

Feb 22nd, 2022

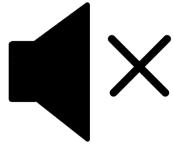
Cloud Data Governance and Catalog – Databricks Notebook Case Study

Srinivasa Gopal - Principal Technologist, Customer Success

Sachin Jain - Principal Technologist, Customer Success



Housekeeping Tips



- Today's Webinar is scheduled for **1 hour**
- The session will include a webcast and then your questions will be answered live at the end of the presentation
- All dial-in participants will be muted to enable the speakers to present without interruption
- Questions can be submitted to "All Panelists" via the **Q&A option** and we will respond at the end of the presentation
- The webinar is **being recorded** and will be available on our **INFASupport YouTube channel** and **[Success Portal](#)** - where you can download the **slide deck** for the presentation. The link to the recording will be emailed as well.
- Please take time to complete the **post-webinar survey** and provide your feedback and suggestions for upcoming topics.

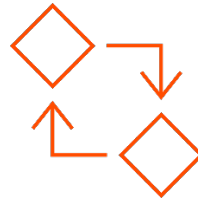
Feature Rich Success Portal



Bootstrap trial and
POC Customers



Enriched Customer
Onboarding
experience



Product Learning
Paths and Weekly
Expert Sessions

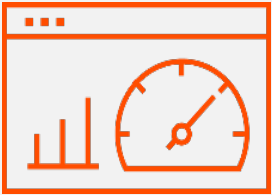


Informatica
Concierge



Tailored training and
content
recommendations

More Information



Success Portal

<https://success.informatica.com>



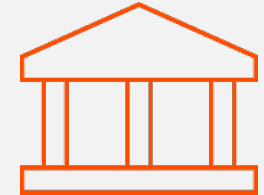
Communities & Support

<https://network.informatica.com>



Documentation

<https://docs.informatica.com>



University

<https://www.informatica.com/in/services-and-training/informatica-university.html>

Safe Harbor

The information being provided today is for informational purposes only. The development, release, and timing of any Informatica product or functionality described today remain at the sole discretion of Informatica and should not be relied upon in making a purchasing decision.

Statements made today are based on currently available information, which is subject to change. Such statements should not be relied upon as a representation, warranty or commitment to deliver specific products or functionality in the future.

Agenda

1

Overview of CDGC

2

Databricks
Architecture
Patterns

3

Databricks
Scanners

4

Databricks
Metadata

5

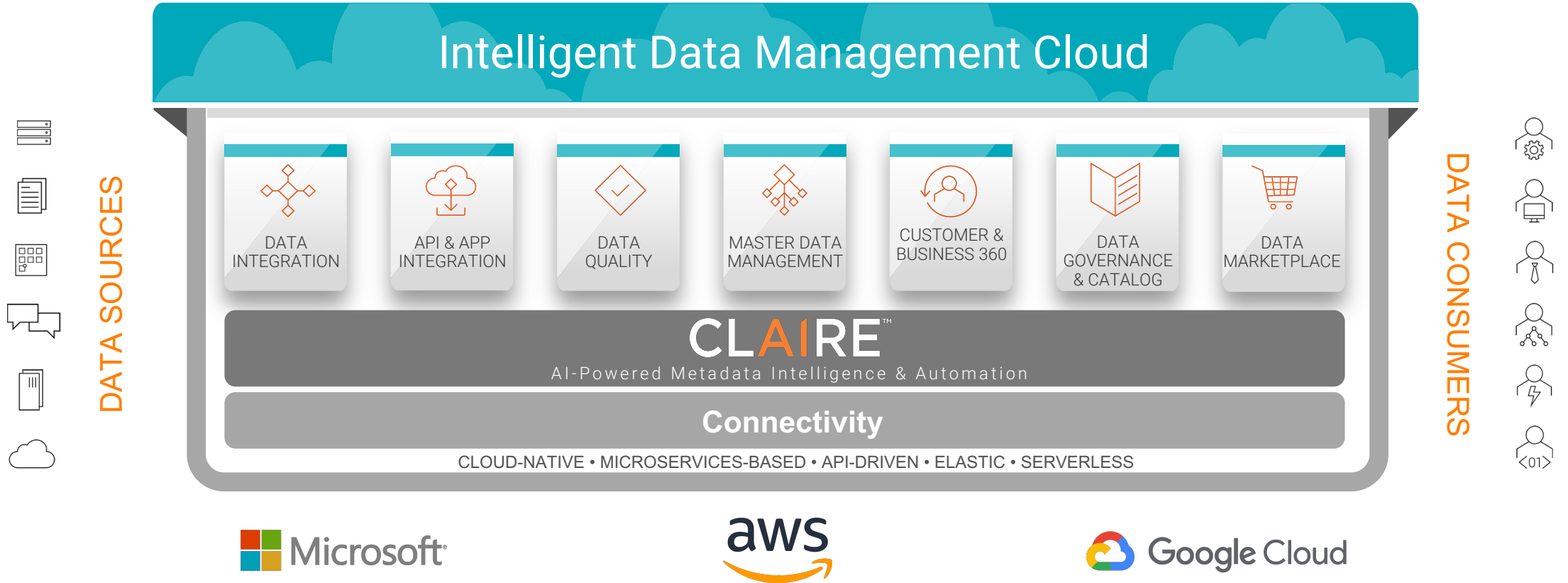
Demo

6

Q&A

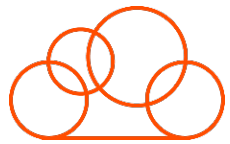
Overview of CDGC

Intelligent Data Management Cloud



Intelligent Data Management Cloud

A SaaS solution that drives the success of your Cloud Data Lake and Data Warehouse implementation—provides data context, lowers privacy risks, builds trust and delivers insight



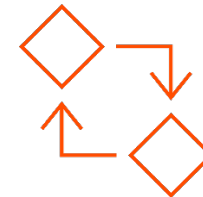
SaaS
Cloud Native



Data **Lineage**,
Profiling, Discovery,
Source Context



Glossary,
Business Context, AI
Models, DQ & Privacy
Standards



Integrated with
Cloud Catalogs,
Integrated with CDQ



Business User,
Navigation
Experience

Cloud Data Governance and Catalog

Core Solution Tenets



SERVERLESS

Serverless compute to work at any scale—there is no infrastructure to deploy or manage



MULTI-CLOUD

Works across all cloud ecosystems and on-premises data sources



INTEGRATED

Deep Integration with data management and data consumption tools



AI

Uses state of the art machine learning and intelligence to ensure end user productivity

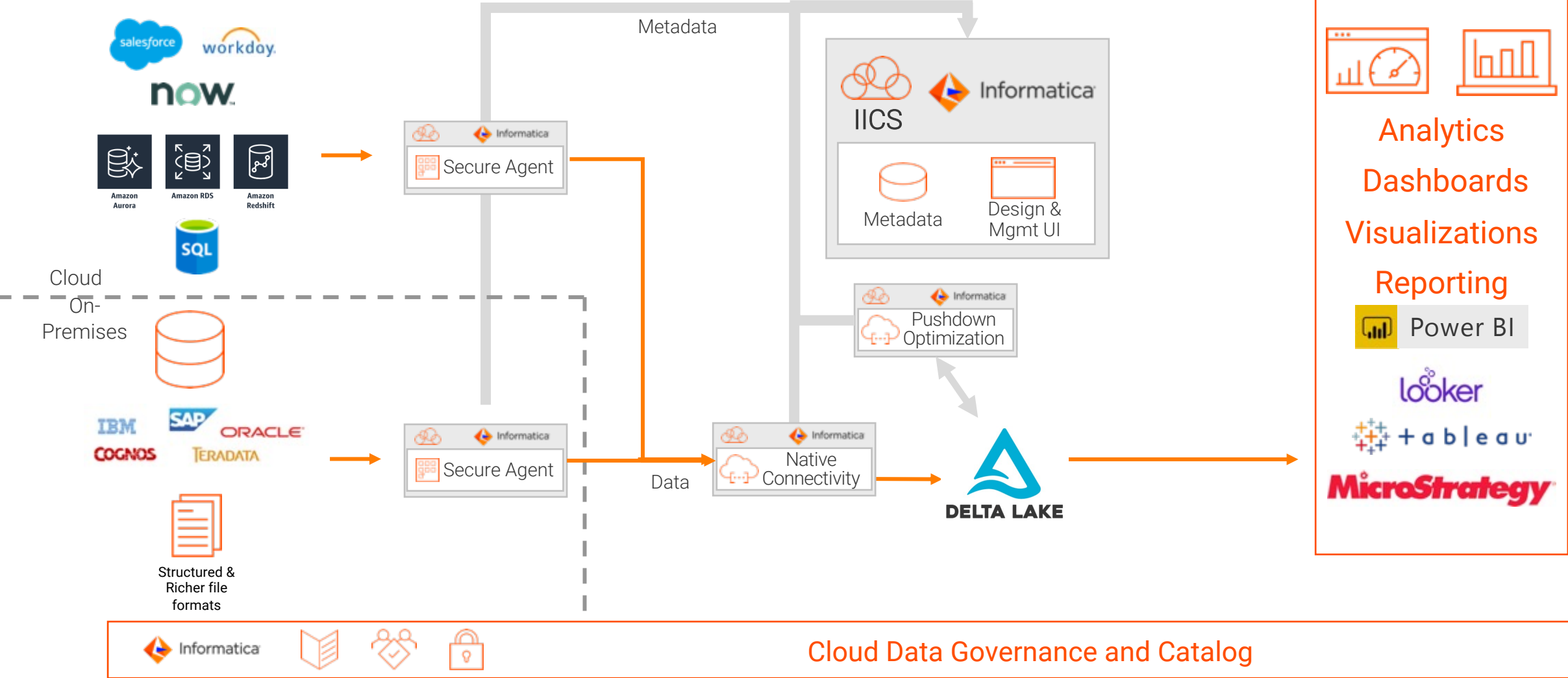


OPEN & EXTENSIBLE

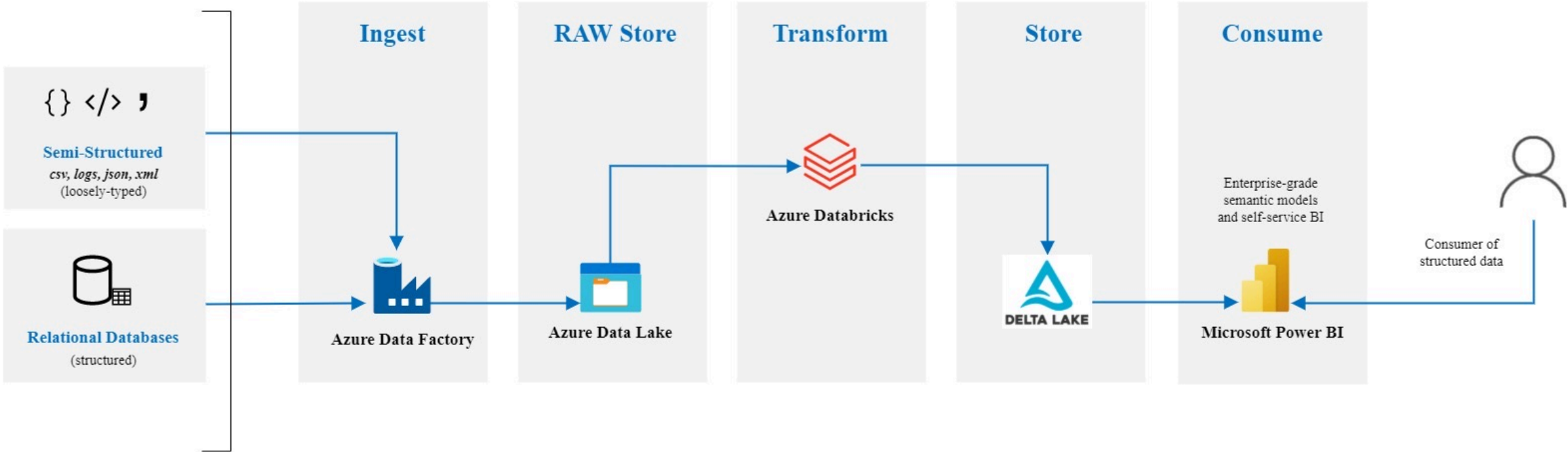
API-based access to metadata repository and functions

Databricks Architecture Patterns

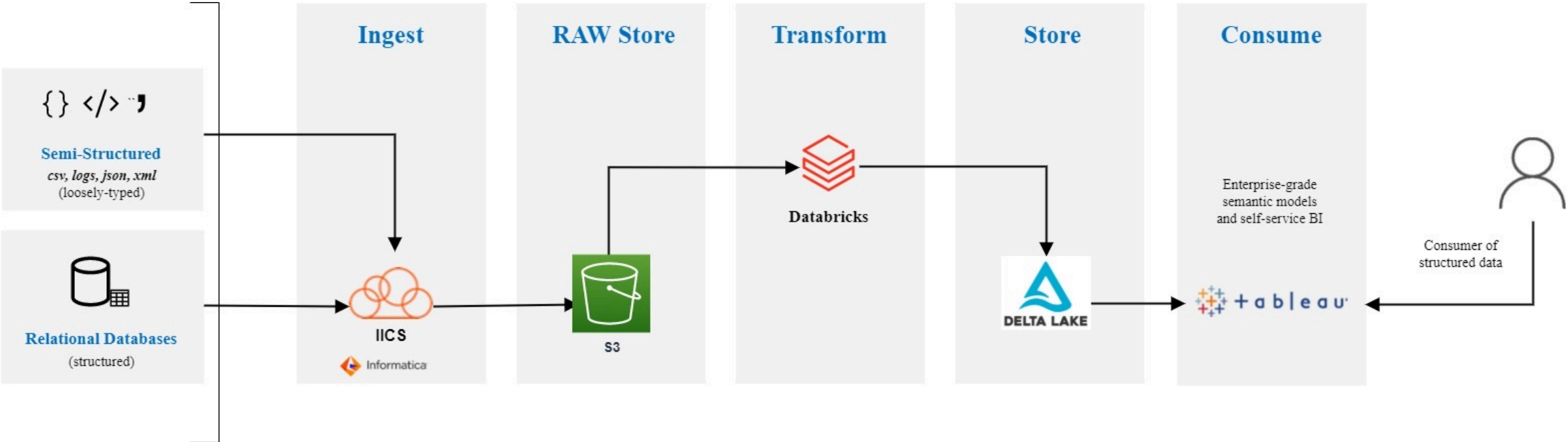
Databricks Delta Lake with IICS



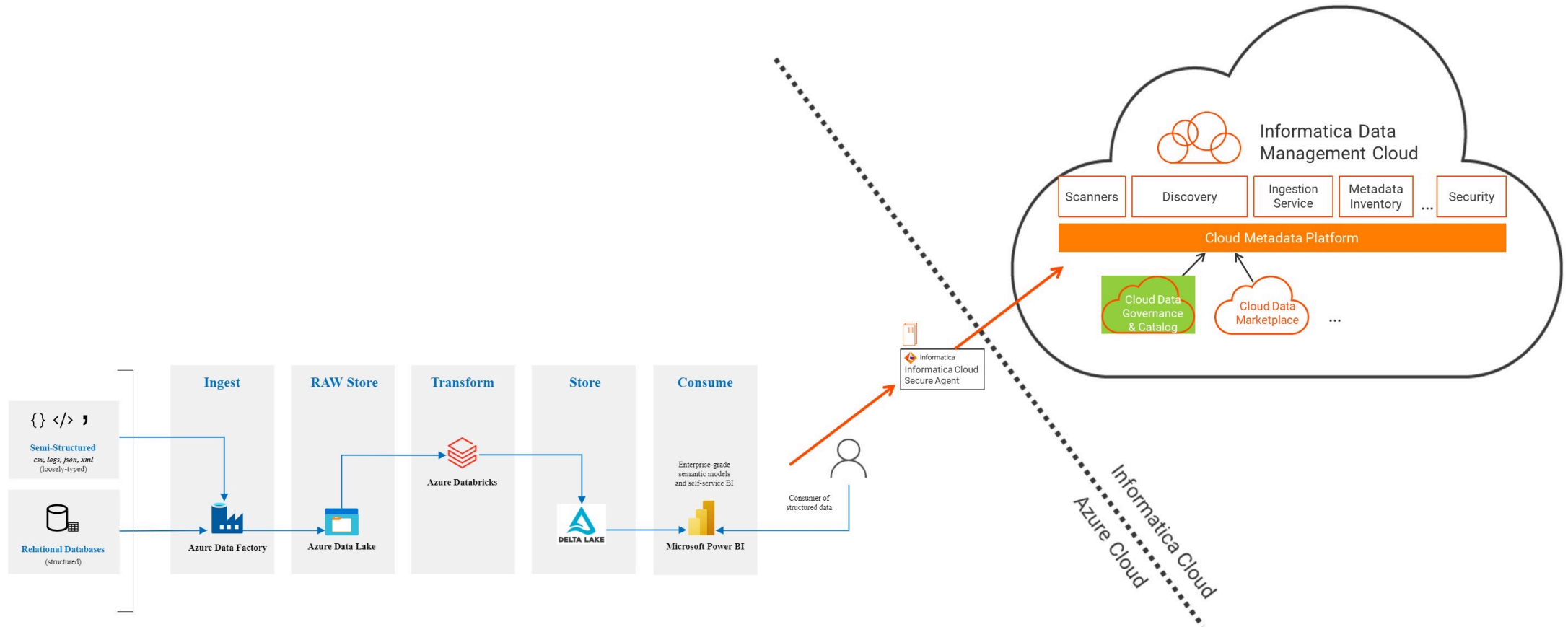
Azure Reference Architecture – Databricks



AWS Reference Architecture – Databricks

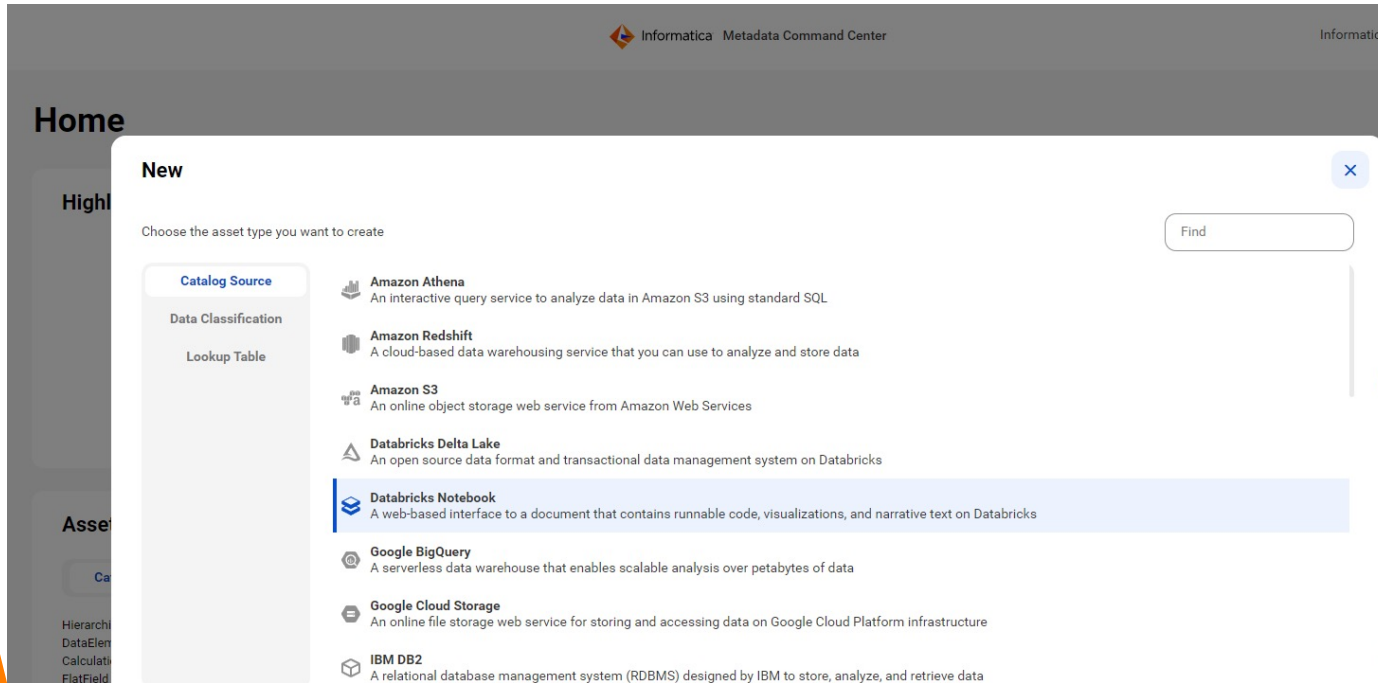


Databricks Delta Lake with CDGC



Databricks Scanners

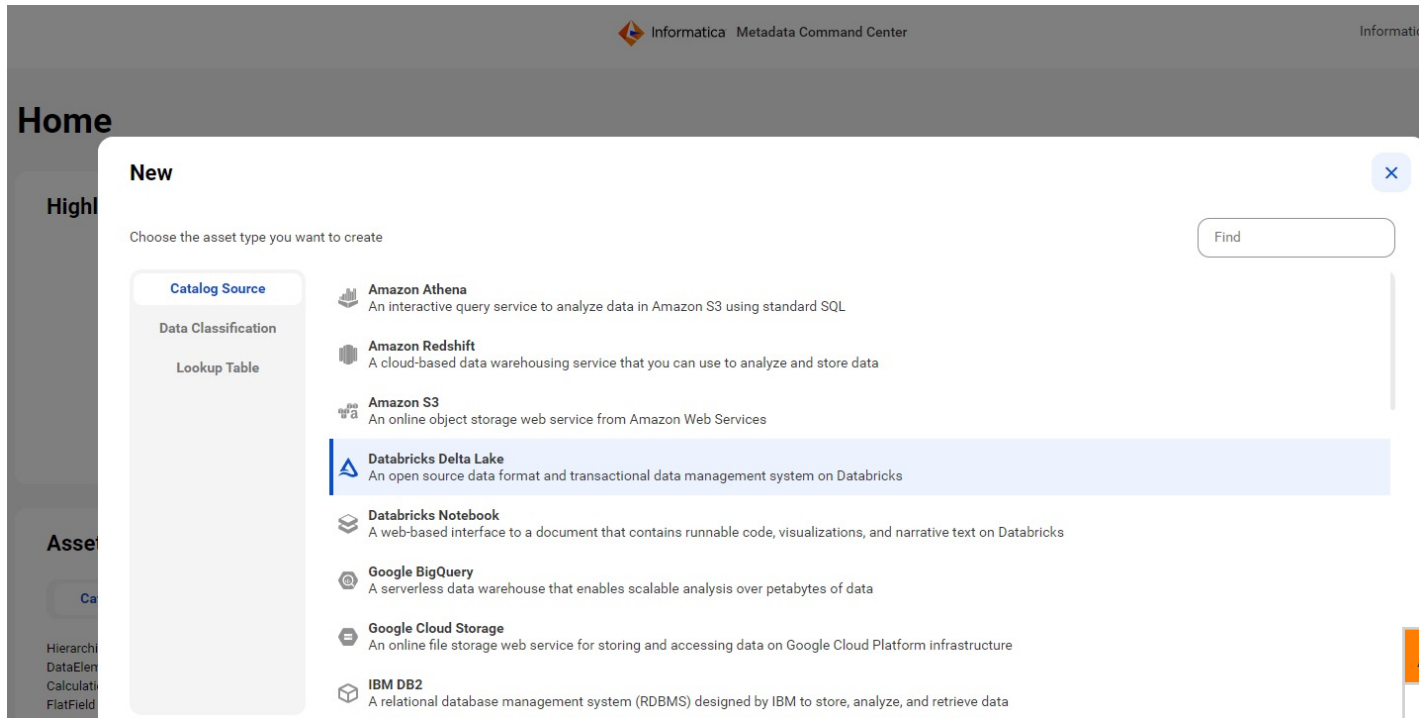
Databricks Notebook Scanner



Databricks Notebook Scanner Config*

Asset	Attributes
Notebook	Notebook name object_type path language object_id
Folder	Folder name object_type path language object_id
Command	Command name Command Content

Databricks Delta Lake Scanner

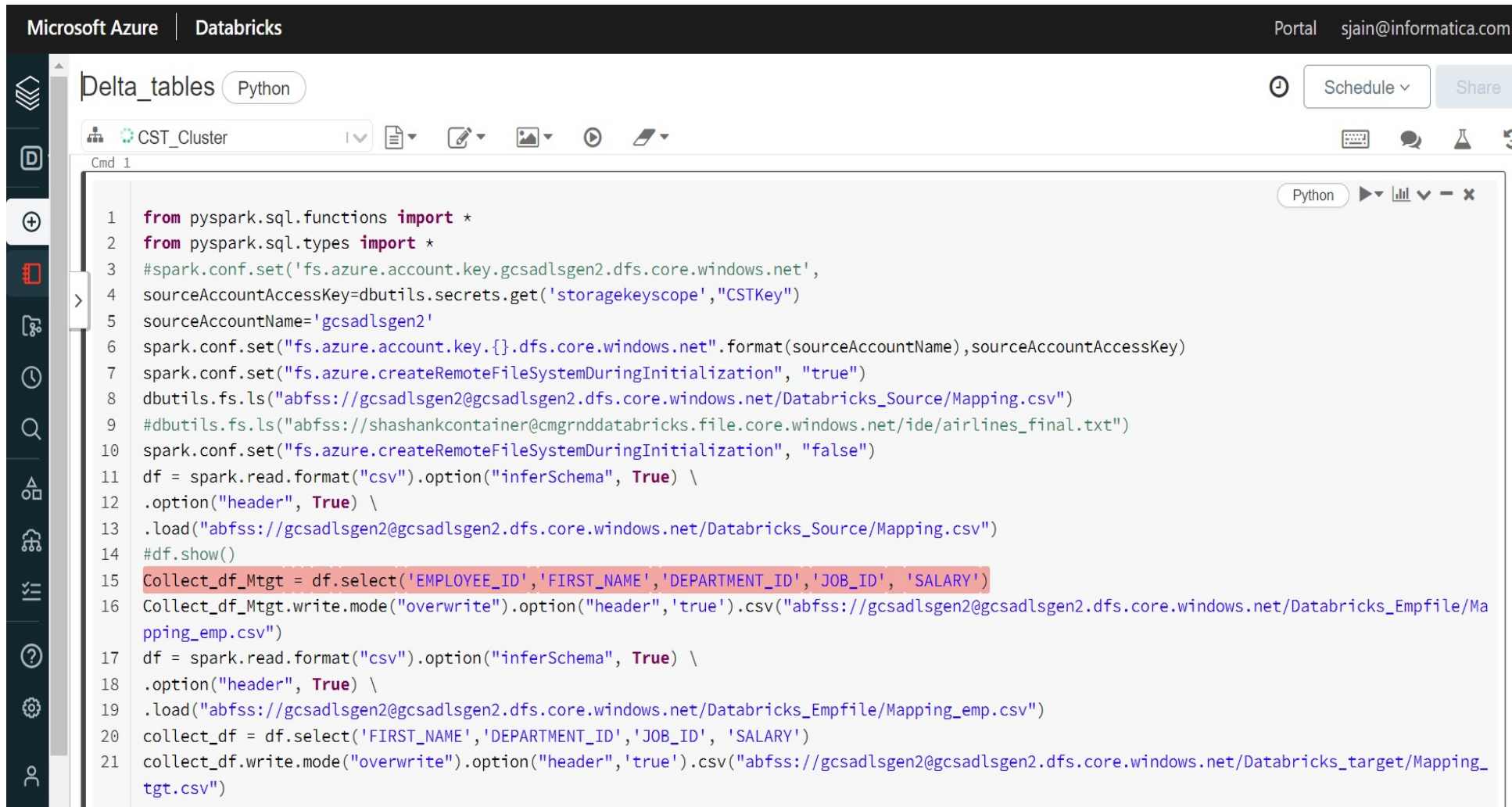


Databricks Delta Table Scanner Config

Asset	Attributes
➤ Database	Name Type Description Path
➤ Schema	
➤ Table	
➤ Column	
➤ View	
➤ View Column	

Databricks Metadata

Databricks Notebook



The screenshot shows a Databricks Notebook interface. The top bar indicates 'Microsoft Azure | Databricks' and 'Portal sjain@informatica.com'. The notebook title is 'Delta_tables' and the language is 'Python'. The code cell contains the following Python code:

```
1 from pyspark.sql.functions import *
2 from pyspark.sql.types import *
3 #spark.conf.set('fs.azure.account.key.gcsadlsgen2.dfs.core.windows.net',
4 sourceAccountAccessKey=dbutils.secrets.get('storagekeyscope','CSTKey')
5 sourceAccountName='gcsadlsgen2'
6 spark.conf.set("fs.azure.account.key.{}.dfs.core.windows.net".format(sourceAccountName),sourceAccountAccessKey)
7 spark.conf.set("fs.azure.createRemoteFileSystemDuringInitialization", "true")
8 dbutils.fs.ls("abfss://gcsadlsgen2@gcsadlsgen2.dfs.core.windows.net/Databricks_Source/Mapping.csv")
9 #dbutils.fs.ls("abfss://shashankcontainer@cmgrnddatabricks.file.core.windows.net/ide/airlines_final.txt")
10 spark.conf.set("fs.azure.createRemoteFileSystemDuringInitialization", "false")
11 df = spark.read.format("csv").option("inferSchema", True) \
12 .option("header", True) \
13 .load("abfss://gcsadlsgen2@gcsadlsgen2.dfs.core.windows.net/Databricks_Source/Mapping.csv")
14 #df.show()
15 Collect_df_Mtgt = df.select('EMPLOYEE_ID', 'FIRST_NAME', 'DEPARTMENT_ID', 'JOB_ID', 'SALARY')
16 Collect_df_Mtgt.write.mode("overwrite").option("header", 'true').csv("abfss://gcsadlsgen2@gcsadlsgen2.dfs.core.windows.net/Databricks_Empfile/Mapping_emp.csv")
17 df = spark.read.format("csv").option("inferSchema", True) \
18 .option("header", True) \
19 .load("abfss://gcsadlsgen2@gcsadlsgen2.dfs.core.windows.net/Databricks_Empfile/Mapping_emp.csv")
20 collect_df = df.select('FIRST_NAME', 'DEPARTMENT_ID', 'JOB_ID', 'SALARY')
21 collect_df.write.mode("overwrite").option("header", 'true').csv("abfss://gcsadlsgen2@gcsadlsgen2.dfs.core.windows.net/Databricks_target/Mapping_tgt.csv")
```

Databricks Notebook

```
Microsoft Azure | Databricks

Delta_tables Python

CST_Cluster

1 from pyspark.sql.functions import *
2 from pyspark.sql.types import *
3 #spark.conf.set('fs.azure.account.key.gcsadlsgen2.dfs.core.windows.net',
4 sourceAccountAccessKey=dbutils.secrets.get('storagekeyscope','CSTKey')
5 sourceAccountName='gcsadlsgen2'
6 spark.conf.set("fs.azure.account.key.{}.dfs.core.windows.net".format(sourceAccountName),sourceAccountAccessKey)
7 spark.conf.set("fs.azure.createRemoteFileSystemDuringInitialization","true")
8 dbutils.fs.ls("abfss://gcsadlsgen2@gcsadlsgen2.dfs.core.windows.net/Databricks_Source/Mapping.csv")
9 spark.conf.set("fs.azure.createRemoteFileSystemDuringInitialization","false")
10 df = spark.read.format("csv").option("inferSchema", True) \
11 .option("header", True) \
12 .load("abfss://gcsadlsgen2@gcsadlsgen2.dfs.core.windows.net/Databricks_Source/Mapping.csv")
13 #df.show()
14 df1 = df.select('EMPLOYEE_ID','FIRST_NAME','DEPARTMENT_ID','JOB_ID','SALARY')
15 df1.write.format("delta").saveAsTable("default.mapping_report1")
16 df = spark.read.format("csv").option("inferSchema", True) \
17 .option("header", True) \
18 .load("abfss://gcsadlsgen2@gcsadlsgen2.dfs.core.windows.net/Databricks_Empfile/Mapping_emp.csv")
19 df2 = df.select('FIRST_NAME','DEPARTMENT_ID','JOB_ID','SALARY')
20 df2.write.format("delta").saveAsTable("default.mapping_report2")
21 df = spark.read.format("csv").option("inferSchema", True) \
22 .option("header", True) \
23 .load("abfss://gcsadlsgen2@gcsadlsgen2.dfs.core.windows.net/Databricks_target/Mapping_tgt.csv")
24 df3 = df.select('FIRST_NAME','DEPARTMENT_ID','JOB_ID','SALARY')
25 df3.write.format("delta").saveAsTable("default.mapping_report3")
```


Databricks Notebook – Overview

The screenshot displays the Databricks Notebook interface for a notebook named "Command 1". The breadcrumb navigation at the top shows the path: new_databricks_notebook / sjain@informatica.com / Delata_tables / . The notebook title "Command 1" is prominently displayed, with a "COMMAND" label below it. To the right, the lifecycle status is "PUBLISHED" and the last updated time is "16 Feb 2022, 06:09". A horizontal menu below the title includes tabs for Overview, Hierarchy, Lineage, Relationships, Data Quality, Stakeholders, Properties, Tickets, and History. The "Properties" tab is currently selected, showing the source code of the notebook. The code is a PySpark script that reads CSV files from Azure storage, processes them, and writes the results back to Azure storage.

Source code:

```
from pyspark.sql.functions import *\nfrom pyspark.sql.types import *\n\nspark.conf.set('fs.azure.account.key.gcsadlsgen2.dfs.core.windows.net', sourceAccountAccessKey=dbutils.secrets.get('storagekeysco...'))\nspark.conf.set('fs.azure.createRemoteFileDuringI...')\nspark.conf.set('fs.azure.writeRemoteFileDuringI...')\n\nndbutils.fs.ls('abfss://gcsadlsgen2@gcsadlsgen2.dfs.core.windows.net/Databricks_Source/Mapping.csv')\nndbutils.fs.ls('abfss://sha...')\n\nndf = spark.read.format('csv').option('inferSchema', True).option('header', True)\n\nload('abfss://gcsadlsgen2@gcsadlsgen2.dfs.core.windows.net/Databricks_Source/Mapping.csv')\nshow()\n\nCollect_df_Mtgt =\ndf.select('EMPLOYEE_ID', 'FIRST_NAME', 'DEPARTMENT_ID', 'JOB_ID',\n          'SALARY')\nCollect_df_Mtgt.write.mode('overwrite').option('header', True).csv('abfss://gcsadlsgen2@gcsadlsgen2.dfs.core.wirldows.net/Dat...')\n\nread('abfss://gcsadlsgen2@gcsadlsgen2.dfs.core.windows.net/Databricks_Source/Mapping.csv')\n\nload('abfss://gcsadlsgen2@gcsadlsgen2.dfs.core.windows.net/Databricks_Empfile/Mapping_emp.csv')\n\nCollect_df =\ndf.select('FIRST_NAME', 'DEPARTMENT_ID', 'JOB_ID', 'SALARY')\n\nCollect_df.write.mode('overwrite').option('header', True).csv('abfss://gcsadlsgen2@gcsadlsgen2.dfs.core.windows.net/Databricks_target/...')
```

Databricks Notebook scan

The screenshot displays the Databricks interface for a dataset named **Mapping.csv** (FLAT FILE). The breadcrumb navigation shows the path: **ADLS_Databricks_Mapping / gcsadlsgen2 / gcsadlsgen2 / Databricks_Source /**. A **Create Dataset** button is visible in the top right. The **LIFECYCLE** is **PUBLISHED**, and it was **LAST UPDATED** on **9 Feb 2022, 09:09**.

The **Lineage** tab is active, showing a dependency graph. The graph starts with the **Mapping.csv** dataset (highlighted in blue) under the **ADLS_Databricks_Mapping** folder. This dataset is used by **Command 1** and **Command 2** within a **new_databricks_notebook**. **Command 1** feeds into **Mapping_tgt.csv** (under **ADLS_databricks_target**) and **Mapping_emp.csv** (under **ADLS_databricks_EMP**). **Command 2** feeds into **mapping_report2**, **mapping_report3**, and **mapping_report1** (all under **Databricks_delta_lake_demo**). **Mapping_tgt.csv** and **Mapping_emp.csv** also feed into **Command 1** and **Command 2** of another **new_databricks_notebook**. Finally, **mapping_report1** feeds into a **databricks_report** (under **PowerBI_Report**), which is used in another **databricks_report** (under **PowerBI_Report**).

Delta Tables Scan

Informatica Data Governance and Catalog

Top Level

Databricks_delta_lake_demo
CATALOG SOURCE

Overview Properties **Hierarchy** Relationships Stakeholders Tickets History

Items (1)

Name	Asset Type
▼ Delta Lake	Database
▼ default	Schema
▶ contacts_tables	Lake house Table
▶ financial_sample_csv	Lake house Table
▶ mapping_report	Lake house Table
▶ mapping_report1	Lake house Table
▶ mapping_report2	Lake house Table
▶ mapping_report3	Lake house Table
▶ tgt_customer	Lake house Table

Microsoft Azure | Databricks

databricks

Data Science &...

+

Create

Workspace

Repos

Recents

Search

Data

Compute

Jobs

Help

Settings

Data

Databases ✓ ▼

Filter Databases

default

Create Table

Tables

Filter Tables

- contacts_tables
- financial_sample_csv
- mapping_report
- mapping_report1
- mapping_report2
- mapping_report3
- tgt_customer
- tgt_fin

DEMO

CDGC – Databricks CASE STUDY



References

- Cloud Data Governance and Catalog: [Click Here](#)
- Introduction and Getting Started: [Click Here](#)
- Governance: [Bulk Upload Business Metadata](#)
- Metadata Command Center:
 - [Catalog Scanner Configuration](#)
 - [Creating Catalog Source](#)
 - [Configure Runtime Environments](#)