UdeA
Analytics &
DataScience

DATA STREAMING Y SERVICIOS EN LA NUBE

DATABRICKS

Magister - Efraín Alberto Oviedo
alberto.oviedo@udea.edu.co

UNIVERSIDAD DE ANTIOQUIA
FACULTAD DE INGENIERÍA
ESPECIALIZACIÓN EN ANALÍTICA Y CIENCIA DE DATOS

AGENDA

UdeA
Analytics &
DataScience

1. **Databricks**
2. Cluster Spark
   - SQL
   - R
   - Python
   - Scala
3. Streaming

Qué es Databricks

- Plataforma unificada de analítica de datos desarrollada por los creadores de Apache Spark

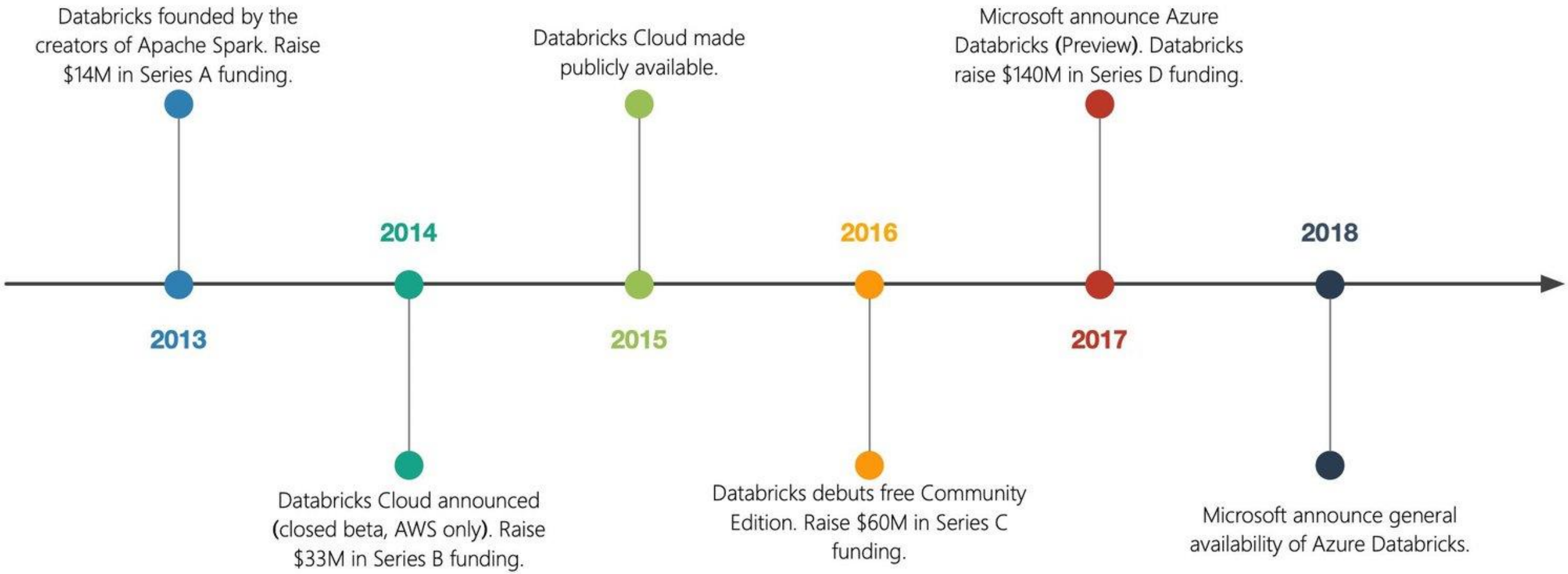
- Compatible con lenguajes:

  Python – SQL – Scala - R

- Servicio de Clúster Spark en la nube
  - Disponible en versión community
  - Compatible con:

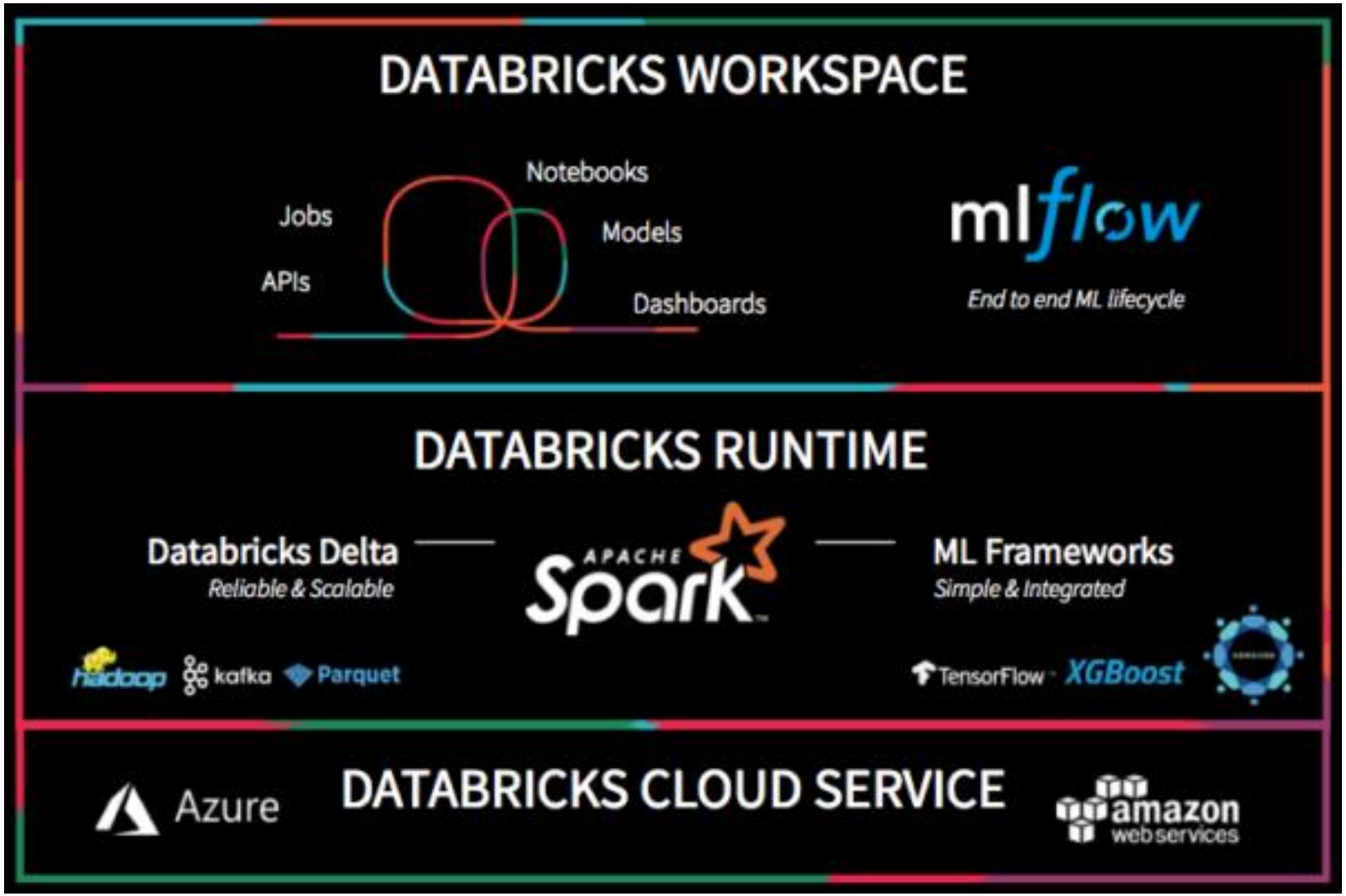
aws Microsoft Azure Google Cloud

Historia Databricks



https://www.taygan.co/blog/2018/12/02/azure-databricks

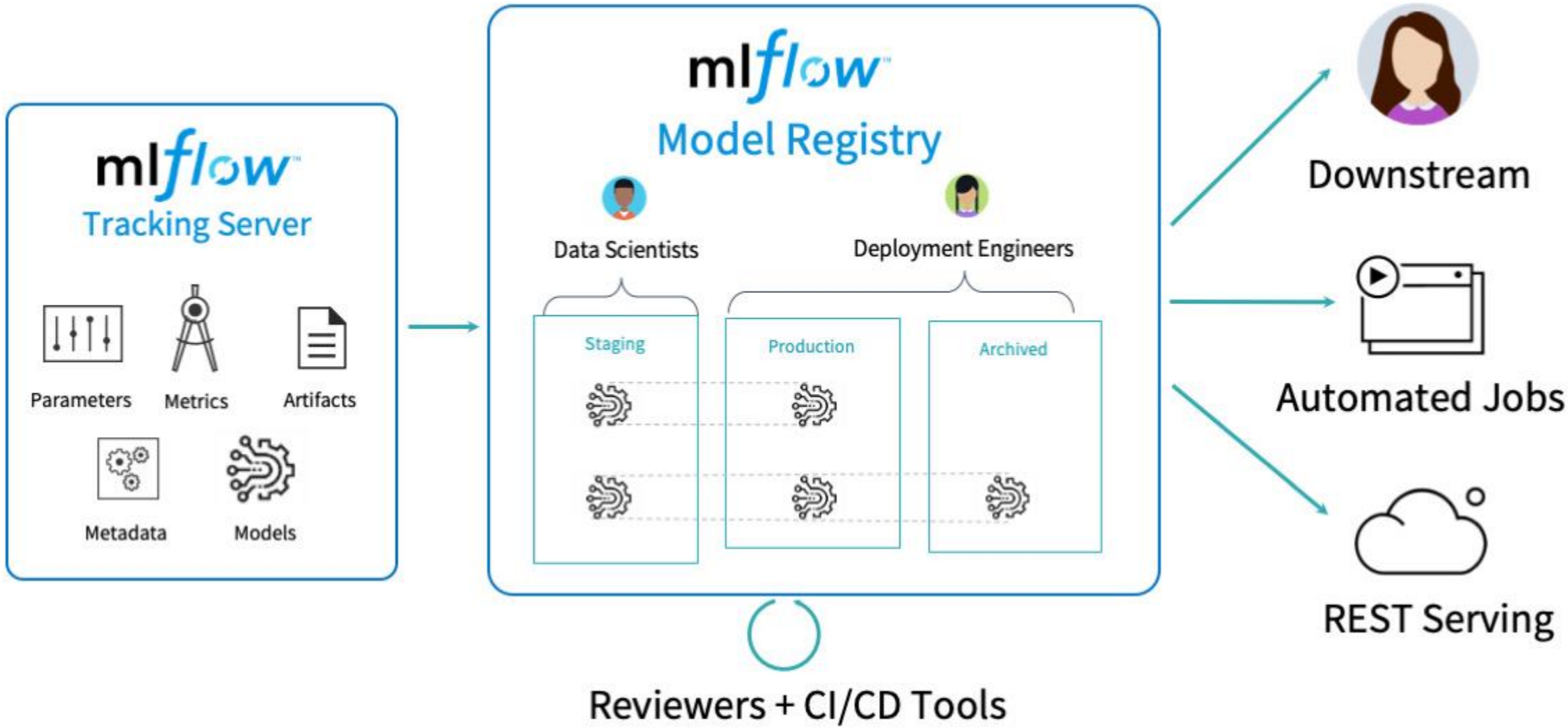
Componentes Databricks



https://www.databricks.com/blog/2018/06/05/accelerating-innovation-with-unified-analytics.html

Delta Lake

- Capa de almacenamiento open Source desarrollada por Databricks
- Proporciona transacciones ACID:
  - Atomicidad: Todas las transacciones tienen éxito o fallan por completo
  - Consistencia: Cada cambio debe conducir a un estado válido
  - Aislamiento: Resuelve conflictos en las operaciones simultáneas
  - Durabilidad: Cambios permanentes
- Permite consultas interactivas rápidas
- Control escalable de metadatos
- Permite operaciones de actualización y eliminación de registros



https://www.databricks.com/blog/2020/04/15/databricks-extends-mlflow-model-registry-with-enterprise-features.html

Figure 1: Magic Quadrant for Data Science and Machine Learning Platforms

Source: Gartner (March 2021)

Plataformas de Ciencia de Datos

https://www.databricks.com/blog/2021/03/04/databricks-named-a-leader-in-2021-gartner-magic-quadrant-for-data-science-and-machine-learning-platforms.html

Casos de Uso


Healthcare & Life Sciences
McKESSON
SANOFI
gsk
AstraZeneca
OPTUM
REGENERON
SANFORD HEALTH
Financial Services
HSBC
Commerce Bank
Jefferies
BNP PARIBAS
FIS
finra
FRANKLIN TEMPLETON INVESTMENTS
Nationwide
NEUBERGER BERMAN
Media & Entertainment
COMCAST NBCUNIVERSAL
RIOT GAMES
A&E
nielsen
SHOWTIME
viacom
Turner
Retail, CPG, and eCommerce
CARMAX
Foot Locker
Altria
sam's club
Columbia
H&M
Electrolux
zalando
overstock.com
Public Sector
CMS
BLACK|SKY
Be The First To Know
SEVATEC
noblis
THE WORLD BANK
DigitalGlobe
Manufacturing
DAIMLER
MAGNETI MARELLI
devon
BECHTEL
LENNOX
ZEISS
Technology
McAfee
hp
CISCO
SAMSUNG
Adobe
NVIDIA
GoPro
Barracuda
Other
OpenTable
Hotels.com
myfitnesspal
Quby
GIPHY
TOMTOM
JAMCITY

Casos de Uso en América Latina

Links de Apoyo

- Documentación Oficial

https://docs.databricks.com/introduction/index.html

- Databricks Community

https://community.databricks.com

- Azure Databricks

https://learn.microsoft.com/es-es/azure/databricks/getting-started/free-training?source=recommendations

- AWS

https://aws.amazon.com/es/quickstart/architecture/databricks/

Crear Cuenta Databricks

1. Ingrese a la página oficial de Databricks Community
   https://www.databricks.com/try-databricks#account 1

2. Llene el formulario con los datos personales

3. Indicar su proveedor de nube, en este caso seleccione la **Community Edition**

4. Revise su correo electrónico y verifique la cuenta

Crear Cuenta Databricks

databricks

Try Databricks free

Test-drive the full Databricks platform free for 14 days on your choice of AWS, Microsoft Azure or Google Cloud.

- Simplify data ingestion and automate ETL
  Ingest data from hundreds of sources. Use a simple declarative approach to build data pipelines.
- Collaborate in your preferred language
  Code in Python, R, Scala and SQL with coauthoring, automatic versioning, Git integrations and RBAC.
- 12x better price/performance than cloud data warehouses
  See why over 7,000 customers worldwide rely on Databricks for all their workloads from BI to AI.

2

Create your Databricks account 1/2

First name

Last name

Email

Company

Title

Phone (Optional)

Country

Colombia

By submitting, I agree to the processing of my personal data by Databricks in accordance with our Privacy Policy. I understand I can update my preferences at any time.

Continue

databricks

Choose a cloud provider

- Amazon Web Services
- Microsoft Azure
- Google Cloud Platform

Get started

By clicking "Get started", you agree to the **Privacy Policy** and **Terms of Service**

Don't have a cloud account?

Community Edition is a limited Databricks environment for personal use and training.

Get started with Community Edition

By clicking "Get started with Community Edition", you agree to the **Privacy Policy** and **Community Edition Terms of Service**

3

databricks

Check your email to start your trial.

Thank you for signing up. Please validate your email address to start your trial.

Here are some resources to help you deploy your first workspace.

1. Review the administration guide on the requirements to set up your Databricks service.
    - Not an admin on your AWS Account? Share this guide with your admin to deploy a workspace for you!
2. Follow our Quickstart guide to create your first workspace.

You can also check out our Docs and Community sites to get your questions answered.

Note: if you signed up for Community Edition, you'll go to your first workspace as soon as you verify your email address.

Welcome to Databricks! Please verify your email address. Recibidos ×

Databricks <noreply@databricks.com>
para efrain.oviedo

inglés > español Traducir mensaje

databricks

Welcome to Databricks Community Edition!

Databricks Community Edition provides you with access to a free micro-cluster as well as a cluster manager and a notebook environment - ideal for developers, data scientists, data engineers and other IT professionals to get started with Spark.

We need you to verify your email address by clicking on this link. You will then be redirected to Databricks Community Edition!

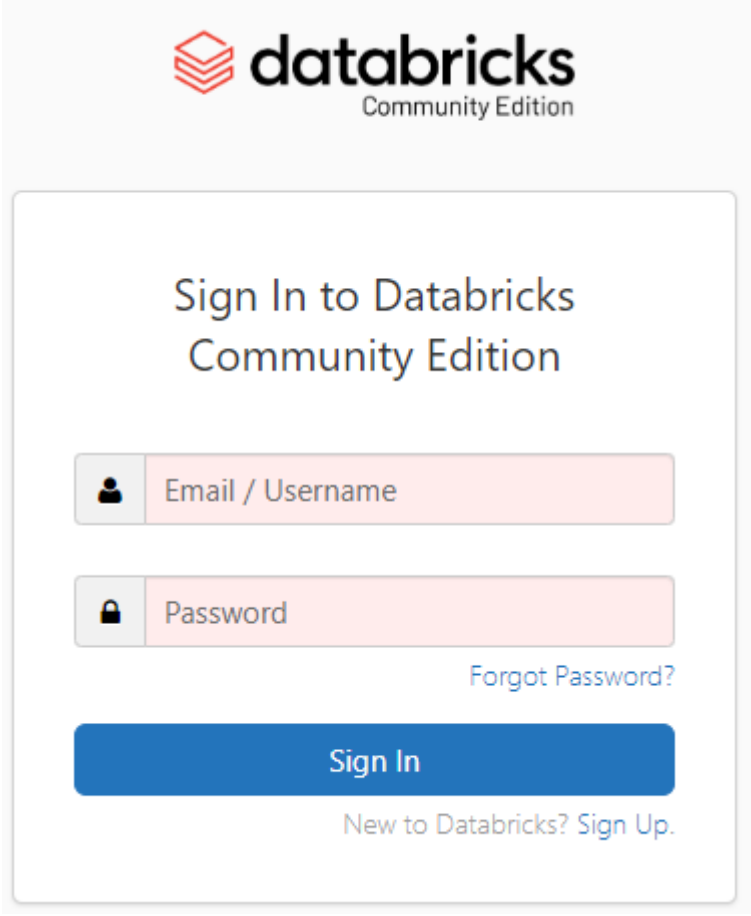
4

Your sign-in email: efrain.oviedo@upb.edu.co

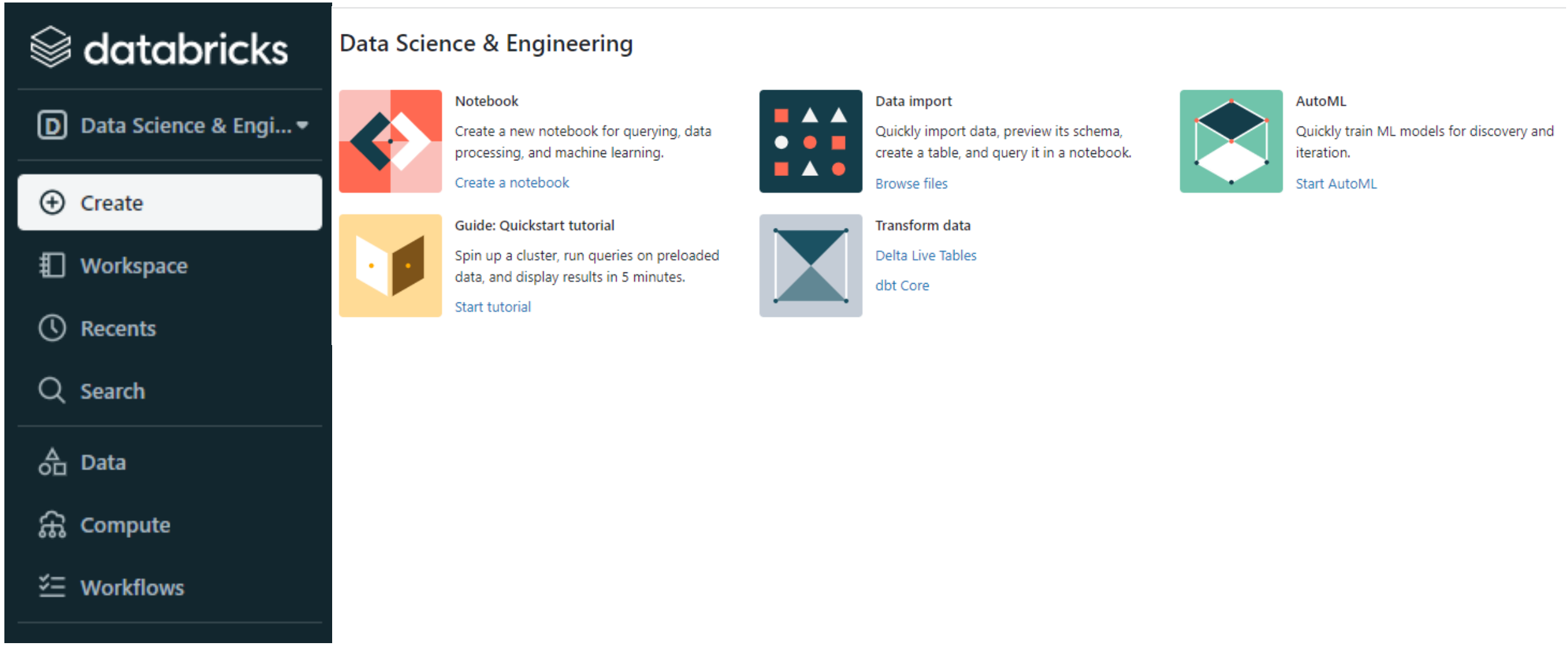
Get started by visiting: https://community.cloud.databricks.com/login.html?resetpassword&username=efrain.oviedo%40upb.edu.co&expiration=-60000&token=2555d74de4b7987b9b3bd6b9a76d3d47a0da246a&accountid=0fbe09ab-8085-4272-bf9b-2cb68bf0e02a

If you have any questions, please contact feedback@databricks.com.

- The Databricks Team



Interfaz Databricks

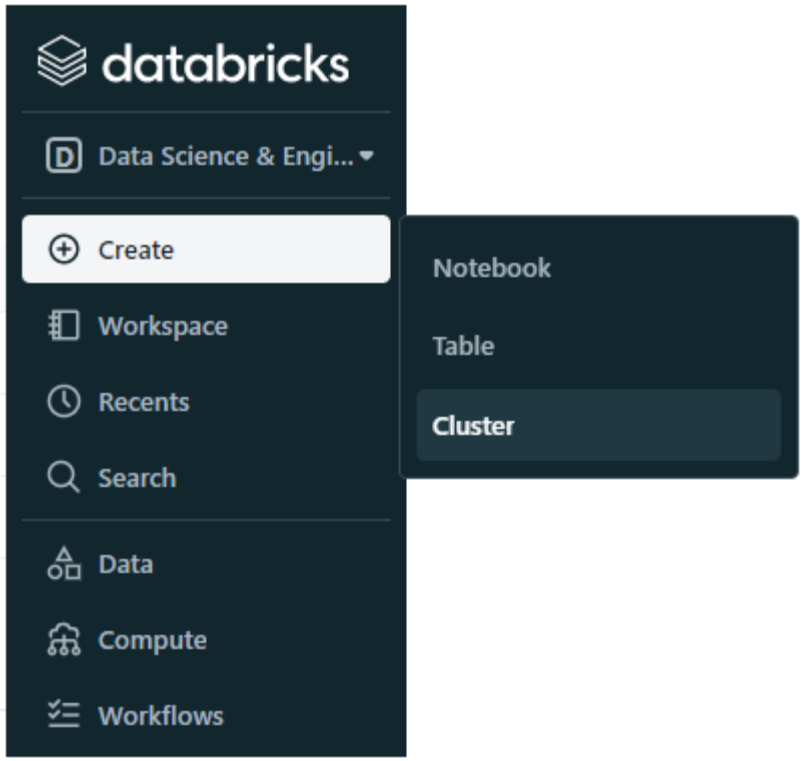


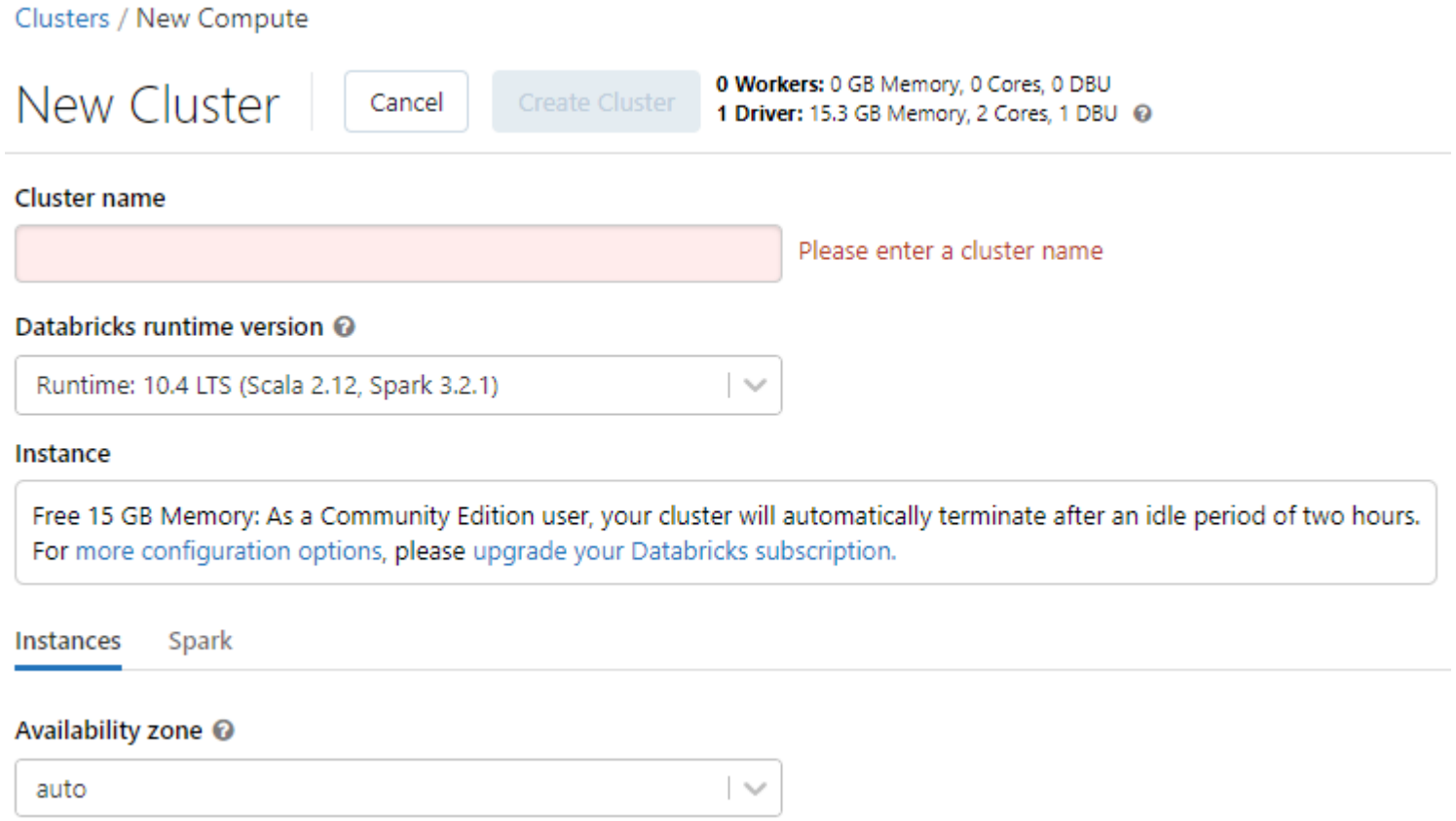
AGENDA

UdeA
Analytics &
DataScience

1. Databricks
2. **Cluster Spark**
   - SQL
   - R
   - Python
   - Scala
3. Streaming

Cluster Spark



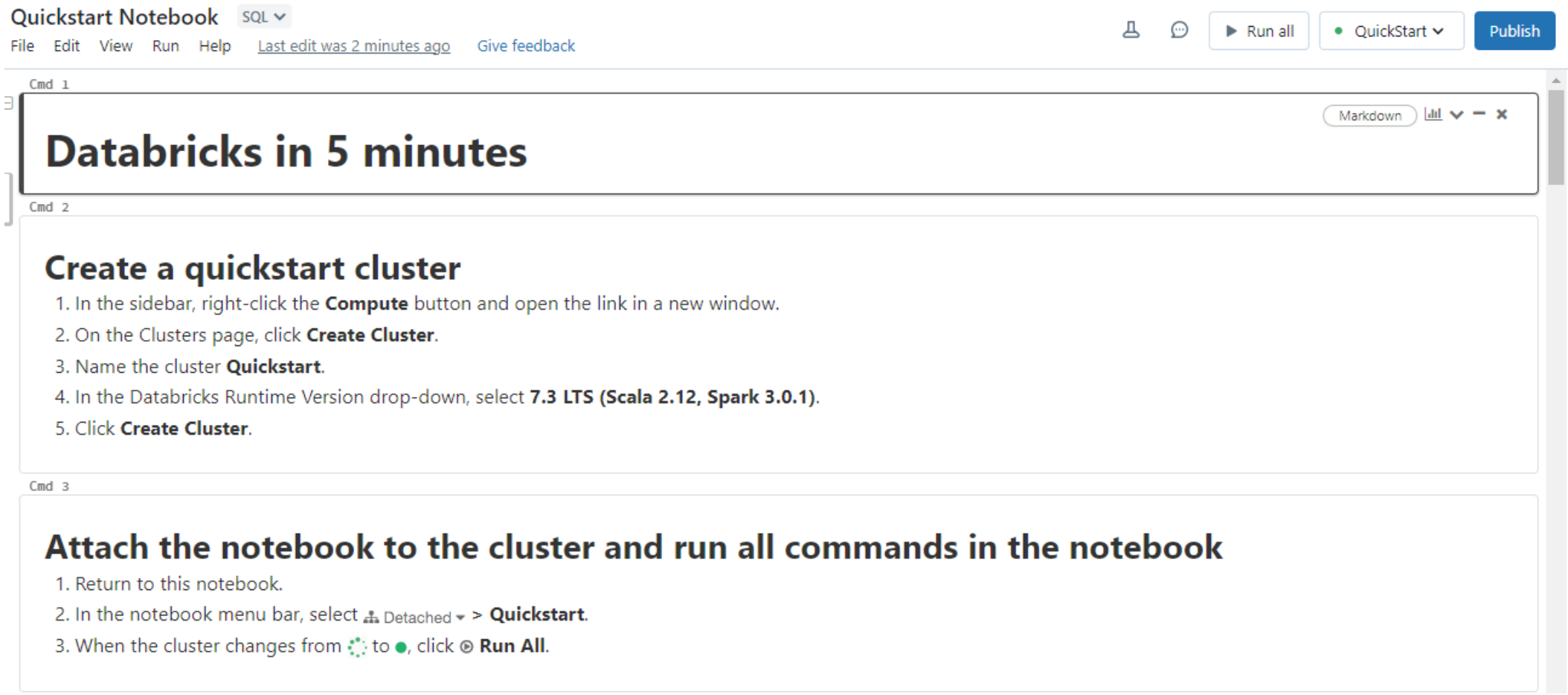


Quickstart Notebook

Guide: Quickstart tutorial

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

Start tutorial



https://www.kaggle.com/datasets/shivam2503/diamonds

Quickstart Notebook

Table +

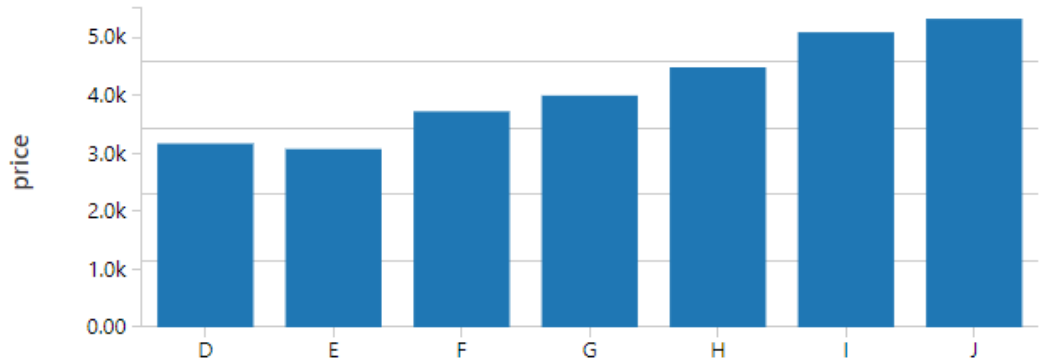
	_c0	carat	cut	color	clarity	depth	table	price	x	y	z
1	1	0.23	Ideal	E	SI2	61.5	55	326	3.95	3.98	2.43
2	2	0.21	Premium	E	SI1	59.8	61	326	3.89	3.84	2.31
3	3	0.23	Good	E	VS1	56.9	65	327	4.05	4.07	2.31
4	4	0.29	Premium	I	VS2	62.4	58	334	4.2	4.23	2.63
5	5	0.31	Good	J	SI2	63.3	58	335	4.34	4.35	2.75
6	6	0.24	Very Good	J	VVS2	62.8	57	336	3.94	3.96	2.48

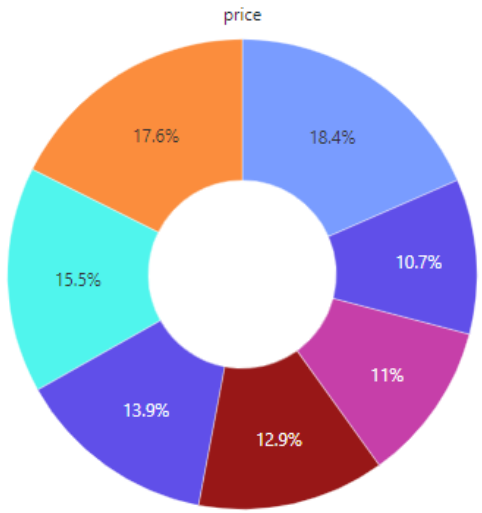
Truncated results, showing first 1,000 rows. | 1.22 seconds runtime Refreshed 20 minutes ago

Table Bar chart Pie +

	color	price
1	D	3169.9540959409596
2	E	3076.7524752475247
3	F	3724.886396981765
4	G	3999.135671271697
5	H	4486.669195568401
6	I	5091.874953891553
7	J	5323.81801994302

Table Bar chart Pie +





Conjuntos de datos Databricks

Agregue la siguiente celda

```
%python
display(dbutils.fs.ls('/databricks-datasets'))
```

▸ (3) Spark Jobs

Table +

	path	name	size	modificationTime
1	dbfs:/databricks-datasets/	databricks-datasets/	0	0
2	dbfs:/databricks-datasets/COVID/	COVID/	0	0
3	dbfs:/databricks-datasets/README.md	README.md	976	1532468253000
4	dbfs:/databricks-datasets/Rdatasets/	Rdatasets/	0	0
5	dbfs:/databricks-datasets/SPARK_README.md	SPARK_README.md	3359	1455043490000
6	dbfs:/databricks-datasets/adult/	adult/	0	0
7	dbfs:/databricks-datasets/airlines/	airlines/	0	0

Showing all 55 rows. | 1.44 seconds runtime

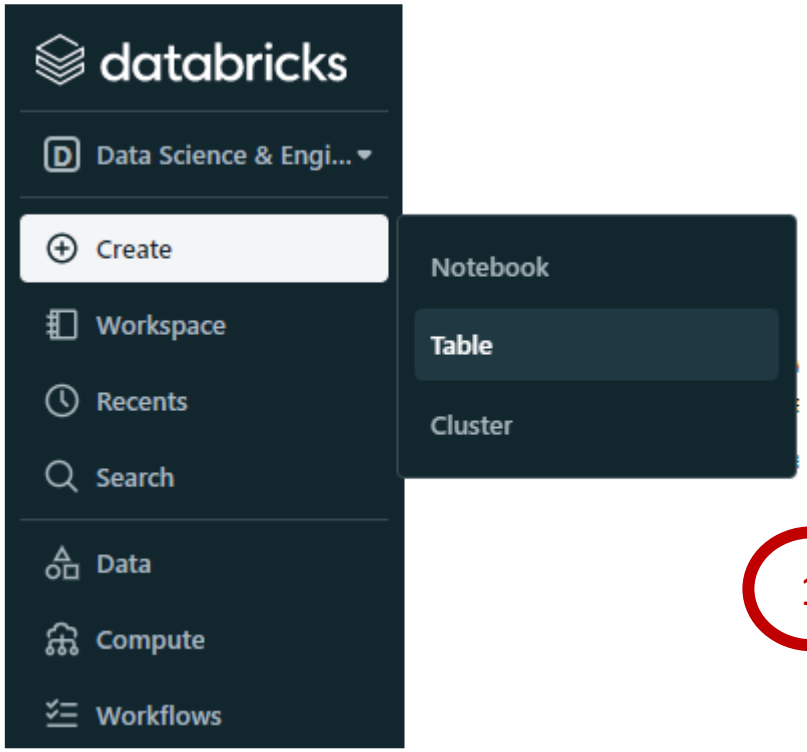
Utilidades DBFS: https://docs.databricks.com/dev-tools/databricks-utils.html#file-system-utility-dbutilsfs

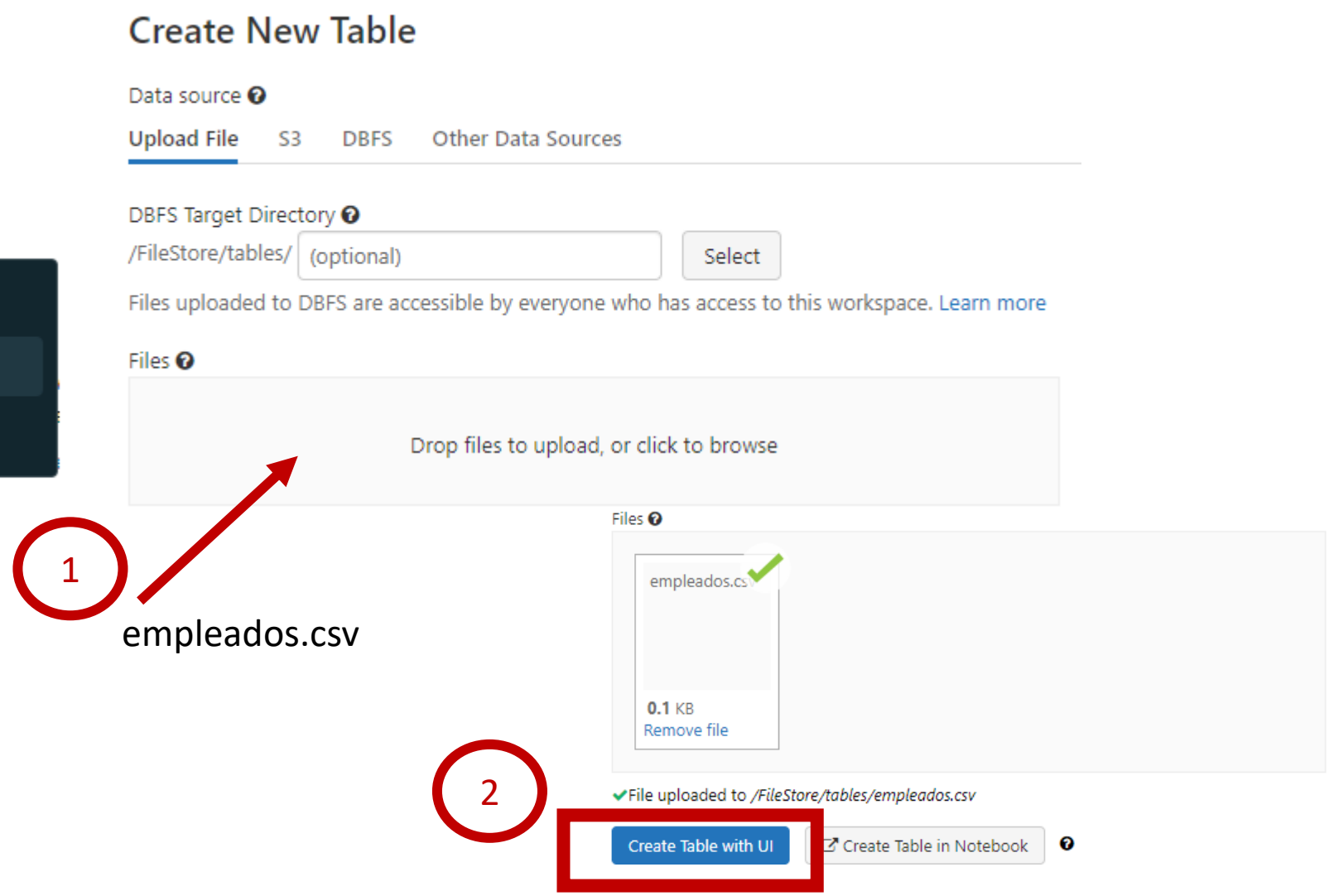
AGENDA

UdeA
Analytics &
DataScience

1. Databricks
2. **Cluster Spark**
   - **SQL**
   - R
   - Python
   - Scala
3. Streaming

Crear Tabla


databricks
Data Science & Engi...
Create
Workspace
Recents
Search
Data
Compute
Workflows
Notebook
Table
Cluster


Create New Table
Data source
Upload File
S3
DBFS
Other Data Sources
DBFS Target Directory
/FileStore/tables/
(optional)
Select
Files uploaded to DBFS are accessible by everyone who has access to this workspace. Learn more
Files
Drop files to upload, or click to browse
1
empleados.csv
Files
empleados.cs
0.1 KB
Remove file
2
File uploaded to /FileStore/tables/empleados.csv
Create Table with UI
Create Table in Notebook

Crear Tabla

Select a Cluster to Preview the Table

Choose a cluster with which you will read and preview the data.

Cluster

Test1

3

Preview Table

4

Specify Table Attributes

Specify the Table Name, Database and Schema to add this to the data UI for other users to access

Table Name

empleados_csv

Create in Database

default

File Type

CSV

Column Delimiter

,

First row is header

Infer schema

Multi-line

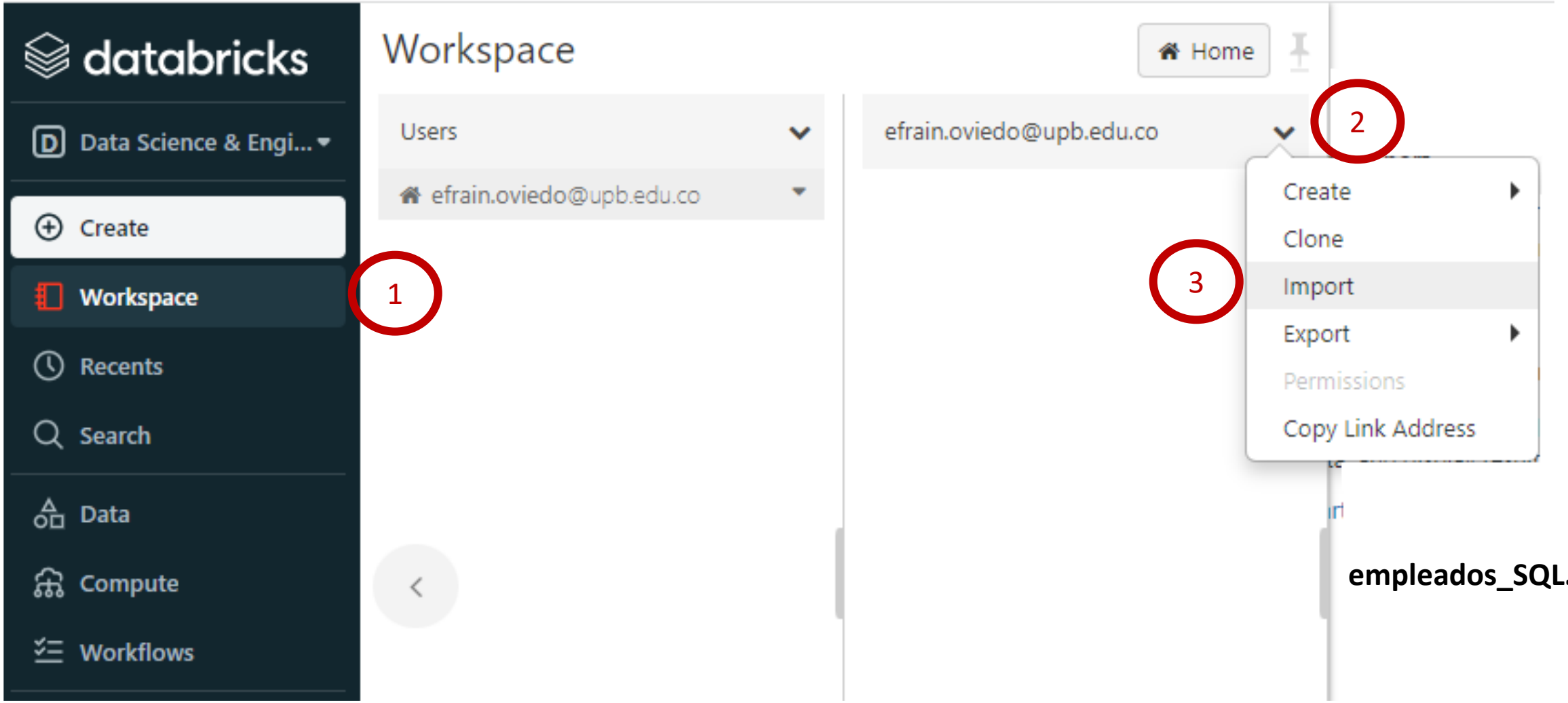
5

Create Table

Table Preview

Id	Nombre	Edad	Sexo
INT	STRING	INT	STRING
1	Camilo Sainz	24	Masculino
2	Javier Ballesteros	45	Masculino
3	Jacinta Pinto	42	Femenino
4	Martina Quevedo	35	Femenino
5	Braulio Blasco	41	Masculino
6	Claudia Palacio	22	Femenino

Importar Notebook



Empleados SQL

AGENDA

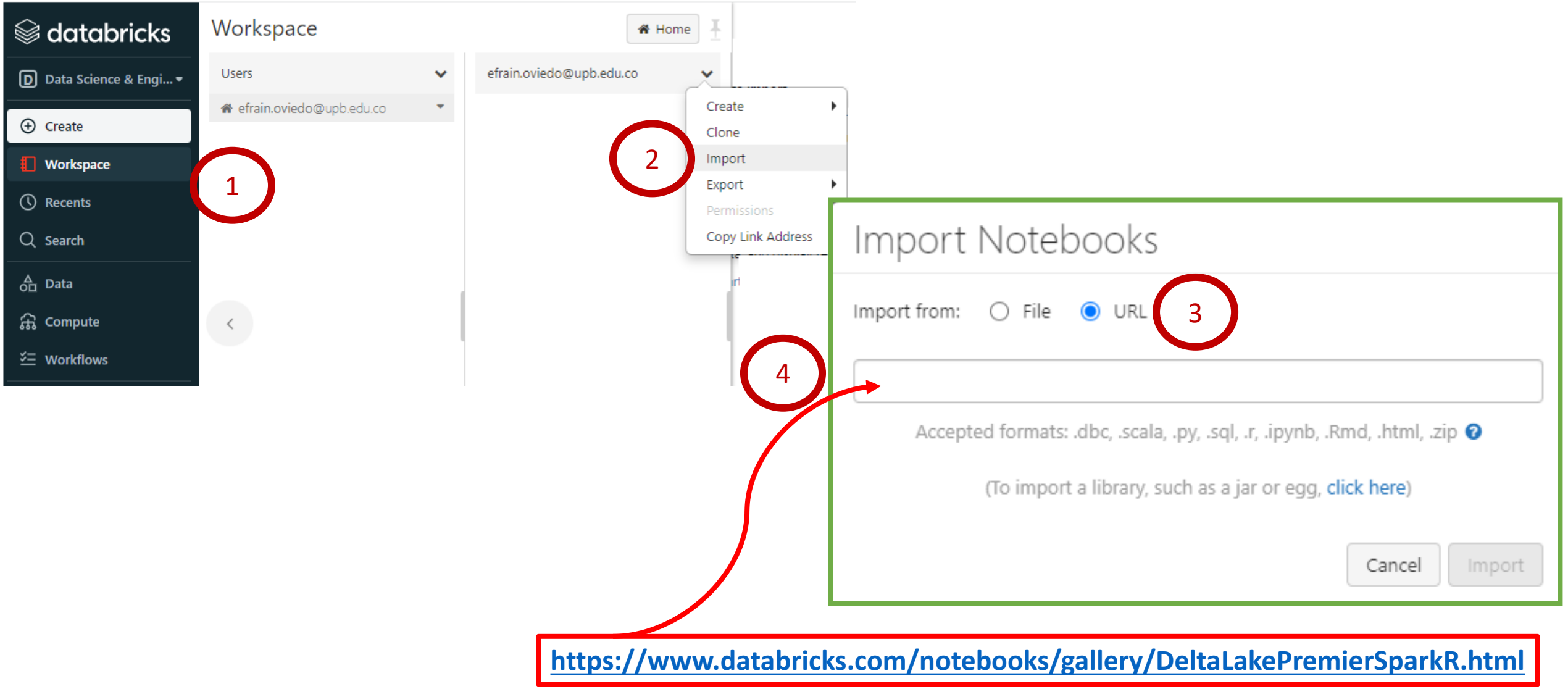
UdeA
Analytics &
DataScience

1. Databricks
2. **Cluster Spark**
   - SQL
   - **R**
   - Python
   - Scala
3. Streaming

Generalidades de R

- Lenguaje de programación interpretado enfocado en el análisis estadístico
- Entorno de Software libre bajo licencia GNU GPL
- Proyecto colaborativo y abierto
- Incluye utilidades gráficas para la visualización de datos
- Utilizado en Big data para manipulación, procesamiento y visualización de datos

Importar Notebook desde URL



Notebook R

Markdown

Description

Delta Lake Primer - SparkR

This is a companion notebook to provide a Delta Lake example against the Lending Club data. It illustrates all functionality available in Delta Lake such as:

- Import data from Parquet to Delta Lake
- Batch and streaming updates
- Delete, update, and merge DML operations
- Schema evolution and enforcement.
- Time Travel

Run this cell by cell. Some cells will fail to illustrate lack of missing functionality in Parquet files but the subsequent operation on Delta Lake will

AGENDA

UdeA
Analytics &
DataScience

1. Databricks
2. **Cluster Spark**
   - SQL
   - R
   - **Python**
   - Scala
3. Streaming

Llamadas a bomberos de San Francisco

https://data.sfgov.org/Public-Safety/Fire-Department-Calls-for-Service/nuek-vuh3

AGENDA

UdeA
Analytics &
DataScience

1. Databricks
2. **Cluster Spark**
   - SQL
   - R
   - Python
   - **Scala**
3. Streaming

Generalidades Scala

- Lenguaje de programación de propósito general desarrollado en 2001 que se ejecuta sobre la JVM
- Multiparadigma: Combina propiedades de lenguajes orientados a objetos y lenguajes funcionales
- Diseñado para expresar patrones comunes de programación en forma concisa, elegante y con tipos seguros

Datasets Scala

Importar Notebook: Datasets_Scala.dbc

Cmd 1

Markdown

Spark Datasets with Scala

This notebook demonstrates a number of common Spark Dataset functions using Scala. It also demostrates how structure enables developers to express high-level queries that are readable and composable. They look like SQL queries you would express, or domain specific language computation you would perform on your data set.

Keep this URL or open in the new tab to consult Dataset API

Cmd 2

Setup: Create Sample Data to demonstrate Datasets.

Cmd 3

```scala
// Create the case classes for our domain
case class Department(id: String, name: String)
case class Employee(firstName: String, lastName: String, email: String, salary: Int)
case class DepartmentWithEmployees(department: Department, employees: Employee)

// Create the Departments
val department1 = new Department("123456", "Computer Science")
val department2 = new Department("345678", "Mechanical Engineering")
val department3 = new Department("123456", "Theater and Drama")
val department4 = new Department("901234", "Indoor Recreation")

```

https://databricks-prod-cloudfront.cloud.databricks.com/public/4027ec902e239c93eaaa8714f173bcfc/8599738367597028/1499152197856461/3601578643761083/latest.html

AGENDA

UdeA
Analytics &
DataScience

1. Databricks
2. Cluster Spark
   - SQL
   - R
   - Python
   - Scala
3. **Streaming**

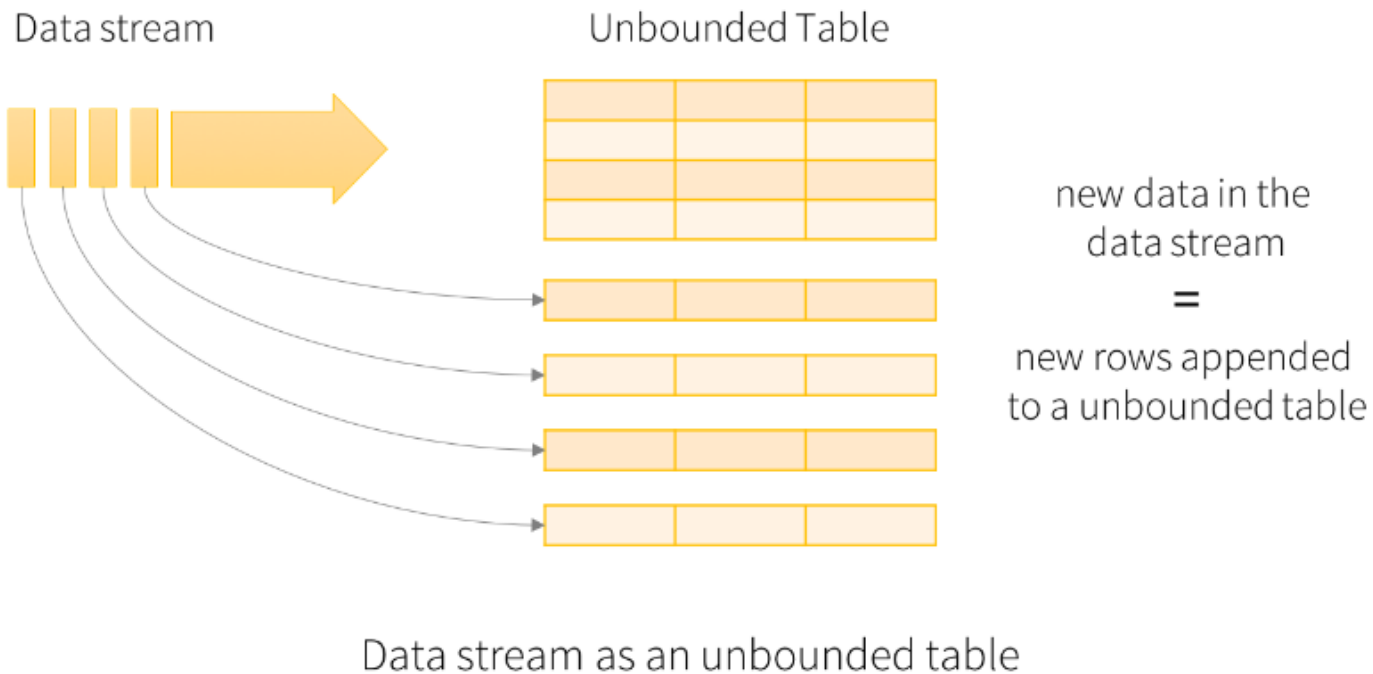
Streaming en Spark

- La implementación de Streaming en Spark permite operar dataframes que reciben transmisiones de datos, de la misma forma como se operan los datos en batch. El procesamiento es rápido, escalable, tolerante a fallas

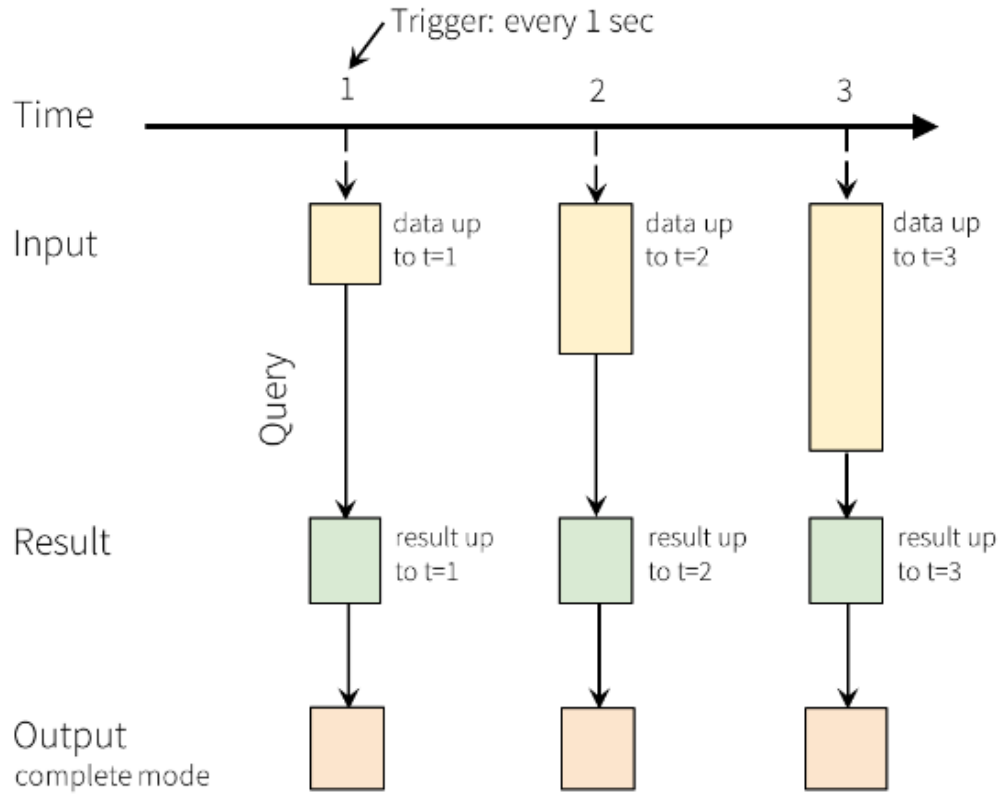
- Por defecto las consultas se ejecutan en microlotes con latencias del orden de 100ms

- También se incluye el procesamiento continuo donde se logran latencias de 1ms

Streaming en Spark



Data stream as an unbounded table



Programming Model for Structured Streaming

https://spark.apache.org/docs/latest/structured-streaming-programming-guide.html

Contador de palabras

```
# Create DataFrame representing the stream of input lines from connection to localhost:9999
lines = spark \
    .readStream \
    .format("socket") \
    .option("host", "localhost") \
    .option("port", 9999) \
    .load()

# Split the lines into words
words = lines.select(
   explode(
       split(lines.value, " ")
   ).alias("word")
)

# Generate running word count
wordCounts = words.groupBy("word").count()
```

Crear el Streaming y definir el procesamiento

Contador de palabras

```
query = wordCounts \
    .writeStream \
    .outputMode("complete") \
    .format("console") \
    .start()

query.awaitTermination()
```

Iniciar el Streaming

```
# TERMINAL 1:
# Running Netcat

$ nc -lk 9999
apache spark
apache hadoop
```

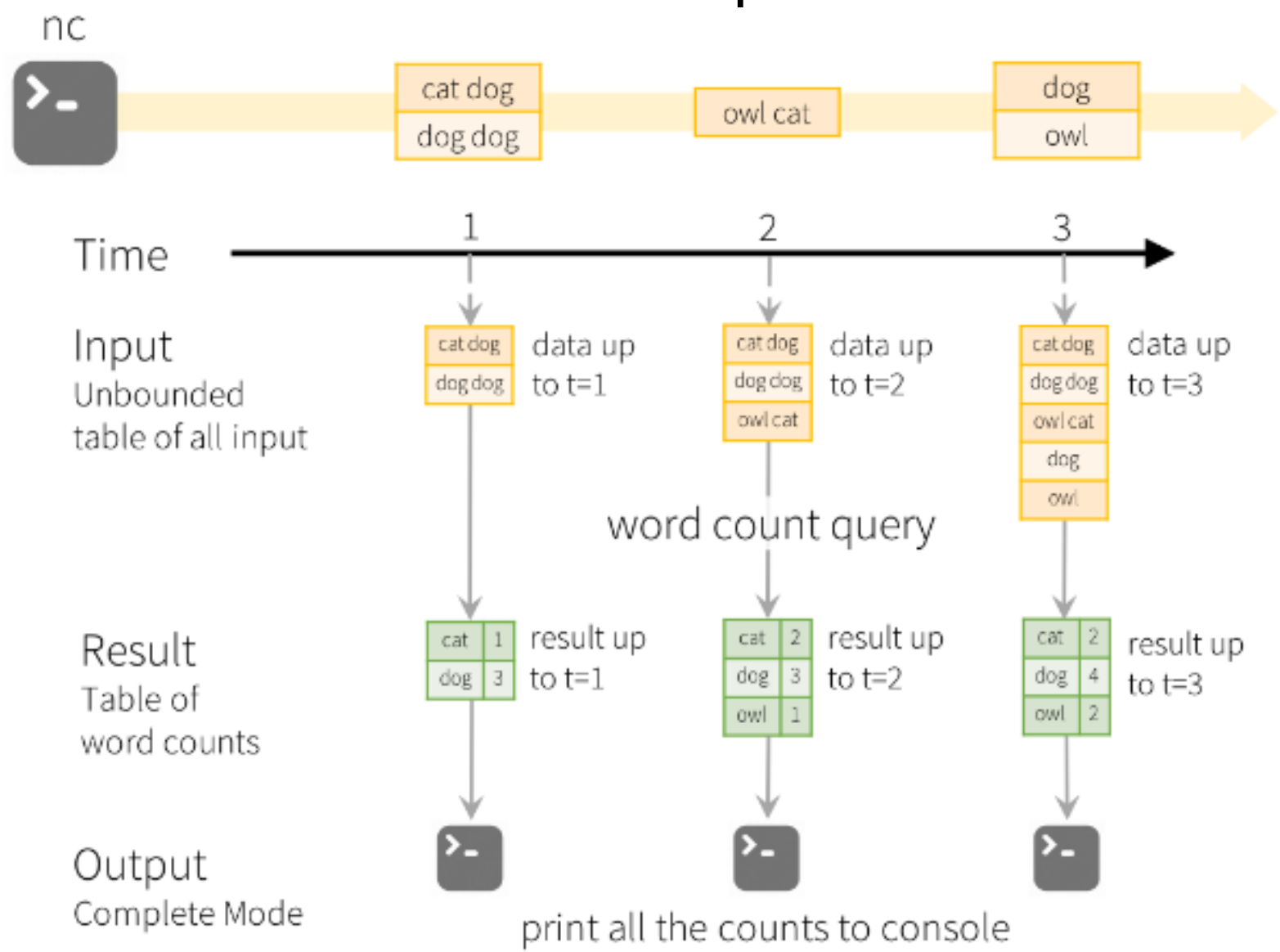
Transmisor

```
# TERMINAL 2: RUNNING structured_network_wordcount.py
-------------------------------------------
Batch: 0
-------------------------------------------
+------+-----+
| value|count|
+------+-----+
|apache|    1|
| spark|    1|
+------+-----+


-------------------------------------------
Batch: 1
-------------------------------------------
+------+-----+
| value|count|
+------+-----+
|apache|    2|
| spark|    1|
|hadoop|    1|
+------+-----+
```

Receptor

Contador de palabras



Ejemplo Batch vs Streaming

Batch

```
dataDeviceSchema = StructType([
    StructField("id",LongType(),False),
    StructField("user_id",LongType(),True),
    StructField("device_id",LongType(),True),
    StructField("num_steps",LongType(),True),
    StructField("miles_walked",FloatType(),True),
    StructField("calories_burnt",FloatType(),True),
    StructField("timestamp",StringType(),True),
    StructField("value",StringType(),True)
])
```

Cmd 6

```
dataDevice_df = spark.read.schema(dataDeviceSchema).json('dbfs:/databricks-datasets/iot-stream/data-device/')
```

Ejemplo Batch vs Streaming

Streaming

Cmd 22

```python
dataDeviceSchema = StructType([
    StructField("id",LongType(),False),
    StructField("user_id",LongType(),True),
    StructField("device_id",LongType(),True),
    StructField("num_steps",LongType(),True),
    StructField("miles_walked",FloatType(),True),
    StructField("calories_burnt",FloatType(),True),
    StructField("timestamp",StringType(),True),
    StructField("value",StringType(),True)
])
```

Cmd 23

```python
dataDevice_df = spark.readStream.schema(dataDeviceSchema).json('dbfs:/databricks-datasets/iot-stream/data-device/')
```

```python
smokerAgg_df.writeStream.format("delta").outputMode("complete").option("checkpointLocation",
"/mnt/delta/eventsByCustomer/_checkpoints/streaming-agg").start("default.smokerAgg")
```