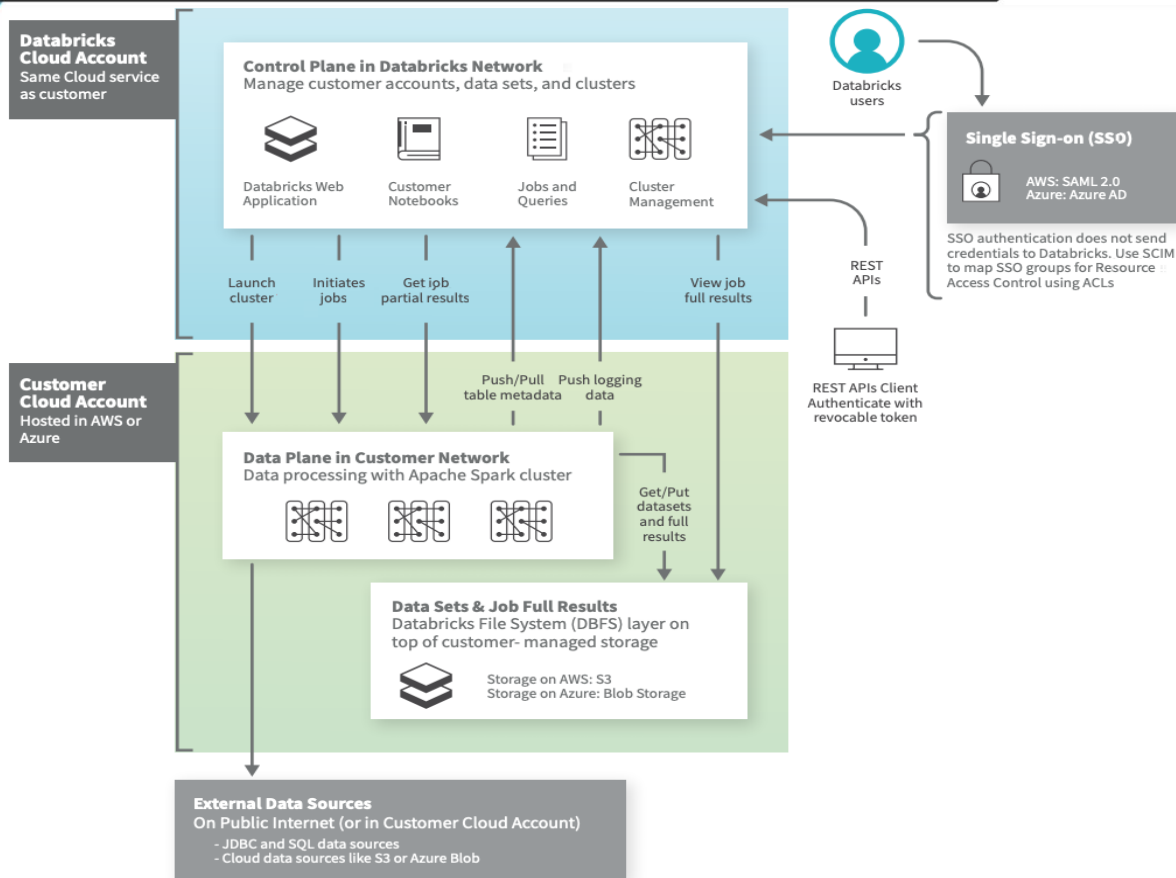


## Unified End to End Data Bricks Solution

DataBricks provides End to End solution for Data Engineers and Data Scientist for their below requirements by with an End to End Spark as a Service model

- Batch ETL pipelines
- Real-Time Data Pipelines
- Connectors for Easy Integrations
- Job Scheduling capabilities
- Auto Scaling of Cluster Size
- Integrated Data Visualization components
- Integrated Notebook for development
- Real-Time Query engine which stores data in cloud storage and DeltaLake
- Built-in Libraries for Data Analytics

# Introduction to DataBricks







- DataBricks is a cloud based Apache Spark Cluster Service
- DataBricks is available in AWS, Azure and GCP
- Offers Scalable spark clusters with more control and ease of management
- Developed by the same group of developers who created Apache Spark
- Offers access of all the spark components and libraries
- Provides connectivity and integration with most of the data sources, message queues and Cloud components


- Architecture based on Notebooks and folders
- Has a Spark cluster Manager for the resource allocations
- DataBricks got Job Manager and Scheduler
- Also auto scale in and scale down of cluster is possible
- User management is available using AWS, Azure and GCP based user management components
- Integrated with Strong Data Visualization components with facilities to export reports and Dashboards


- Notebooks can be executed in
  - Python
  - Scala
  - R and
  - SQL
- Notebooks can be shared and deployed in different cluster
- Libraries can be imported to the cluster and can be called in the notebooks
- Notebook is best suited for those that have very little or no experience with Spark


# Introduction to DataBricks


  
databricks


  
Home


  
Workspace


  
Recents

  
Data


  
Clusters

  
Jobs

  
Search

Upgrade ? 


## Welcome to databricks™



### Explore the Quickstart Tutorial


Spin up a cluster, run queries on preloaded data, and display results in 5 minutes.

Drop files or [click to browse](#)



### Import & Explore Data







Quickly import data, preview its schema, create a table, and query it in a notebook.








### Create a Blank Notebook

Create a notebook to start querying, visualizing, and modeling your data.

#### Common Tasks

-  New Notebook
-  Create Table
-  New Cluster
-  New Job
-  New MLflow Experiment
-  Import Library

#### Recents

-  2020-04-03 - DBFS Example
-  Delta
-  XML
-  Test\_NB
-  Quickstart Notebook

#### What's new in v3.16

[View latest release notes](#)

# Contents

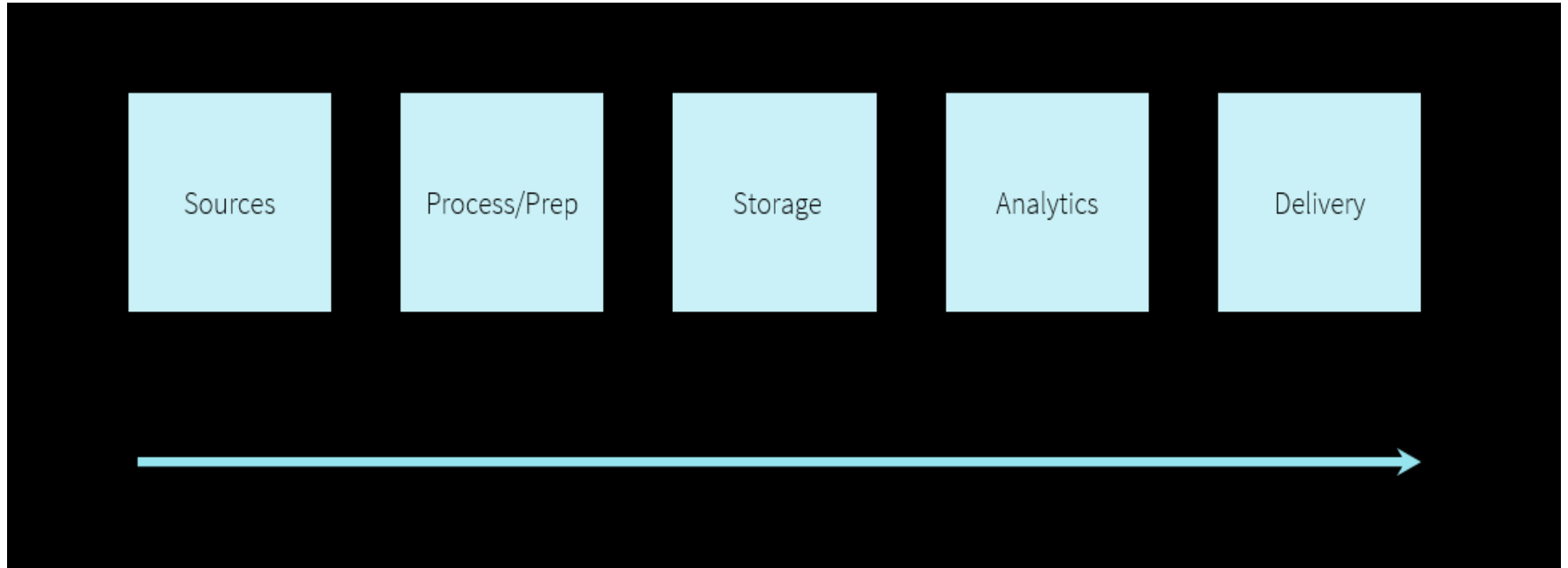
1. **Course Introduction**
2. **Why Apache Spark?**
3. **Spark Cluster Managers**
4. **Introduction to DataBricks**
5. DataBricks Components
6. File systems and sources supported by Spark
7. DeltaLake
8. RDD
9. Transformations and Actions in RDD
10. DataFrame
11. Transformation and Actions in Dataframe
12. Working with DataFrames
13. SparkSQL
14. Spark Applications
15. Batch ETL using Spark
16. Introduction to Kafka
17. Real-Time ETL and Event partition using Kafka and Spark
18. Spark MLLib and Machine Learning using Spark



# DataBricks Components

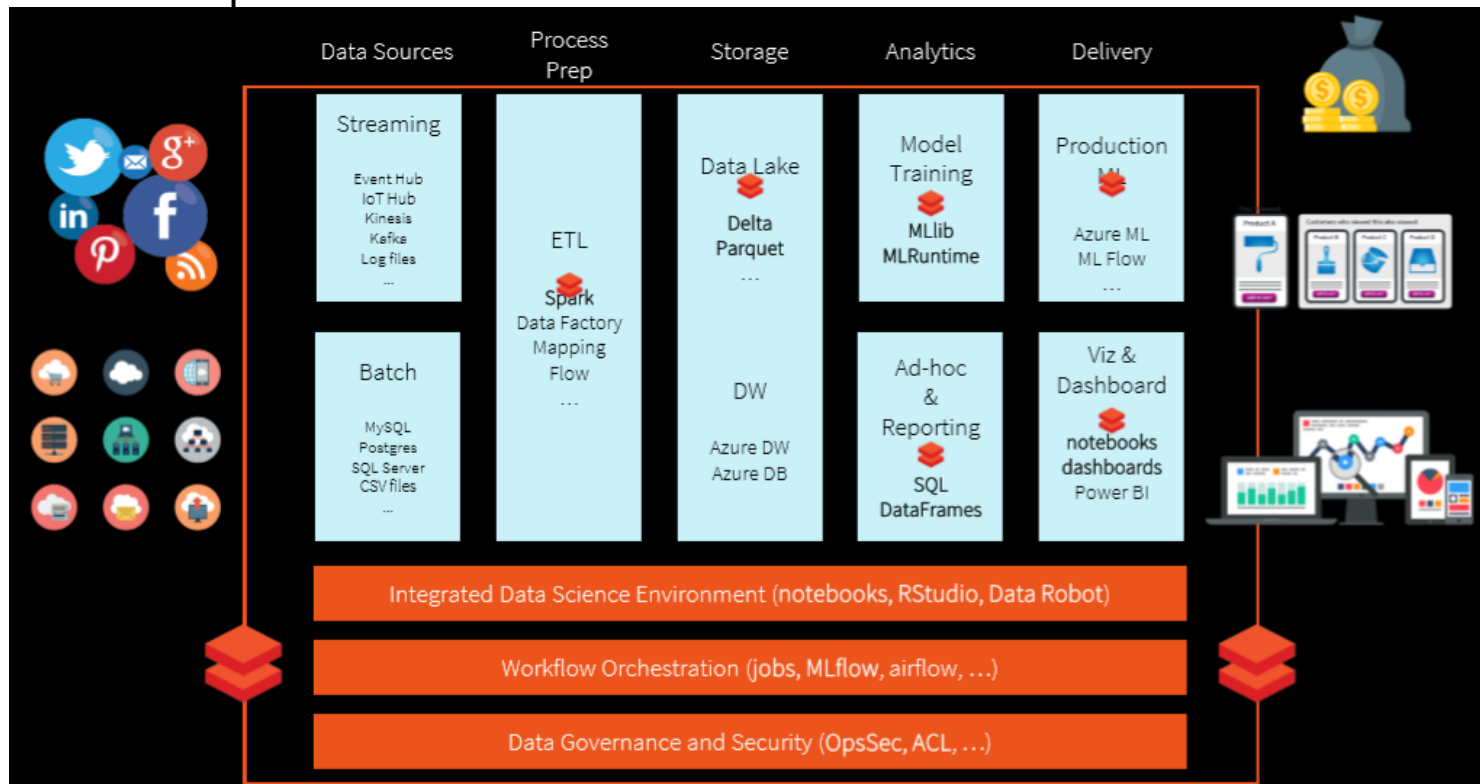
# DataBricks Components

## Requirement of a Data Platform



# DataBricks Components

## DataBricks Components





# DataBricks Components

## Create your Own DataBricks Workspace

1. DataBricks provides Platform Free Trial based on Azure and AWS.
2. DataBricks also provides Community Edition which is fully resources. As a user, we are not required to link the community account with any Cloud providers
3. Community edition is good for learning and making good hands-on based on DataBricks Usage, Spark and DeltaLake
4. DataBricks with Azure provides features like REST API, User Management using LDAP and AD, Job Schedulers which are not available in Community Edition. In this training we will be using community Edition.

# DataBricks Components

## Create your Own DataBricks Workspace

1. To Sign up Community Edition of DataBricks. Please visit the below URL

[https://databricks.com/try-databricks?\\_ga=2.231253720.636689393.1585744376-1730487577.1573900075](https://databricks.com/try-databricks?_ga=2.231253720.636689393.1585744376-1730487577.1573900075)

2. After you provide your Details Select the community Edition option

### DATABRICKS PLATFORM – FREE TRIAL

For businesses looking for a zero-management cloud platform built around Apache Spark

- Unlimited clusters that can scale to any size
- Job scheduler to execute jobs for production pipelines
- Fully interactive notebook with collaboration, dashboards, REST APIs
- Advanced security, role-based access controls, and audit logs
- Single Sign On support
- Integration with BI tools such as Tableau, Qlik, and Looker
- 14-day full feature trial (excludes cloud charges)

GET STARTED

### COMMUNITY EDITION

For students and educational institutions just getting started with Apache Spark

- Single cluster limited to 6GB and no worker nodes
- Basic notebook without collaboration
- Limited to 3 max users
- Public environment to share your work

GET STARTED

# DataBricks Components

## Create your Own DataBricks Workspace

1. Community Edition provides
  1. Free Spark Cluster for Spark Learning Purpose
  2. DeltaLake
  3. External Libraries
  4. NoteBook
  5. DBFS
  6. Visualization Components

## DataBricks Components

### Exercise 1 – Creating your own Databricks Community Edition workspace

Refer the Exercise manual for step wise and screenshot based reference

## DataBricks Components

### Creating Cluster in DataBricks

- DataBricks committed to create spark clusters as per the request.
- These spark clusters are managed by DataBricks Cluster Manager service.
- Users can be assigned for this clusters
- When integrating with Azure, DataBricks makes use of Azure VMs to create the Cluster, which means the cluster VMs will be accessed from Azure.

## Creating Cluster in DataBricks

- DataBricks provides options to Auto Scale the cluster
  - Identifying the right number of executors required for a single job
  - Based upon significant trail and error, DataBricks determine the right numbers of Executors for the job
- Sub Optimal Resource utilization
  - Production Spark jobs typically have multiple Spark stages. Some stages might require huge compute resources compared to other stages.
  - Users provide a number of executors based on the stage that requires maximum resources. Having such a static size allocated to an entire Spark job with multiple stages results in suboptimal utilization of resources.

## DataBricks Components

### Creating Cluster in DataBricks

- DataBricks Introduces DataBricks Optimized AutoScaling
- The new optimized autoscaling service for compute resources allows clusters to scale up and down more aggressively in response to load and improves the utilization of cluster resources automatically without the need for any complex setup from users.
- Databricks' optimized autoscaling solves this problem by periodically reporting detailed statistics on idle executors and the location of intermediate files within the cluster. The Databricks service uses this information to more precisely target workers to scale down when utilization is low.

### Creating Cluster in DataBricks

- Since Databricks can precisely target workers for scale-down under low utilization, clusters can be resized much more aggressively in response to load. In particular, under low utilization, Databricks clusters can be scaled down aggressively *without* killing tasks or recomputing intermediate results.
- It can also Scale up Aggressively based upon the load
- Overall this helps on drastic cost reduction on the enterprise when compared to traditional environments



# DataBricks Components

## Creating Cluster in DataBricks

- DataBricks will price in Azure based on DBU(DataBricks Unit).

Azure Databricks Premium - Data Analytics								
Cluster Type	VM Type	VM Configuration	DBU#	DBU Price/hour*	Linux VM Cost/hour	DBU Cost	VM Cost	Total Cost / month
General Purpose	Standard_D8s_v3	32 GB Memory, 8 Cores, 1.50 DBU	1.5	\$0.550	\$0.384	\$594.00	\$276.48	\$870.48
General Purpose	Standard_DS4_v2	28 GB Memory, 8 Cores, 1.50 DBU	1.5	\$0.550	\$0.458	\$594.00	\$329.76	\$923.76
General Purpose	Standard_D8_v3	32 GB Memory, 8 Cores, 1.50 DBU	1.5	\$0.550	\$0.384	\$594.00	\$276.48	\$870.48
Memory Optimised	Standard_DS13_v2	56 GB Memory, 8 Cores, 2.00 DBU	2	\$0.550	\$0.598	\$792.00	\$430.56	\$1,222.56
Memory Optimised	Standard_D13_v2	56 GB Memory, 8 Cores, 2.00 DBU	2	\$0.550	\$0.598	\$792.00	\$430.56	\$1,222.56
Memory Optimised	Standard_E8s_v3	64 GB Memory, 8 Cores, 2.00 DBU	2	\$0.550	\$0.532	\$792.00	\$383.04	\$1,175.04
Storage Optimised	Standard_L8s	64 GB Memory, 8 Cores, 2.00 DBU	2	\$0.550	\$0.686	\$792.00	\$493.92	\$1,285.92
Compute Optimised	Standard_F8s_v2 (beta)	16 GB Memory, 8 Cores, 2.00 DBU	2	\$0.550	\$0.338	\$792.00	\$243.36	\$1,035.36
Compute Optimised	Standard_F8	16 GB Memory, 8 Cores, 1.00 DBU	2	\$0.550	\$0.398	\$792.00	\$286.56	\$1,078.56

## DataBricks Components

### Creating Cluster in DataBricks

- In the community Environment Databricks provides 15.3GB RAM and 2 Cores based free Spark cluster for practice.
- Kindly use Exercise 2 for setting up the cluster in your workspace.
- In Community version, your cluster will be terminated automatically if you are not using for 2 hours. So it will be better, if we turn off the cluster once the usage is stopped.

## DataBricks Components

### Adding User in DataBricks

- You can add multiple users in Azure DataBricks using the LDAP or AD integration.
- Your Administrator will have the access on creating the Users and providing the relevant access to the users.
- DataBricks workspace is having an admin console to understand the user privileges.

# DataBricks Components

## Adding User in DataBricks

Click on the upper right human icon to get the admin console link

The screenshot displays the Databricks home interface. On the left is a dark sidebar with navigation icons and labels: databricks, Home, Workspace, Recents, Data, Clusters, Jobs, and Search. The main content area features a 'Welcome to databricks' header. Below it are three primary action cards: 'Explore the Quickstart Tutorial' (with a lightbulb icon), 'Import & Explore Data' (with a cloud upload icon), and 'Create a Blank Notebook' (with a plus icon). Each card includes a brief description of the action. At the bottom, there are three sections: 'Common Tasks' (with links for 'New Notebook', 'Create Table', and 'New Cluster'), 'Recents' (with the text 'Recent files appear here as you work.'), and 'What's new in v3.16' (with a link to 'View latest release notes'). In the top right corner, there is an 'Upgrade' button, a help icon, and a user profile icon. A dropdown menu is open from the user profile icon, showing the user is signed in as 'navaneeth@dossieranal...'. The menu options include 'User Settings', 'Admin Console', 'Partner Integrations', 'Manage Account', 'Log Out', and a 'Workspaces' section with a checked item 'Dossier navaneeth@dossieranal...'.

Upgrade ?

Signed in as  
navaneeth@dossieranal...

- User Settings
- Admin Console
- Partner Integrations
- Manage Account
- Log Out

Workspaces

- ✓ Dossier navaneeth@dossieranal...

Home

Workspace

Recents

Data

Clusters

Jobs

Search

# Welcome to databricks™

Explore the Quickstart Tutorial

Spin up a cluster, run queries on preloaded data, and display results in 5 minutes.

Import & Explore Data

Drop files or [click to browse](#)

Quickly import data, preview its schema, create a table, and query it in a notebook.

Create a Blank Notebook

Create a notebook to start querying and modeling your data.

Common Tasks

- New Notebook
- Create Table
- New Cluster

Recents

Recent files appear here as you work.

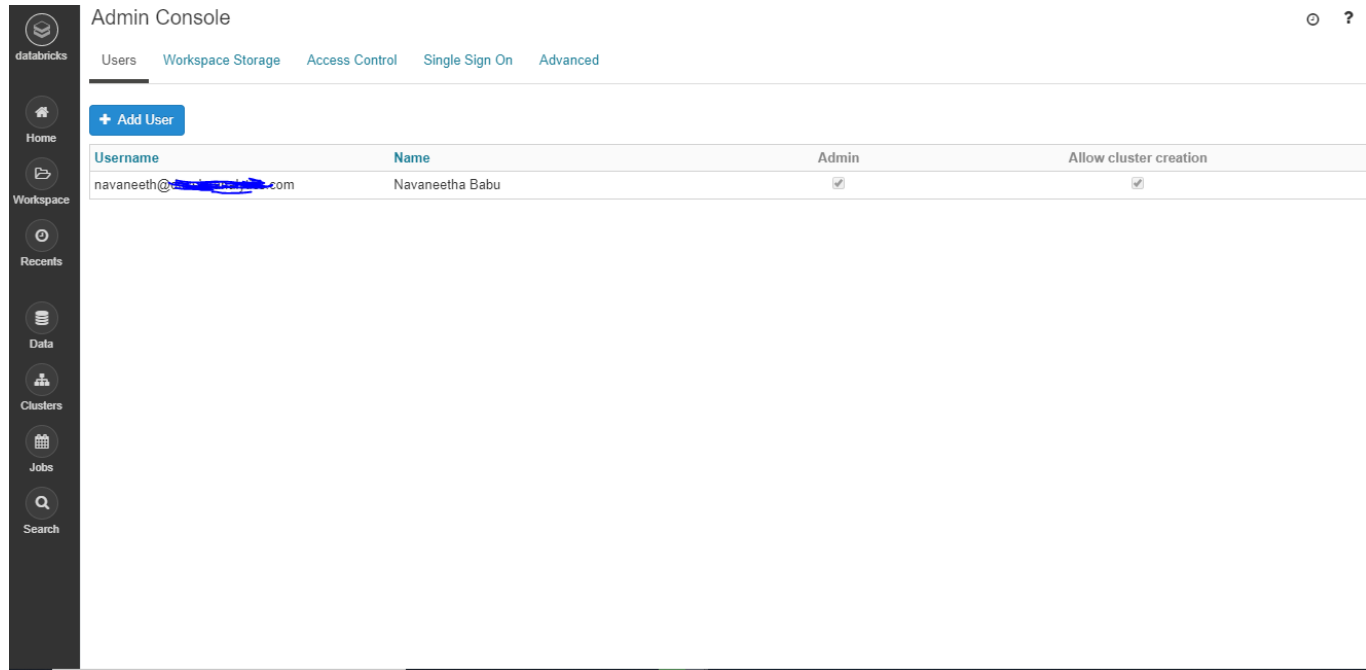
What's new in v3.16

[View latest release notes](#)

# DataBricks Components

## Adding User in DataBricks

This screen provides details about the users. In community version, we can add upto 3 users.



The screenshot displays the Databricks Admin Console interface. On the left is a dark sidebar with navigation icons for Home, Workspace, Recents, Data, Clusters, Jobs, and Search. The main content area is titled 'Admin Console' and has a sub-header 'Users'. Below this, there are tabs for 'Workspace Storage', 'Access Control', 'Single Sign On', and 'Advanced'. A blue '+ Add User' button is located above a table. The table lists user details with columns for Username, Name, Admin status, and Allow cluster creation. One user is listed: navaneeth@...com, Navaneetha Babu, with Admin and Allow cluster creation both checked.

Username	Name	Admin	Allow cluster creation
navaneeth@...com	Navaneetha Babu	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

### Fundamental of NoteBooks

- Notebook provides interface for executing the spark programs in R, Python, Scala or SQL
- Notebook is a cell based processing component. You can write the codes in cell and you can either process it step by step or do overall in a single step.
- Notebook should be linked with the running cluster

### Fundamental of NoteBooks

- Notebook provides interface for executing the spark programs in R, Python, Scala or SQL
- Notebook is a cell based processing component. You can write the codes in cell and you can either process it step by step or do overall in a single step.
- Notebook should be linked with the running cluster

## Fundamental of NoteBooks

- Kindly use Exercise manual to have demo about the notebooks



### Adding Libraries in DataBricks

- DataBricks provides options to use the external libraries like
  - PyPi
  - Maven
  - Cran
- It also provides option to upload the external JARS, Python Egg and Python WHLs
- If we have any custom requirement on the code, we can use the libraries and call the class or functions from it.

### Adding Libraries in DataBricks

- In this Exercise, you will learn about how to use the xml libraries from Maven repository

# DataBricks Components

## Comparing DataBricks with Traditional Spark Cluster

- Traditional Hadoop based Spark clusters provided by Cloudera, Hortonworks and MapR can be implemented in on-premise and cloud virtual machines
- Those infrastructure requires Administrators to Manage each and every components
- Integrating the components is complex
- Job Scheduling becomes spaghetti.

# DataBricks Components

## Comparing DataBricks with Traditional Spark Cluster

- DataBricks provides ACID based Delta Lake which is not available with Traditional Hadoop providers
- Integrated Data Engineering and Data Science Workbench
- Proper Job Scheduling, Easy Auto Scaling the clusters
- Security and Integrations are made easy
- Easy way to develop code

## DataBricks Security - ACLs

- Databricks provides ACLs for
  - Cluster Access
  - Pool Access
  - Job Access
  - Table Access
  - Workspace Access
  - Authentication for connectors

# DataBricks Components

## DataBricks Security - ACLs – Cluster Access

Ability	No Permissions	Can Attach To	Can Restart	Can Manage
Attach notebook to cluster		x	x	x
View Spark UI		x	x	x
View cluster metrics		x	x	x
Terminate cluster			x	x
Start cluster			x	x
Restart cluster			x	x
Edit cluster				x
Attach library to cluster				x
Resize cluster				x
Modify permissions				x

# DataBricks Components

## DataBricks Security - ACLs – Pool Access

By default, all users can create and modify pools unless an administrator enables pool access control. With pool access control, permissions determine a user's abilities.

Ability	No Permissions	Can Attach To	Can Manage
Attach cluster to pool		x	x
Delete pool			x
Edit pool			x
Modify pool permissions			x

# DataBricks Components

## DataBricks Security - ACLs – Pool Access

By default, all users can create and modify pools unless an administrator enables pool access control. With pool access control, permissions determine a user's abilities.

Ability	No Permissions	Can Attach To	Can Manage
Attach cluster to pool		x	x
Delete pool			x
Edit pool			x
Modify pool permissions			x



# DataBricks Components

## DataBricks Security - ACLs – Job Access

Ability	No Permissions	Can View	Can Manage Run	Is Owner	Can Manage (admin)
View job details and settings	x	x	x	x	x
View results, Spark UI, logs of a job run		x	x	x	x
Run now			x	x	x
Cancel run			x	x	x
Edit job settings				x	x
Modify permissions				x	x
Delete job				x	x
Change owner					x

# DataBricks Components

## DataBricks Security - ACLs – WorkSpace Access

Ability	No Permissions	Read	Run	Edit	Manage
List items in folder	x	x	x	x	x
View items in folder		x	x	x	x
Clone and export items		x	x	x	x
Create, import, and delete items					x
Move and rename items					x
Change permissions					x

## DataBricks Components

### DataBricks Security - ACLs – WorkSpace Access

- Folder access control can also be possible
- MLFlow Experiment Permissions are possible in DataBricks
- MLFlow Model Permissions are also possible

### DataBricks Security - ACLs – Table Access Control

- By default, all users have access to all data stored in a cluster's managed tables unless an administrator enables table access control for that cluster. Once table access control is enabled for a cluster, users can set permissions for data objects on that cluster.
- Grant and revoke access to your data using the Databricks view-based access control model.



**THANK YOU**