Informatica
from Salesforce

24 Mar, 2026

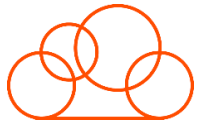
Data Lineage in CDGC: A Practical Guide to Key Categories and Setup Approaches - Part 2

- Puneet Dudeja, Senior Solutions Architect, CSA
- Rebecca South, Principal Customer Success Architect, CSA
- Paul Urban, Senior Solutions Architect, CSA

Where data & AI come to LIFE

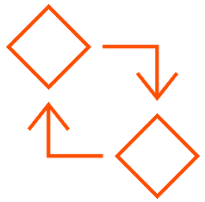
Housekeeping Tips

- 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 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.



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

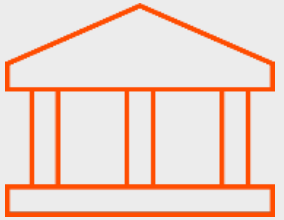
https://success.informatica.com

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

Disclaimer: The information being provided herein is for informational purposes only. The development, release and timing of any Informatica product, service or functionality described herein remain at the sole discretion of Informatica and should not be relied upon in making a purchasing decision. Statements made herein are based on information currently available, which is subject to change. Such statements should not be relied upon as a representation, warranty or commitment to deliver specific products, services or functionality in the future.

Agenda

1. Introduction
2. Webinar Recap : Part 1
3. External Lineage Ingestion
4. Inferred / Linked Lineage and linking methods
5. Custom Lineage
6. Business Lineage
7. Derive business lineage from technical lineage
8. Q&A

Quick Recap - What is Data Lineage

Data Lineage is a visual representation of how data flows from its source(s) through transformation processes to its destination(s). It traces the complete journey of data—showing its origin, every transformation applied, and all downstream dependencies—at multiple levels of granularity.

Why Data Lineage Matters

- ✅ **Impact Analysis**: Understand what breaks if you change a table, column, or process
- ✅ **Root Cause Analysis**: Trace data quality issues back to their source
- ✅ **Compliance & Audit**: Demonstrate data provenance for regulations (GDPR, CCPA, SOX)
- ✅ **Data Governance**: Enforce policies and understand data ownership
- ✅ **Migration & Modernization**: Map dependencies before cloud/platform migrations

Types of Data Lineage supported in CDGC

Technical Lineage

Lineage Type	Source/Trigger	Level	Description	Automation Method
Technical Automated Lineage	ETL Scanners	Dataset/Data Element	Table-to-table and column-to-column flows from scanned metadata (CDI/IICS, PowerCenter, RDBMS)	Automatic via scan + connection assignment
	BI Scanners	Dataset/Data Element	Table to BI Dataset and Column to BI Field from scanned BI source metadata (Tableau, Power BI, MicroStrategy)	Automatic via scan + connection assignment
	DB Scripts / Stored Procedures	Dataset/Data Element	Table-to-table and column-to-column flows from scanned metadata (Stored procedures, script scanners)	Automatic via scan + connection assignment
	AI/ML Models Scanner	Dataset	Table-Model (Databricks, Google Vertex AI)	Automatic via scan + connection assignment
Inferred/Linked Lineage	CLAIRE AI or Rule-based linking	Dataset/Data Element	AI or rule-based matching across catalog sources; can auto-accept based on confidence thresholds	Automatic via Link Catalog Sources jobs

Types of Data Lineage supported in CDGC

Lineage Type	Source/Trigger	Level	Description	Automation Method
External Lineage Ingestion	Databricks Unity Catalog, Apache Atlas, Microsoft Purview	Dataset/Data Element	Imports upstream/downstream lineage from external catalogs when "Lineage from Unity" is enabled	Automatic via scanner configuration
Custom Lineage	Source and target Datasets	Dataset/Data Element	Fill haps where automated lineage can't reach	User curated

Business Lineage

Lineage Type	Source/Trigger	Level	Description	Automation Method
Business Lineage	Business Datasets	Business Dataset/Data Element	System-to-System and dataset-to-dataset providing a high-level view for business stakeholders	Semi-automatic (requires user curation)

Workflow Diagram – Technical Lineage Creation in CDGC

Administrator (Connections)

↓

Test & Save

↓

MCC (Catalog Sources) ← Configure IICS/IDMC Metadata

↓

Run Scans (Sources, Targets, ETL)

↓

Monitor Scan Status

↓

MCC (Connection Assignment)

↓

Assign ETL Connections → Actual Connections

↓

Export Assignment Report (Validate Matches)

↓

CDGC (Lineage Tab)

↓

View Technical Lineage (Catalog/Dataset/Element)

External Lineage Ingestion in CDGC

External lineage ingestion is the process of bringing cross-system lineage into CDGC that spans multiple catalog sources or originates from external platforms—beyond what a single scanner can infer. It enables end-to-end lineage across heterogeneous data ecosystems.

Catalog-of-Catalog Source	Description
Databricks Unity	Ingests metadata from UC system catalog / INFORMATION_SCHEMA
Apache Atlas	Ingests metadata and lineage from Apache Atlas servers (Hive, HBase, HDFS, etc.)
Microsoft Purview	Brings in collections, referenced source objects, and lineage from Microsoft Purview

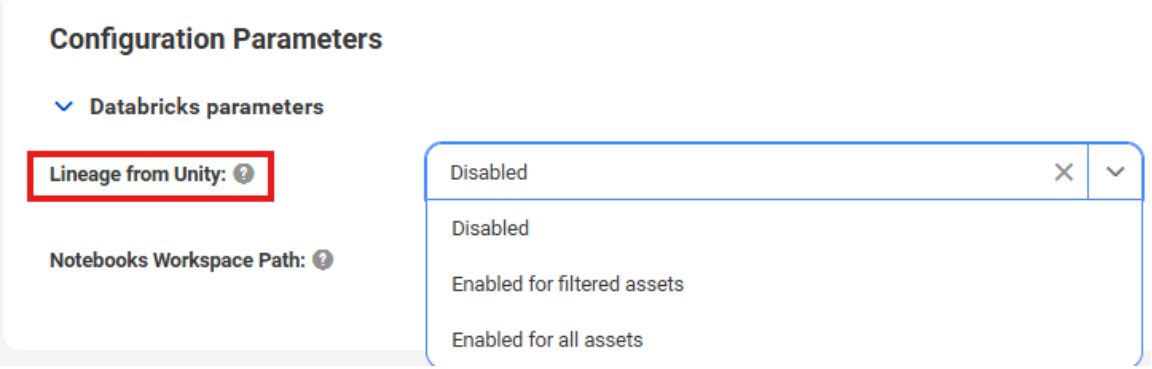
External Lineage Ingestion – Databricks Unity Catalog

Databricks Unity Catalog (UC) is a unified governance layer for data and AI assets across Databricks workspaces. It provides centralized access control, auditing, lineage, and data discovery for tables, views, volumes, models, notebooks, and dashboards.

CDGC calls Databricks API **/2.0/lineage-tracking/table-lineage** to retrieve upstream/downstream graph.

Scanning Databricks in CDGC: Unity Enabled vs. Disabled

CDGC's Databricks scanner can operate in two modes depending on whether Unity Catalog is enabled in your Databricks environment.



Pros of Unity Catalog Enabled

✅ Richer, More Accurate Lineage

•UC lineage APIs capture table/column lineage and notebook context automatically
•No need to parse all notebook code (reduces complexity and errors)
•Lineage retained for up to 1 year (vs. job history limits)

✅ Centralized Governance

•UC access control, audit logs, and compliance features
•Catalog.Schema.Table hierarchy aligns with UC governance model
•Future support for UC tags/glossaries (Roadmap 2026)

✅ Better Performance

•UC system catalog and INFORMATION_SCHEMA are optimized for metadata queries
•Faster scans compared to direct Hive Metastore or cluster queries

✅ Notebook Lineage Without Parsing

•UC lineage includes notebook nodes automatically
•No need to configure variable defaults or WHL files (though still useful for gaps)

Inferred and Linked Lineage in CDGC

Inferred lineage and **linked lineage** are complementary features in CDGC that enable you to **stitch lineage across catalog sources** when scanner-based lineage cannot capture the complete end-to-end data flow.

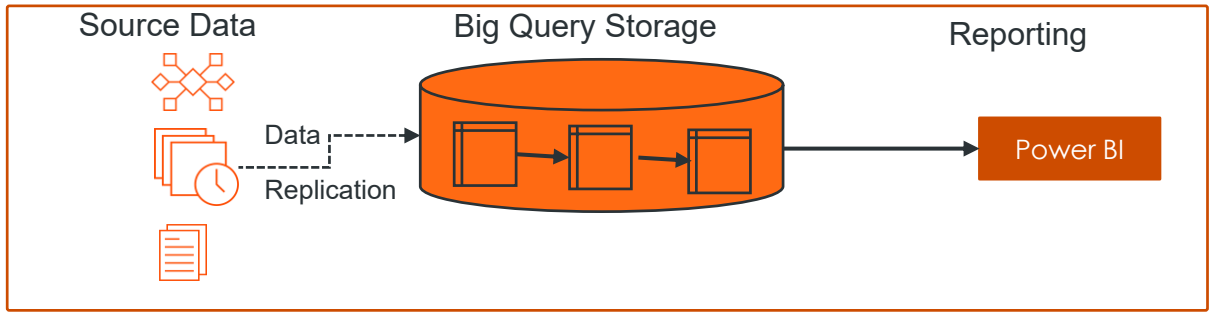
Key Advantages :

- ✅ Close Lineage Gaps When Scanners Cannot See All Hops.
- ✅ Catalog-Source Level Abstraction
- ✅ AI-Powered Discovery with CLAIRE
- ✅ Fine-Grained Control via Rule-Based Linking
- ✅ Governance & Curation Workflow
- ✅ Flexible Access Control

Inferred Lineage – Linking Catalog Sources

Why Use Linked Catalog Sources ?

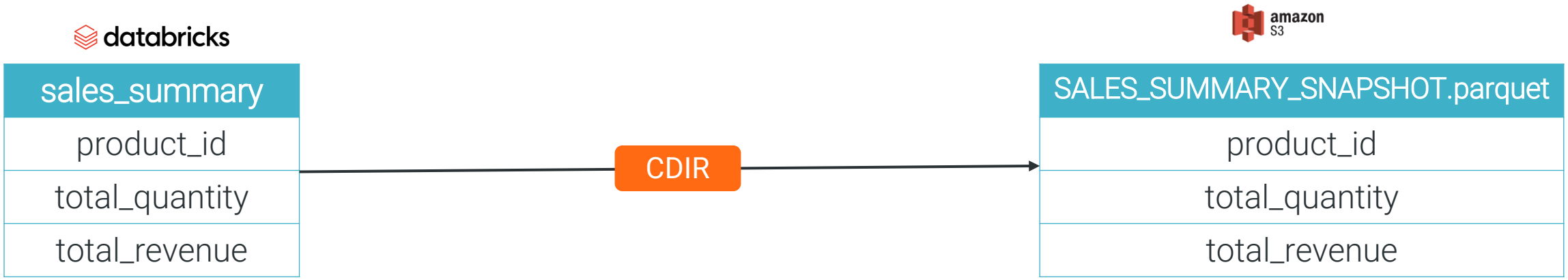
A major medical devices customer consolidated data from multiple databases into their BigQuery storage using data replication software. The customer needed to document the complete data lineage — from the original source through every stage of processing —up to its final consumption in Power BI.



The customer utilized Informatica's native scanning capabilities to perform scans across BigQuery and PowerBI environments. Since the data replication software was not natively scannable by Cloud Data Governance and Catalog (CDGC), the customer leveraged CDGC's rule -based inferred lineage functionality to establish and execute source -to -target lineage rules. Consequently, Informatica automatically inferred one -to -one lineage relationships between source and target tables, both at the dataset and data element levels, to capture end to end lineage

Inferred Lineage – Linking Catalog Sources

How to Use Linked Catalog Sources ?



As metadata extraction from Cloude Data Ingestion and Replication (CDIR) is not supported in CDGC, hence we need to use Inferred lineage

Inferred Lineage - Rule-Based Linking vs. CLAIRE

CDGC offers two methods to link catalog sources and generate lineage:

Aspect	Rule-Based Linking	Automated Linking with CLAIRE
How it works	You define deterministic **Name match** or **Expression** rules to map source/target tables and columns	CLAIRE AI infers lineage based on learned patterns and assigns confidence scores (80–100)
Control	Full manual control; you craft exact matching logic	AI-driven recommendations; you curate and accept/reject
Acceptance	Links are **auto-accepted** by default	Configurable **auto-accept threshold** (default 95%); manual review available
Supported sources	Relational databases **and** file-system sources (e.g., S3, ADLS)	**Relational databases only**; not applicable to file-system sources
Availability	All environments	Currently limited to **AWS US/EMEA** regions
Prerequisites	Metadata scanned; stakeholder permissions	Requires **CLAIRE Generative AI** and **CLAIRE GPT service** enabled; proper role privileges

Inferred Lineage - Rule-Based Linking vs. CLAIRE

When to Use Each Method

Scenario	Recommended Method
Exact name matches or known naming patterns	**Rule-Based** (Expression)
Need to handle case, prefix, or suffix differences	**Rule-Based** (Expression)
File-system sources (S3, ADLS, etc.)	**Rule-Based** (CLAIRE not supported)
Quick coverage across relational databases	**CLAIRE Automated**
High confidence in AI patterns	**CLAIRE Automated** (set threshold 95%+)
Azure/GCP/OCI environments	**Rule-Based** (CLAIRE not available)
No CLAIRE enablement	**Rule-Based**

Inferred Lineage - Rule Based Linking vs. Automated Linking Best Practices

For Rule-Based Linking

1. **Start narrow**: Pilot on a subset of schemas/tables
2. **Constrain both dataset AND data element**: Avoid fan-out and "max links per node" errors
3. **Use expressions for normalization**: Handle case/prefix/suffix differences
4. **Test incrementally**: Validate rules before expanding scope

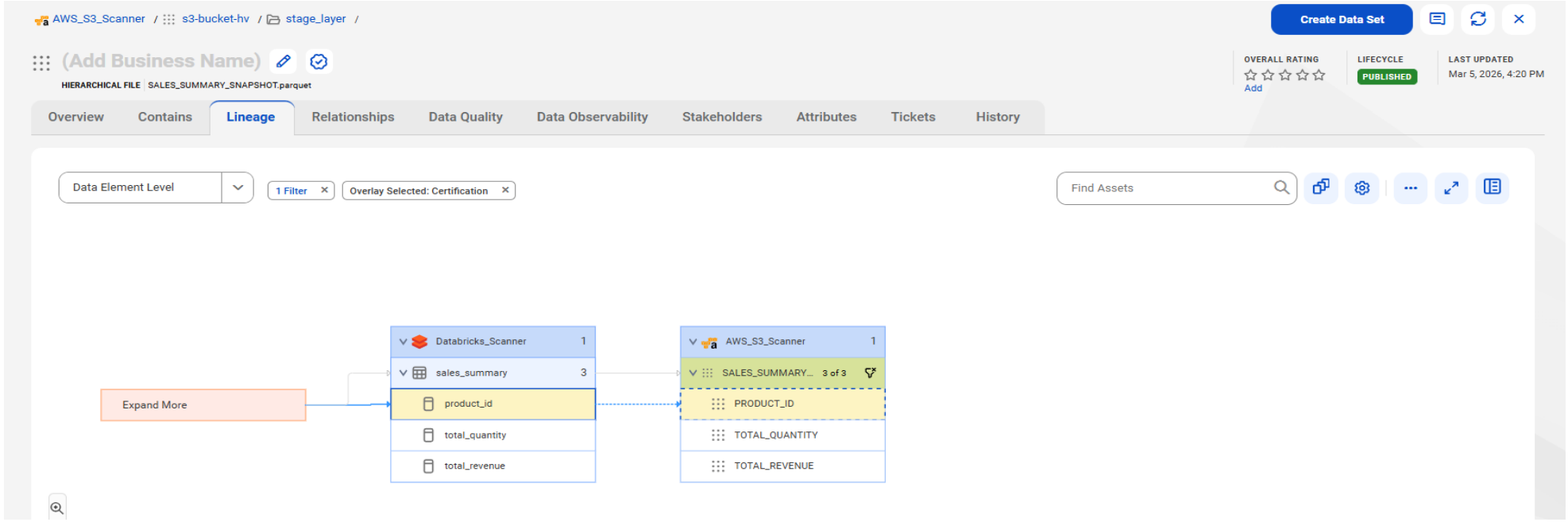
For CLAIRE Automated Linking

1. **Start with high thresholds**: Use 95%+ initially; manually review before relaxing
2. **Curate regularly**: Assign stakeholders to review and accept/reject recommendations
3. **Enable refresh**: Keep links current with new scans
4. **Fall back to rules**: Use rule-based for exact matches where CLAIRE confidence is low

Inferred Lineage - Results

Execute the job to link catalog sources

1.Review results in the Data Catalog. We can now see the additional lineage added onto the end of the lineage from the previous demonstration



Inferred Lineage - Demo

DEMO

Custom Catalog Sources and Lineage

What are Custom Metadata Sources and Lineage?

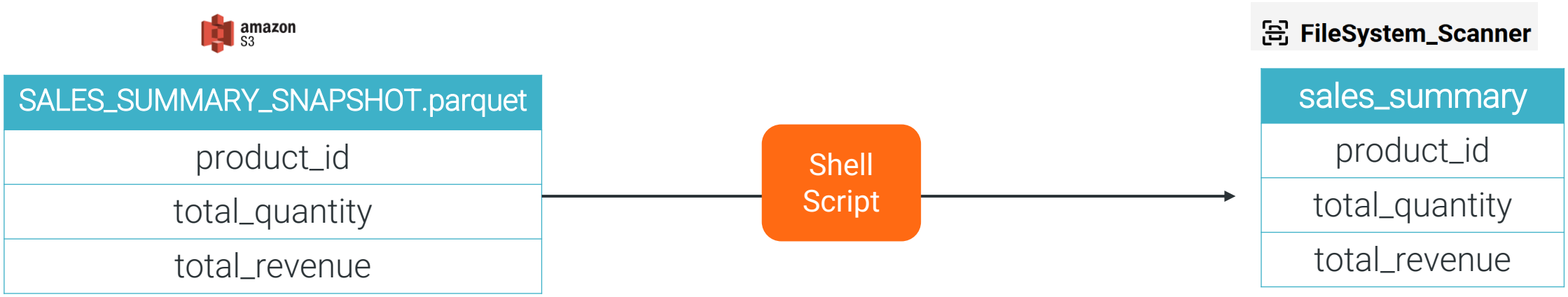
Informatica provides custom metadata integration capabilities for users to create their own custom technical metadata by allowing them to create a custom metadata model, generate templates from that model to load the metadata structures and run a custom scanner to populate the metadata into the catalog.

Why Use Custom Metadata Sources and Lineage?

- There is no out of the box scanner offered by Informatica for the metadata source
- The metadata source is not accessible due to firewalls
- Connectivity challenges prohibit connections to the backend metadata
- The metadata does not exist in any system but is only known by SMEs

Custom Catalog Sources and Lineage DRAFT

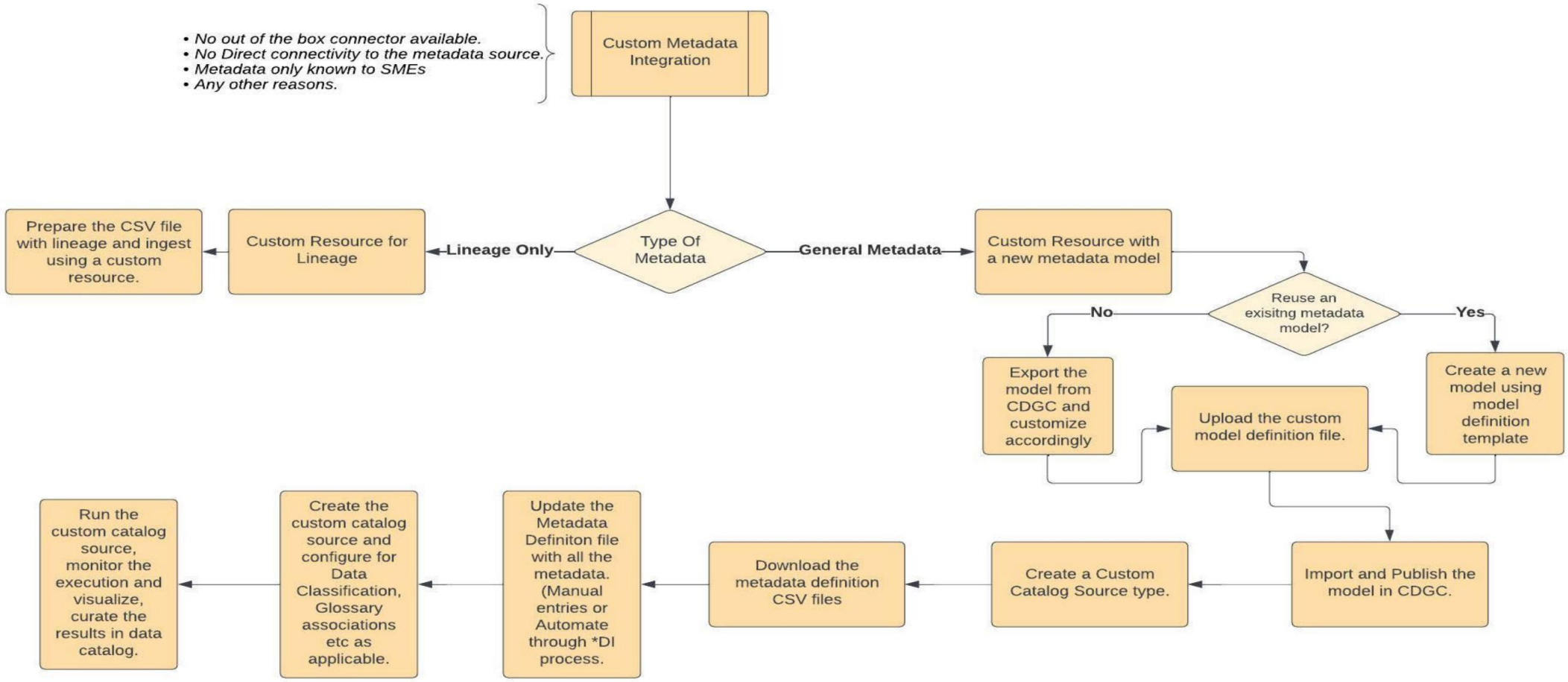
Our use case: we need to extend the data lineage from our S3 bucket to a File



As scanning Shell Scripts is not supported in CDGC, we need to use custom lineage.

Our source and target systems already exist in CDGC and we just will need to create the data lineage links

Custom Metadata Scanner Process



Custom Lineage – Data Preparation

Links.csv -> GenericLinks.zip

Fill in the Reference IDs for the source table and columns
Fill in the Reference IDs for the target table and columns
Fill in the Association for each

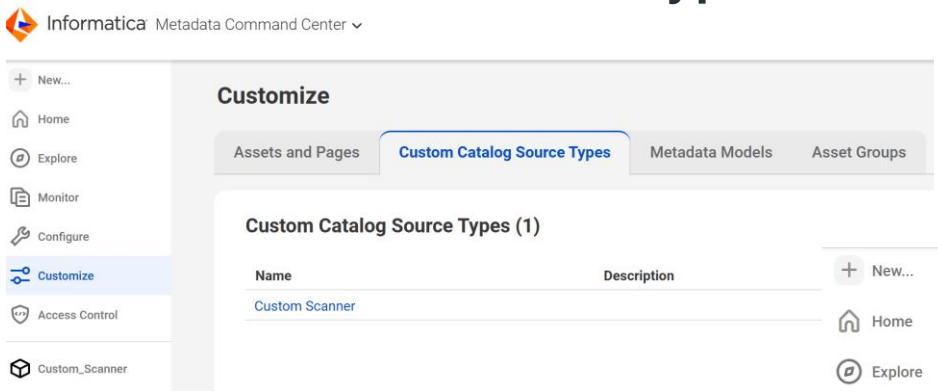
	A	B	C
1	Source	Target	Association
2	68302da6-5c79-3c4b-9a24-55643a847a3e://s3-bucket-hv/stage_layer/SALES_SUMM	1c2f0dd4-6518-3d06-a6f3-cacad12b0199://FileServer/home/idmcagent/infaage	core.DataSetDataFlow
3	68302da6-5c79-3c4b-9a24-55643a847a3e://s3-bucket-hv/stage_layer/SALES_SUMM	1c2f0dd4-6518-3d06-a6f3-cacad12b0199://FileServer/home/idmcagent/infaage	core.DirectionalDataFlow
4	68302da6-5c79-3c4b-9a24-55643a847a3e://s3-bucket-hv/stage_layer/SALES_SUMM	1c2f0dd4-6518-3d06-a6f3-cacad12b0199://FileServer/home/idmcagent/infaage	core.DirectionalDataFlow
5	68302da6-5c79-3c4b-9a24-55643a847a3e://s3-bucket-hv/stage_layer/SALES_SUMM	1c2f0dd4-6518-3d06-a6f3-cacad12b0199://FileServer/home/idmcagent/infaage	core.DirectionalDataFlow
6			

Valid Lineage Association Types (Case-Sensitive)

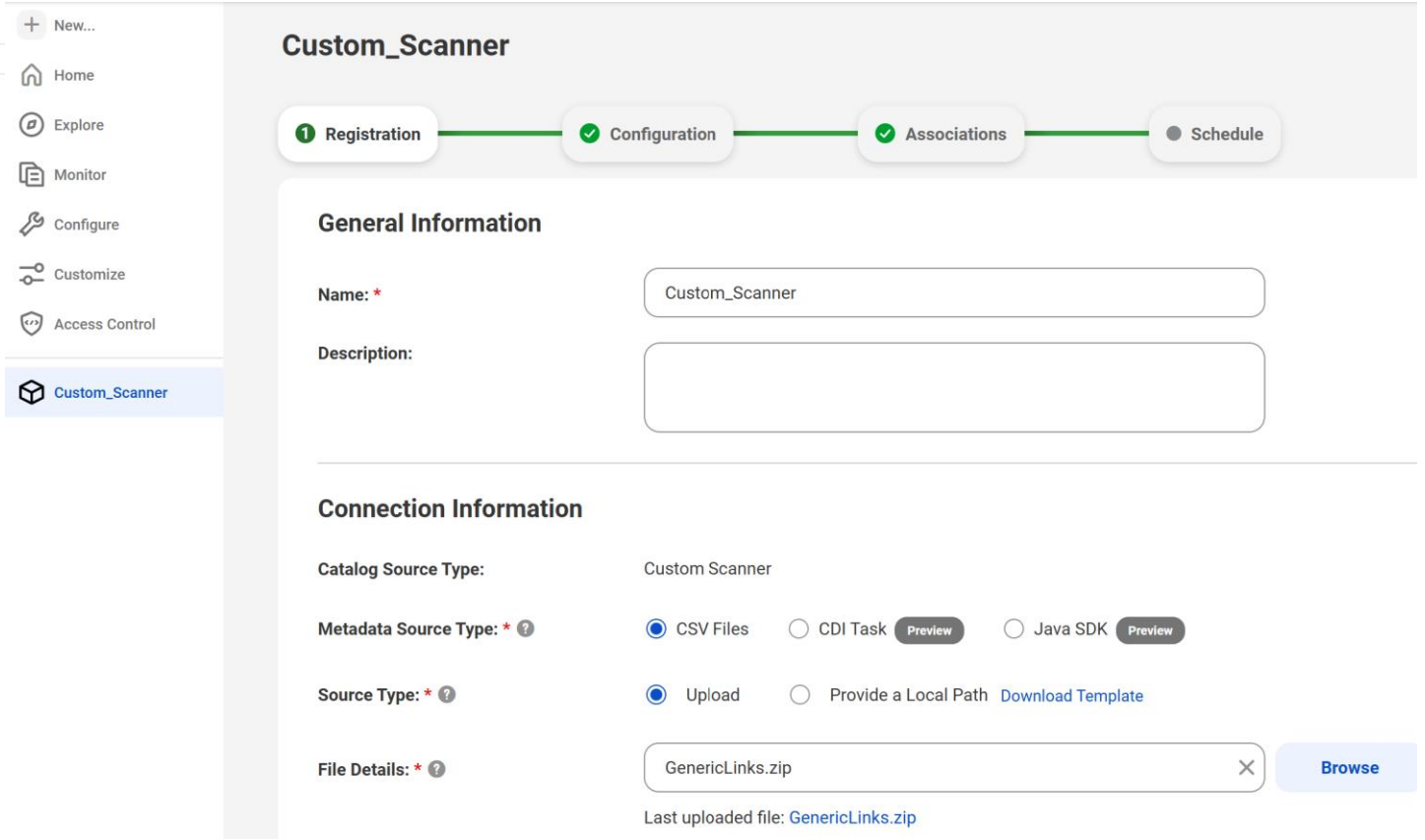
Association Type	Use Case	Example
core.DataSetDataFlow	Table-to-table lineage	customer_master → CUSTOMERS
core.DirectionalDataFlow	Column-to-column lineage	custid → CustomerID
core.ResourceParentChild	Catalog source → Schema	Link custom catalog to schema
core.DataSourceParentChild	Schema → Table	Link schema to table
core.DataSetToDataElementParentship	Table → Column	Link table to column

Custom Lineage – Create Custom Source & Type

Create a custom source type



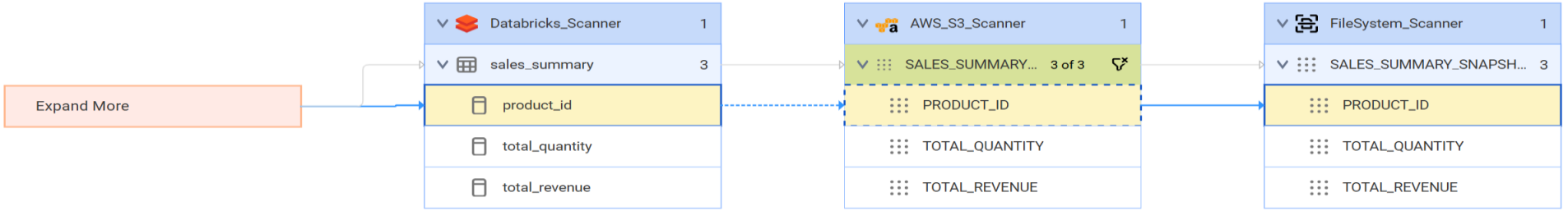
Create a custom resource, save, run



Custom Lineage - Results

Execute the custom scanner to link catalog sources

Review results in the Data Catalog. We can now see the additional lineage added onto the end of the lineage from the Inferred Lineage demonstration



Custom Lineage - Demo

DEMO

Informatica

Business Lineage in Cloud Data Governance and Catalog

Where data & AI come to LIFE

Business Lineage in CDGC

What is Business Lineage?

Business lineage represents the flow of information between Business Data assets.

You can view business lineage for Data Set and System assets. For example, if you are viewing the lineage for a data set, you can see how data flows between data sets and their associated parent system assets. Business analysts typically use business lineage to understand the source of the data and to ensure that the data is coming from a trusted source.

Why Use Business Lineage?

- ☑ To simplify the presentation of lineage across key data sets
- ☑ To showcase lineage which can’t be brought to the catalog thru scanning
- ☑ To establish relationships between assets which are created manually

Technical Lineage vs Business Lineage

How is Business Lineage different from Technical Lineage?

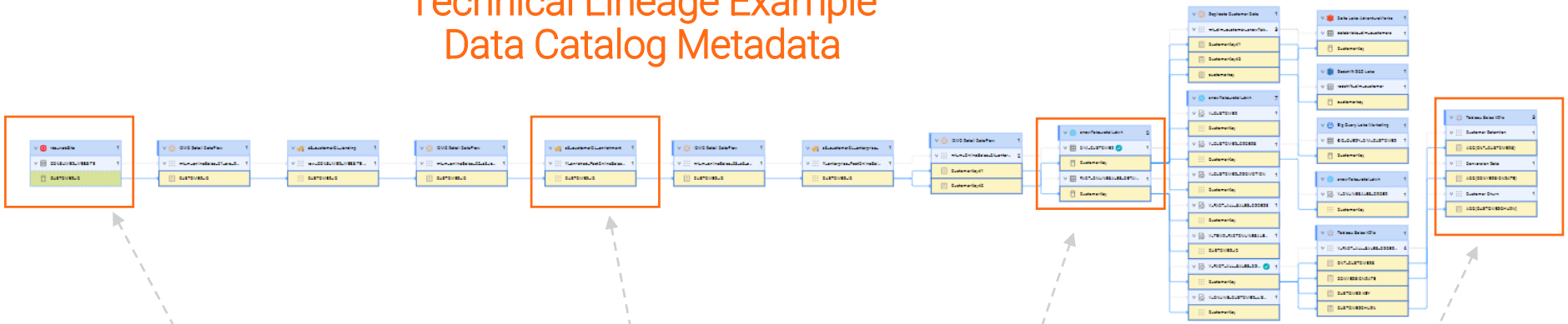
Aspect	Technical Lineage	Business Lineage
Creation	Auto-captured from scanned metadata	Manually curated by mapping business assets to technical assets, business data sets to connected business data sets
Detail Level	Shows detailed data processes (ETL, scripts)	High-level, business-friendly view
Granularity	Catalog source → Technical data set → Data element	System → Business data set → Data element
Audience	Data engineers, architects	Business stakeholders, analysts
Prerequisites	Metadata scans	Requires underlying technical lineage for element-level view - Or - manual lineage

How System & Data Set assets simplify navigation

Technical Lineage will display every “hop” the data element makes based on what the scanner exposes.

This view is ideal for technical developers who need to understand the transformation and location details.

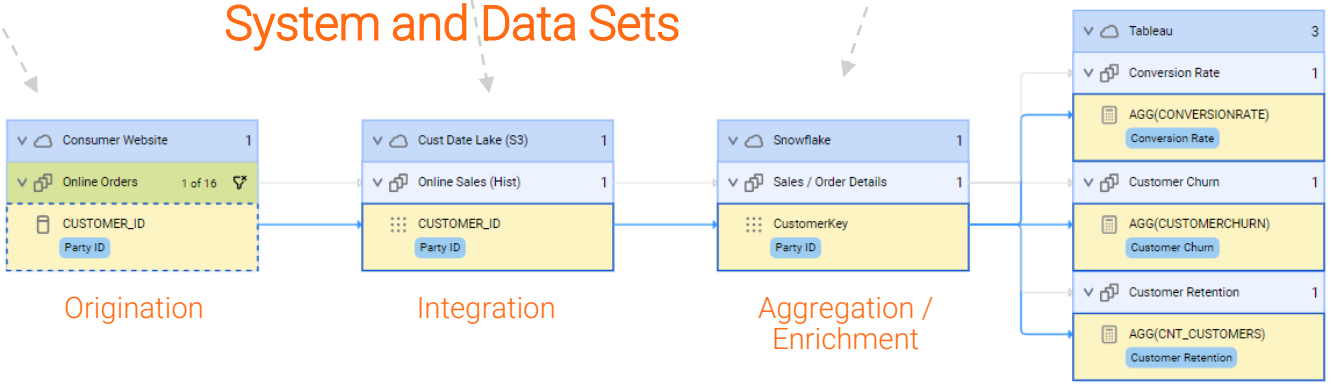
Technical Lineage Example
Data Catalog Metadata



Business Systems and Data Sets do not need to reflect every hop of data. They need only to describe the main data sources that should be under governance processes and definitions.

This view is relevant for Data Consumers, Report Analysts, Business Stewards who don’t need or want to navigate every hop in the data.

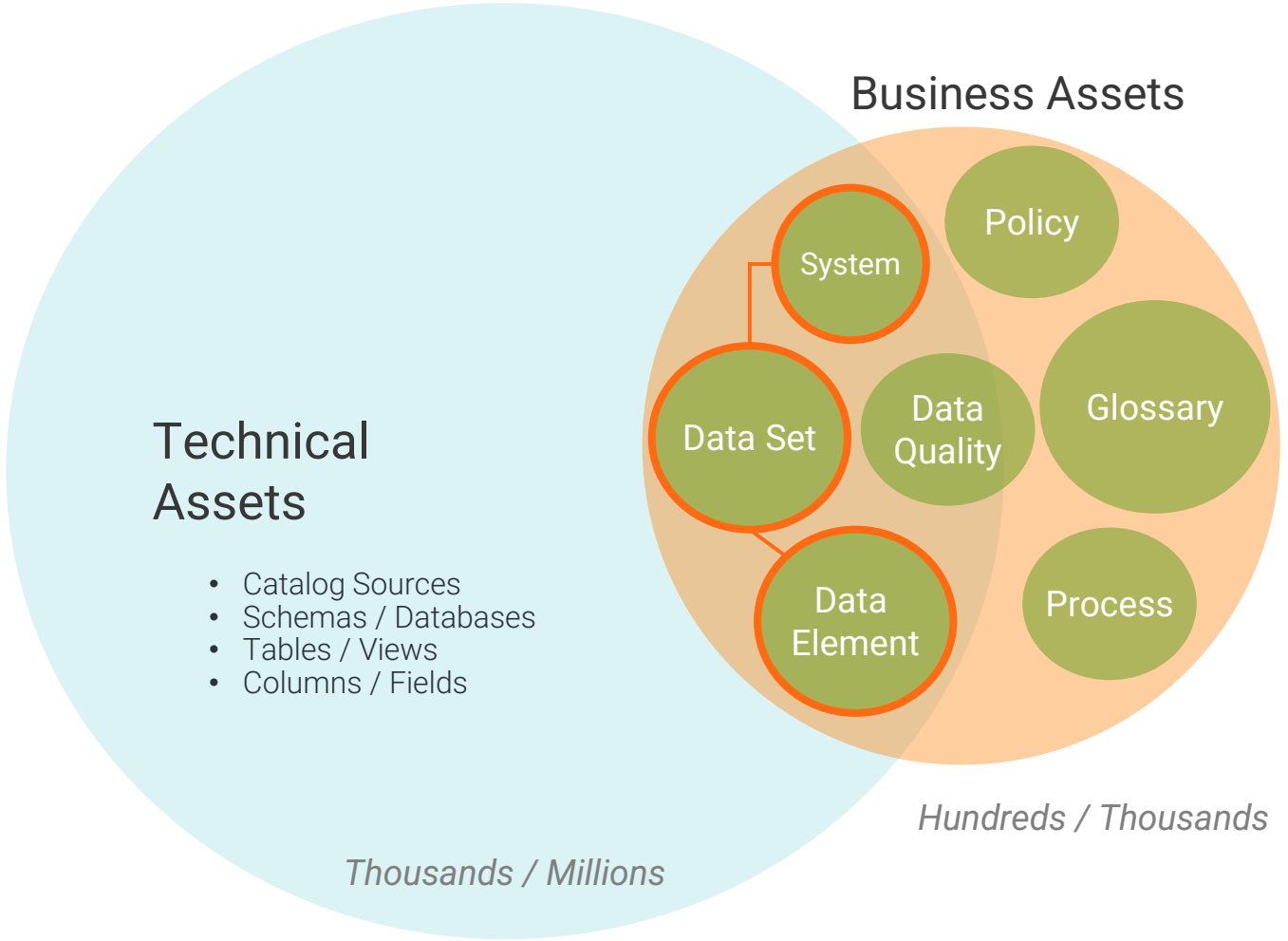
Business Lineage Example
System and Data Sets



Guiding Principles for Business Data Assets

CDGC Business Assets shouldn't cover the full Catalog

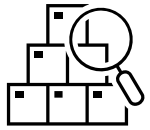
Let the Catalog be source for all technical assets



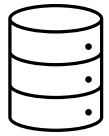
Business Assets should only cover what requires governance or are the relevant key sources

Manually Created Lineage or Business Data

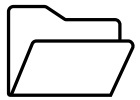
When business metadata is outside the catalog



Scanner is not supported or lineage cannot be scanned



Proprietary/Legacy System



Document location is not accessible

Unstructured or external reports

Business Lineage - Demo

DEMO

Reference Document Links

How to configure Databricks catalog source in CDGC ?

Configure Microsoft Purview scanner

Configure AWS Glue Source scanner

How to link catalog sources to generate lineage in CDGC ?

How to generate links.csv file ?? (Sample csv file available in files attached section on the right)

Create custom catalog source to upload the links.csv file

CDGC Troubleshooting guide for Custom Catalog Source

How to create a business lineage in CDGC ?

Success Accelerator - Technical and Business Lineage in CDGC

Thank You