Exploring Fabric Semantic Link for Power BI Folks

Thank you, partners

KPMG

element61
sharing experience & expertise

PLAINSIGHT

Myrmidons
IT SERVICES

ae
Partner in Digital Excellence

Sparkle

bmatix
Act informed

Cloubis

ilytix | INTELLUS GROUP

datashift

CARTIÈRE

sopra steria

cegeka

JUVO
WWW.JUVO.BE

tillit

randstad digital

Tabular Editor
Better Data Models Faster

U2U

LACO
perform stronger with data

MONIN
Database Managed Services

TriFinance
BEYOND ADVISORY

delaware

redgate

IRiS
POWERING INSIGHT

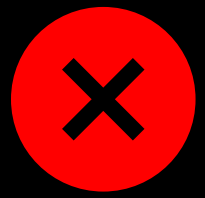
Learning objectives

Fabric	Semantic Link	Query	Document
Understand what Microsoft Fabric is, how this relates to Semantic Link and where it is positioned.	Know exactly what Semantic Link is, how you can use it in your benefit to power your solutions.	Be able to query data and meta data of your semantic model using Semantic Link.	Document your Semantic Models by taking advantage of Semantic Link.

This session is…

- An introduction to Fabric Semantic Link
- Perfectly fitted if you never wrote any line of Python code before

It is not…



- A deep dive on anything Python, Fabric or Notebooks
- Best practices on how you should build data platforms

Expected level:

- Power BI Folks: 300 (advanced but doable 😳)
- Data Engineers: 100 (easy peasy lemon squeezy 🍋)

Marc Lelijveld

Technical Evangelist | Solution Architect
Macaw Netherlands

Microsoft® Most Valuable Professional

linkedin.com/in/MarcLelijveld

Data-Marc.com

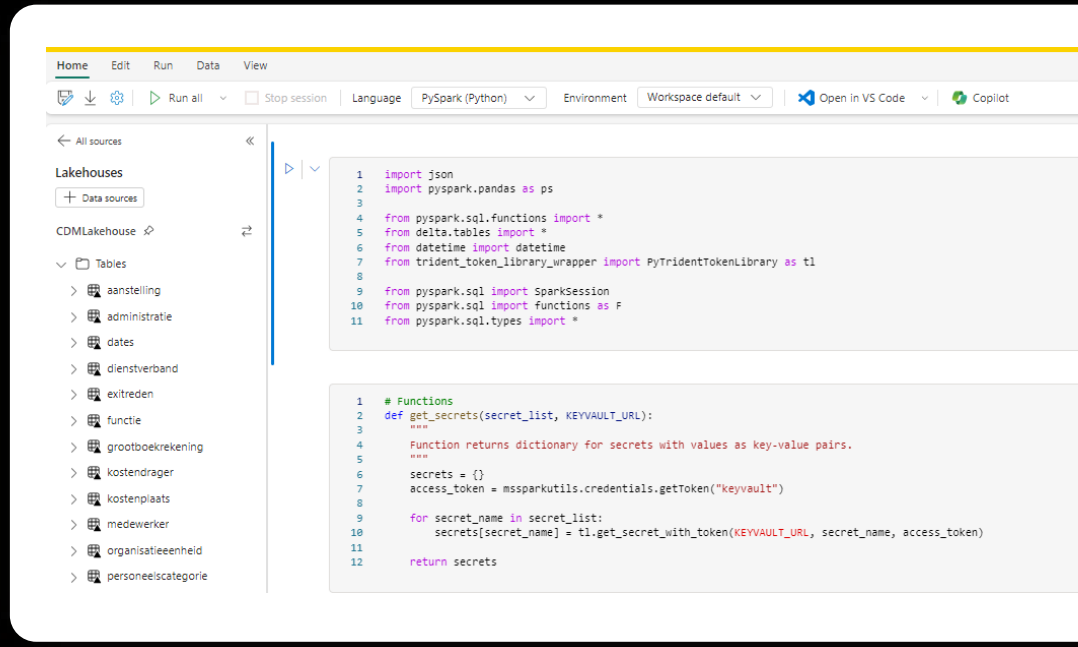
DutchFabricUsergroup.com

FAVORITE STUFF:

What are notebooks?

What are notebooks?

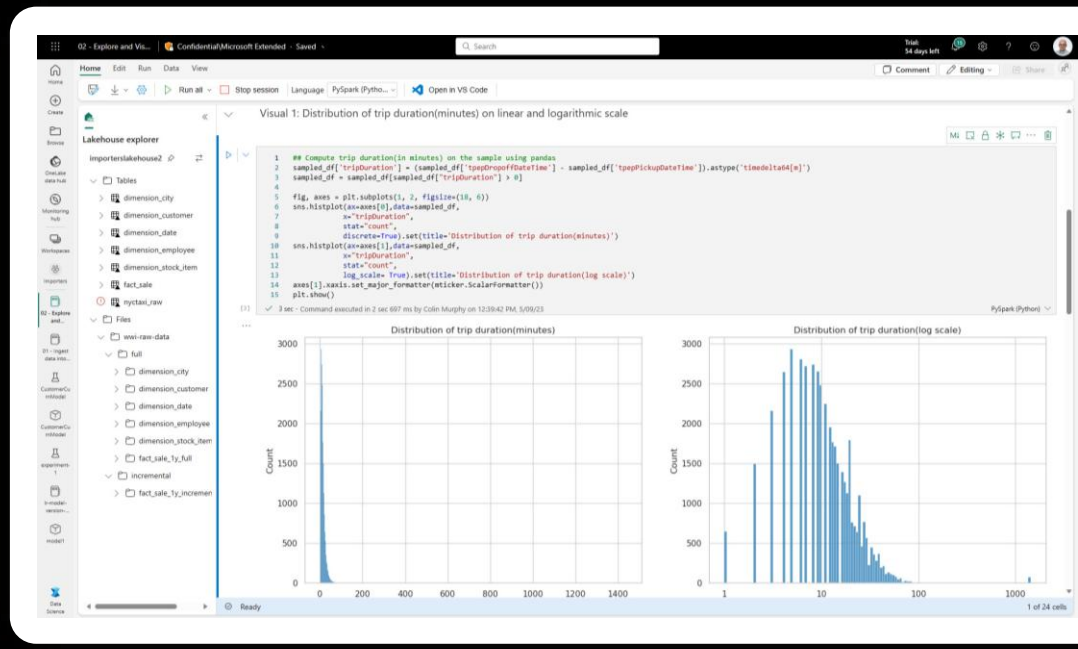
- Code first
- Web-based interface
- Cell based code blocks
- Runs on nodes (part of Fabric capacity)
- Often used languages are Python, Spark & Markdown

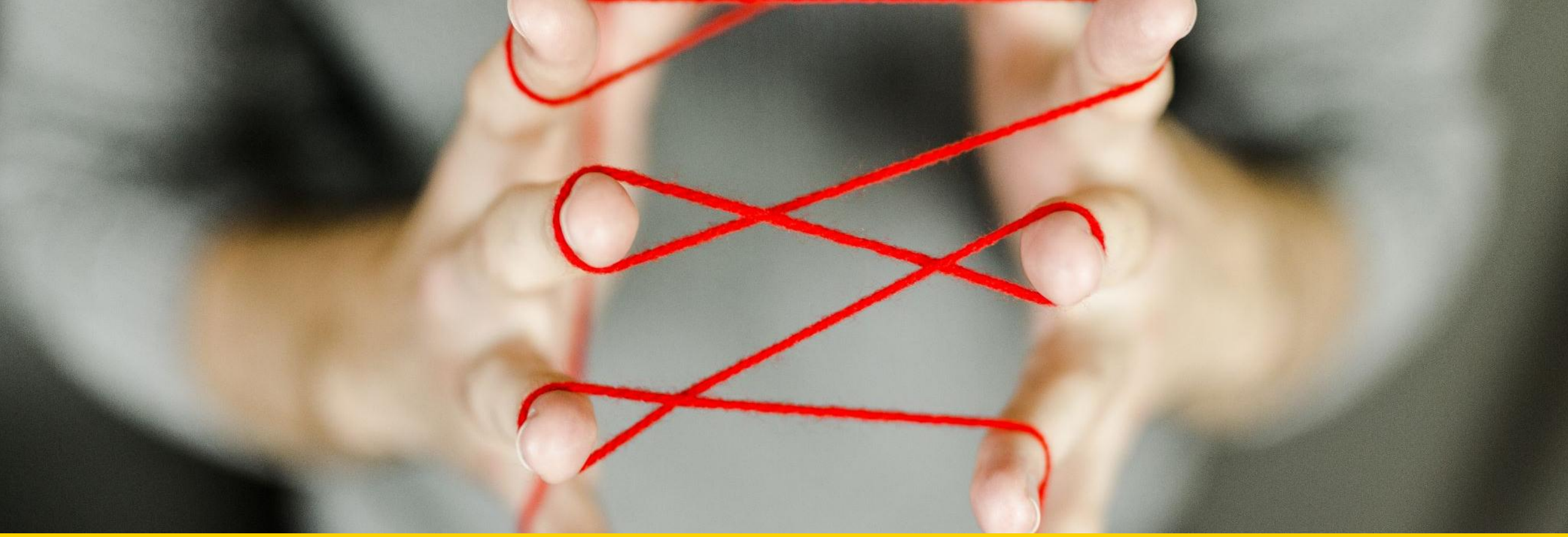


- Used by data engineers for data ingest, prep and transformations
- Used by data scientist for experiments and models

Notebook overview

- Manage your Python and R libraries through in-line installs using commands like %pip install
- Advanced notebook development support with ability to reference notebooks in notebooks, and snapshots for easy troubleshooting
- In context monitoring complete with real time advice and error analysis
- Streamline data prep without giving up the power of reproducibility of Python



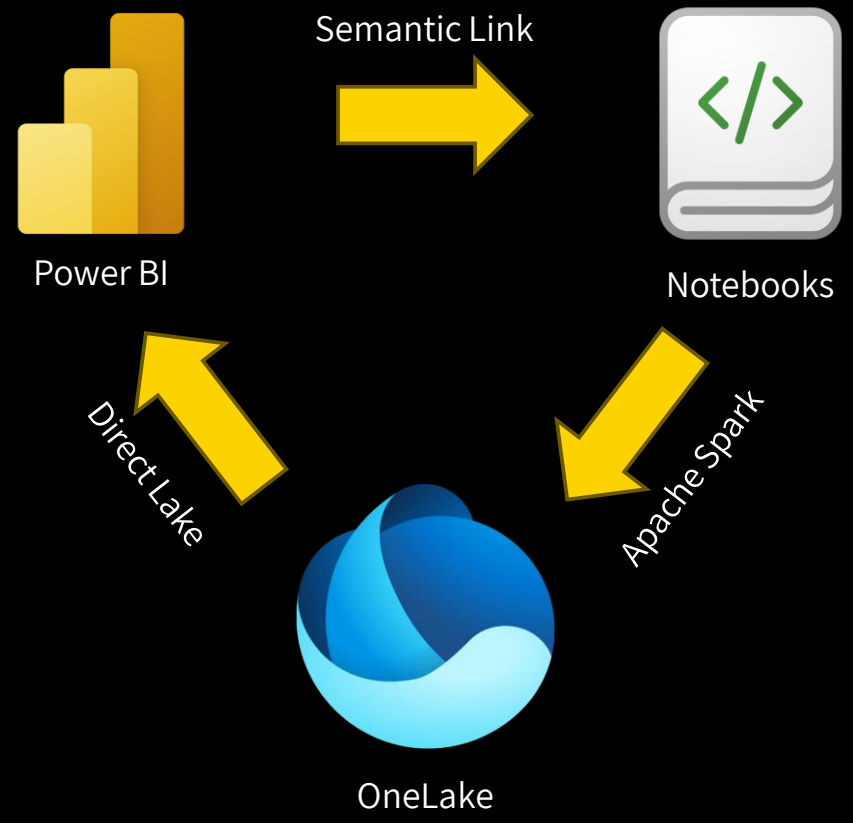


Understanding Semantic Link

What is Semantic Link

Semantic Link is a feature in Microsoft Fabric that allows you to connect from Fabric Notebooks to Power BI Semantic Models.

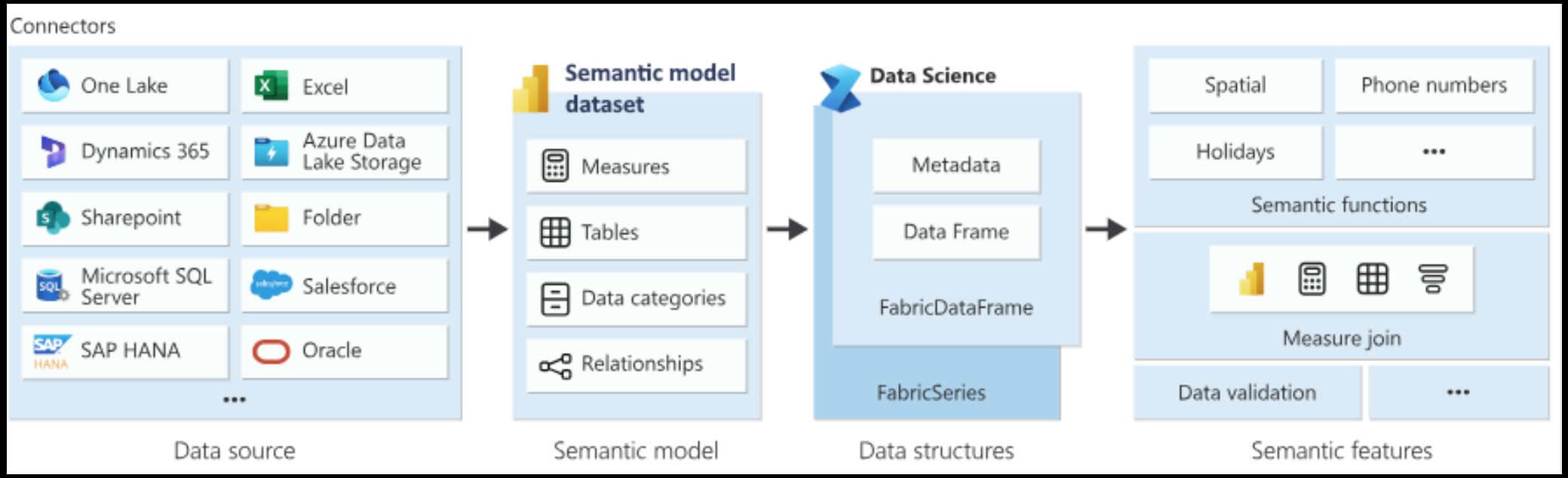
This feature **only** exists and works in Fabric.



- Write DAX, Python, SQL and informative queries directly against your Semantic Model.
- Data as well as metadata.
- Operate and automate tasks like refreshes, partitioning, creation of items etcetera.
- More than only Semantic Models , automation of Fabric tasks in general.
- Open-source extension Semantic Link Labs to do even more!

FabricDataFrame data structure

FabricDataFrame is the data structure of Semantic Link. It makes use of pandas DataFrame and adds meta data such as semantic information and lineage.



Getting started

- The library is part of the default Fabric runtime
- This installs the library which allows us to interact with Semantic Models.

Python Copy

```
import sempy.fabric as fabric

from sempy.relationships import plot_relationship_metadata
from sempy.relationships import find_relationships
from sempy.fabric import list_relationship_violations
```

~~Planning to build multiple notebooks?~~
~~Consider using a custom environment.~~

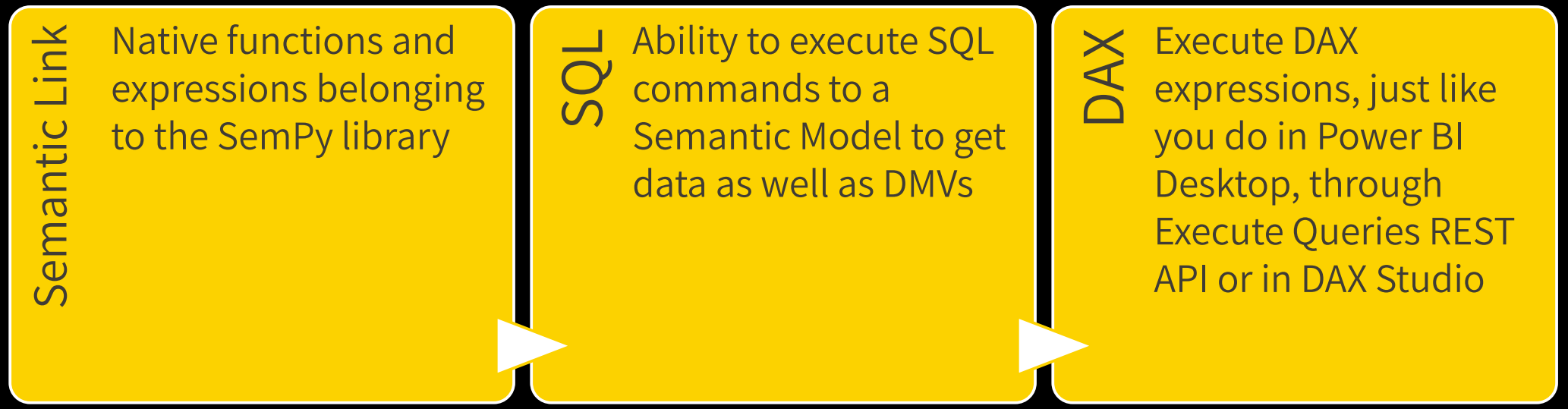
Semantic Link or SemPy?

Both names are used, might cause confusion. But there are differences!

Packages	Description
Semantic-link	Meta-package that depends on all individual Semantic Link packages, easy way to install them all at once.
Semantic-link-sempy	The package that only contains the core Semantic Link functionality
Semantic-link-functions-holidays	A package that contains semantic functions for holidays (determine if a day is a holiday etc.)
Semantic-link-geopandas	Semantic Link packages depending on geopandas to work with spatial data, such as GIS.

Multiple languages and Magic Commands

Notebooks support multiple languages and so does Semantic Link.
There are various options to get your started.



Demo – First exploration of Semantic Link

Query data to use elsewhere

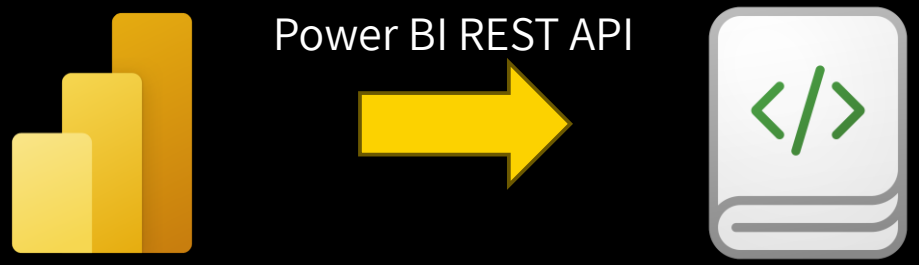
Imagine you invested a lot of time to bring together various data sources. Your Semantic Model turns into a small data warehouse solution. The ability to query data using Semantic Link opens all sorts of new options also to get your data out of Power BI again.

Should you? **NO!**

Power BI is not an ETL tool. Do your data transformations as far upstream as possible – Roche's Maxim

Connectivity

Default uses the Power BI REST API. For certain operations, the XMLA endpoint might be more useful. With *use_xmla=True* you can direct the connection of XMLA.



Python Copy

```
fabric.evaluate_measure(dataset, \
                        measure=["Average Selling Area Size", "Total Stores"], \
                        groupby_columns=["Store[Chain]", "Store[DistrictName]"], \
                        filters={"Store[Territory]": ["PA", "TN", "VA"], "Store[Chain]": ["Lindseys"]}, \
                        use_xmla=True)
```

One level further: Semantic Link Labs

- Migrate import semantic models to Fabric solutions
- Translate model meta data to various languages
- Refresh specified tables/partitions/…
- Run DAX INFO functions to gain information about your model
- Execute Best Practice Analyzer from Fabric notebooks
- Run Vertipaq analyzer to gain statistics from your model
- Identify potential Direct Lake fallbacks

Semantic Link – Use cases

Power BI general

- Documenting Power BI items
- Move Power BI items across workspaces
- Detect broken reports
- Rebind reports
- Set a report theme
- Migration of report-level measures to the semantic model
- Tenant Settings tracking

Semantic Models

- Best Practice Analyzer
- Vertipaq Analyzer
- Semantic model edits (TOM)
- Metadata translations
- Semantic model refresh
- Visualize a refresh
- Semantic model backups
- Run DAX with impersonation
- Manage Query Scale Out

Direct Lake

- Migration to Direct Lake
- Check Direct Lake guardrails
- Warm the cache for Direct Lake
- Analyze Delta tables for Direct Lake
- Fallback to DirectQuery diagnostics
- Update connection of a Direct Lake semantic model

Capacities

- Migration from P SKUs to F SKUs
- Migration from FT SKUs to F SKUs
- Capacity management

And more…

Use cases

Use case 1
Create / maintain measures

- Bulk operations to add measures
- Validate if a measure already exists
- Add measure to multiple semantic models in one go

Demo – Measure management

Use case 2
What about visual calculations?

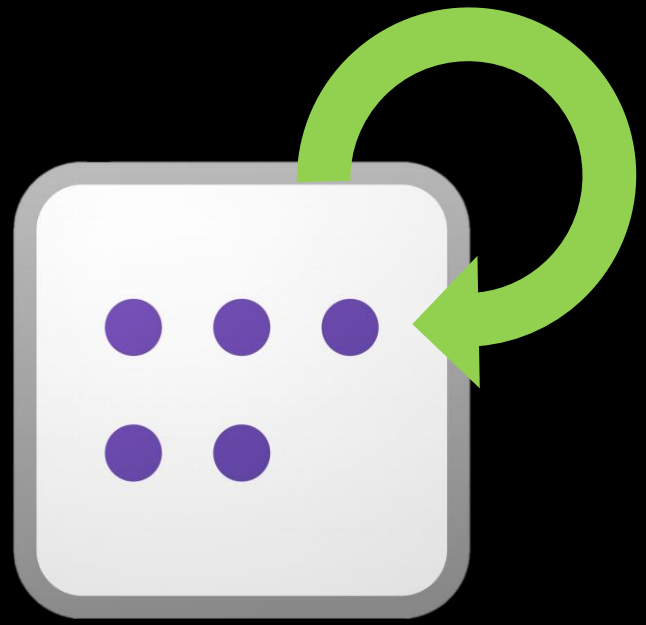
- Visual calculations live do not live in a semantic model
- They are part of the visual matrix
- Visual calculations allow users to deviate from your centrally managed and validated measures
- You may want to track definitions / find where visual calculations are used

Demo – Visual Calcs

Use case 3
Orchestration

Refresh your semantic models via a notebook and trigger dependent actions.

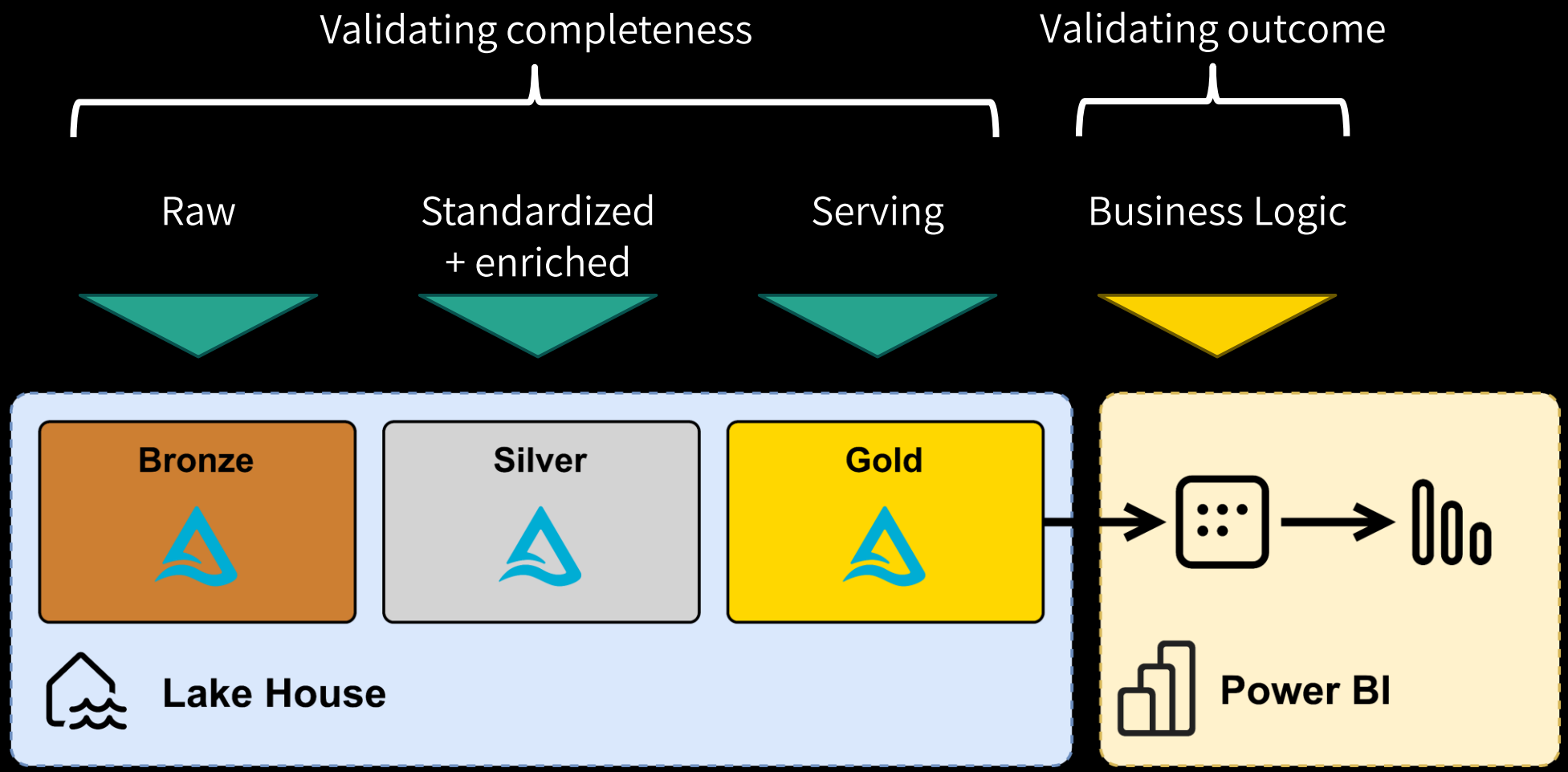
- Given Semantic Link uses the REST APIs, you can orchestrate not only your semantic model refresh, but also trigger upstream dataflows for example
- Refresh individual tables, partitions or reprocess partitions through enhanced refresh API
- Anything else what is possible with the REST API



```
Python                                                    Copy
PowerBIRestClient(token_provider: TokenProvider | None = None)
```

Demo – REST API usage for refreshes

Use case 4
Data validation



Data validation

Data validation based on enrichments (measures) in Semantic Model, therefore different than validation on Gold layer in lakehouse.

- Makes use of public library Great Expectations
- Can be run against Tables, Measures, DMVs
- Sets rules on data types and validates them
  - E.g. Postal codes needs to have 4 numbers and 2 letters (in Dutch system)
  - E.g. Value in column X must be in range between A and B
  - E.g. Units Sold should always be a full number, no decimals

Python Copy

```
suite_measure = context.add_expectation_suite("Retail Measure Suite")
suite_measure.add_expectation(ExpectationConfiguration(
    "expect_column_values_to_be_between",
    {
        "column": "TotalUnits",
        "min_value": 50000
    }
))

context.add_or_update_expectation_suite(expectation_suite=suite_measure)
```

Semantic Model Quality

Mainly depending on DMVs to query Semantic Model meta data and ability to trace dependencies in queries or relationship integrity for example.

E.g. Is the one-side of your relationship, really unique?

```
1   from sempy.relationships import find_relationships, list_relationship_violations
2
3   tables = {
4       "FactInternetSales": fabric.read_table(dataset_name, "FactInternetSales"),
5       "DimDate": fabric.read_table(dataset_name, "DimDate"),
6       "DimProduct": fabric.read_table(dataset_name, "DimProduct"),
7   }
8   relationships = find_relationships(tables)
9
10  list_relationship_violations(tables, relationships)
```

[40] ✓ 2 sec -Command executed in 3 sec 645 ms by Demo User on 1:27:02 PM, 3/01/24

No violations

Multiplicity	From Table	From Column	To Table	To Column	Type	Message

Demo – Model BPA

Use case 5
Warm-up Direct Lake Semantic Models

- When using a Semantic Model with **Direct Lake** storage mode (Fabric only), your data is loaded on-demand to memory.
- This means, only columns that are queried are loaded into the capacities memory. Once loaded, the column will get a **temperature**.
- Every time a new column is loaded, there is a slight **performance** impact since data must be loaded from storage to memory.
- Over time, temperature will drop to zero, and eventually data will **evicted** from memory.

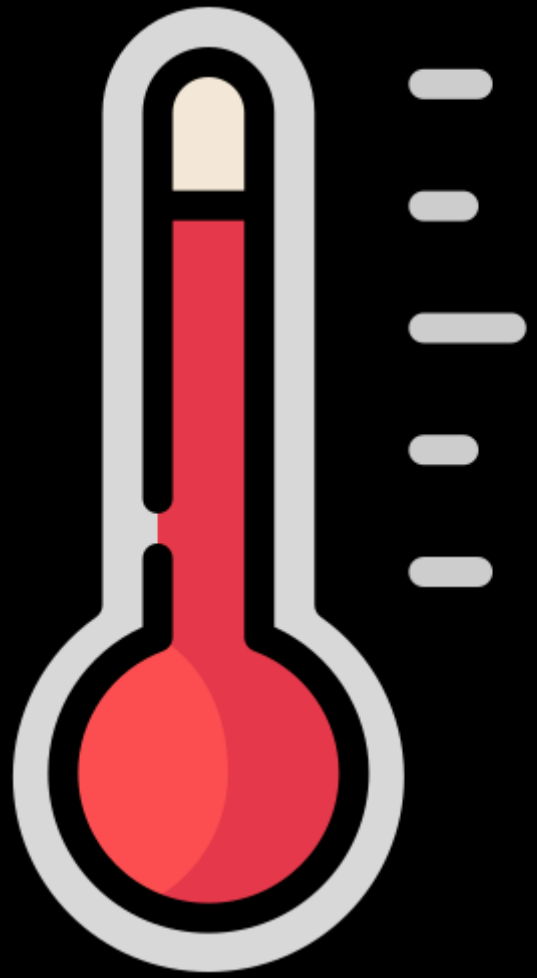
Warm-up Direct Lake Semantic Models

What will be evicted?

Basically, your data will be evicted from active memory, that you want to always have available!

How can you influence that?

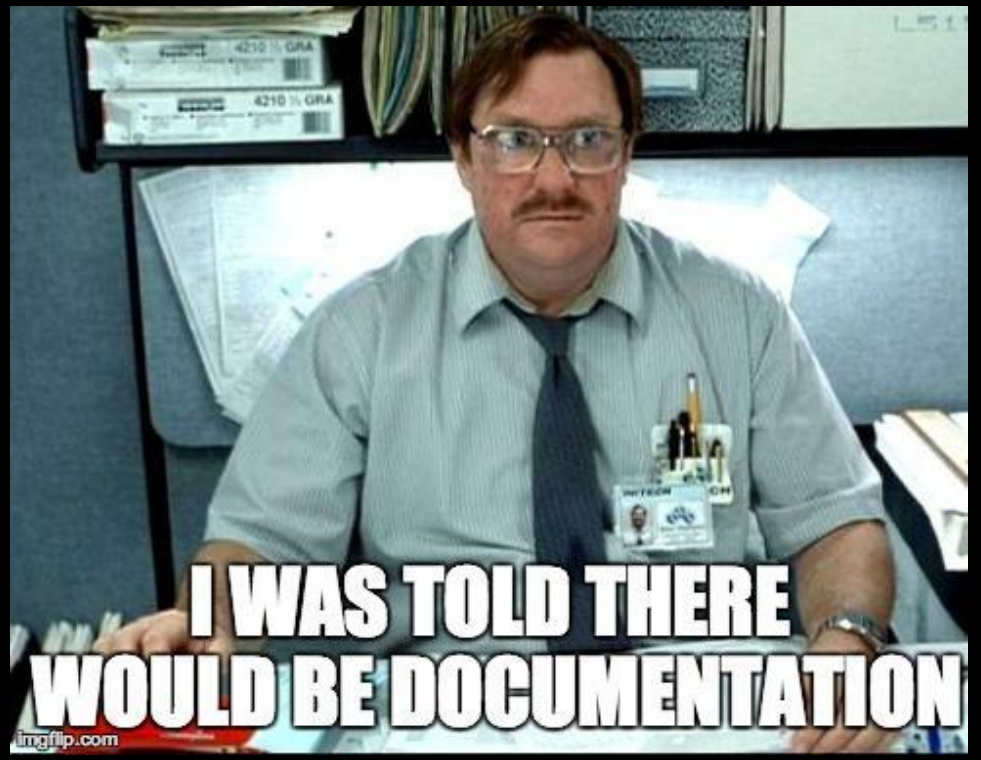
Consider setting up a process (notebook, other automated setup) to pro-actively execute queries to keep certain data **WARM**!



Demo – Direct Lake data warm-up

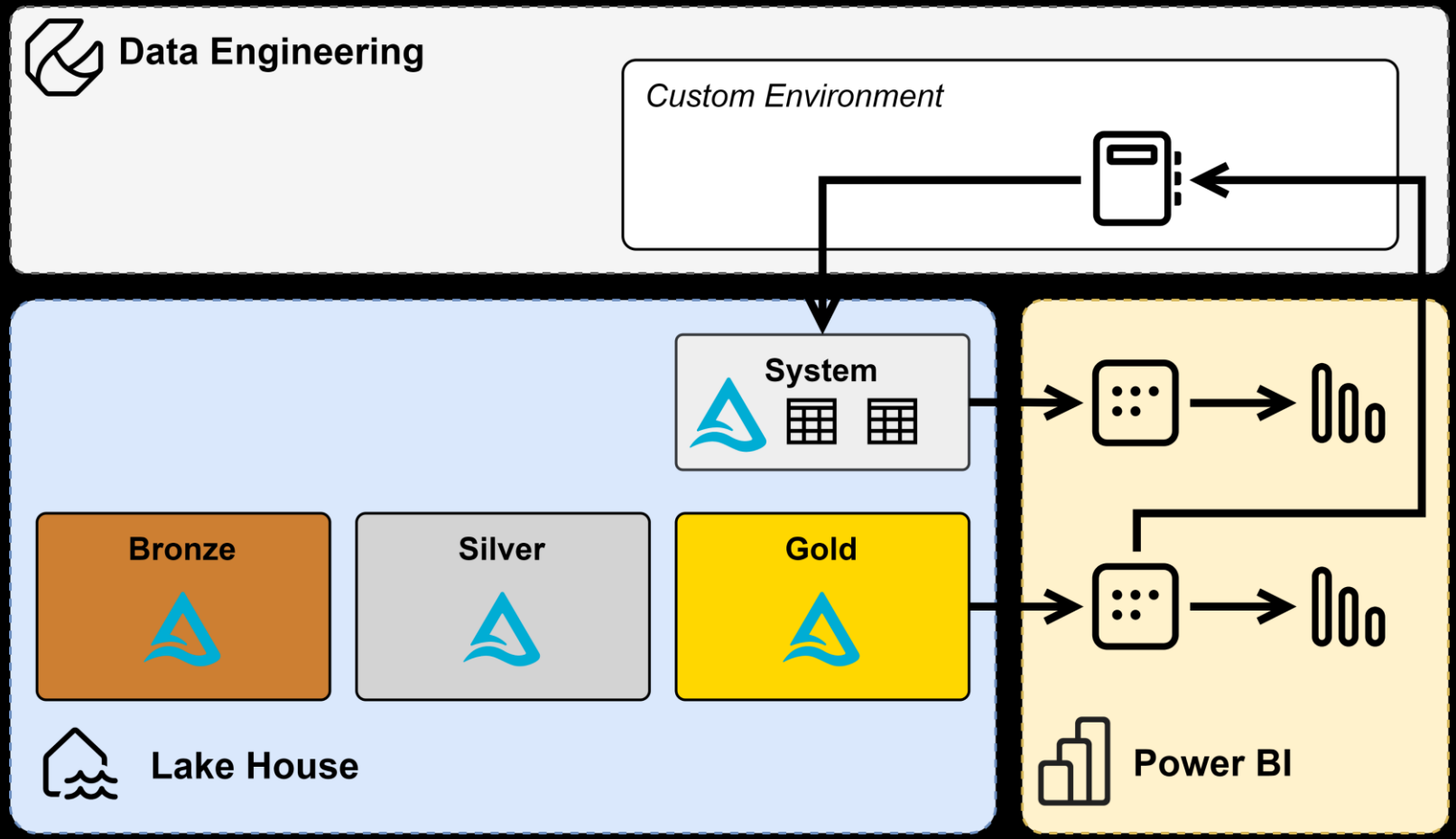
Use case 6
Documenation across multiple semantic models

- Document measures / columns / descriptions to model objects
- Publicly share to those using the model
- Let them understand how the relationships flow



How does it fit in - conceptual overview

Using Semantic-Link to read Meta Data from an existing Sematic Model, which is saved in a Lake House, with a Semantic Model and Report on top to visualize the output.



Demo – Document your solution using Semantic Link

Wrap-up and resources

Keep in mind that…

- Semantic Link as a whole, only works in **Fabric Notebooks**
- Some use cases only apply to **Fabric specific** solutions (warm-up story)
- There is **limited** content available online – but growing!

Wrap up

- Semantic Link allows you to **connect** to your Semantic Model via a Notebook.
- It only works in **Fabric Notebooks**, no limitations on SKUs.
- You can query **data**, **Dynamic Management Views** and any kind of **meta data**.
- Semantic Link allows you to validate both **semantic model quality** and **data quality**.
- Can be used to **extract data** from Power BI to other tools, but **you shouldn't** IMO.
- It perfectly works to **generate documentation** that updates as part of you end-to-end pipeline after refreshes.

Resources

Semantic-Link overview documentation
https://learn.microsoft.com/en-us/fabric/data-science/semantic-link-overview

Semantic-Link Python reference
https://learn.microsoft.com/en-us/python/api/semantic-link-sempy

Fabric Semantic Link and Use Cases by Sandeep Pawar
https://fabric.guru/fabric-semantic-link-and-use-cases

Refreshing (historical) partitions in Power BI Incremental Refresh Semantic Models using Fabric Semantic Link
https://data-marc.com/2024/05/28/dynamically-refreshing-historical-partitions-in-power-bi-incremental-refresh-semantic-models-using-fabric-semantic-link/

These slides
https://github.com/marclelijveld/Slide-decks

Session Feedback 💖

https://bit.ly/dMC2025_SessionFeedback