fabric FEBRUARY

Deep dive into Direct Lake

Mathias Halkjaer & Marc Lelijveld

Thank you to our Fabric February Friends!

twoday

bouvet

sopra steria

DATAMASTERMINDS

ADVANCING ANALYTICS

Evidi

Profisee
Master Data Management

Tabular Editor

KURANT

Fraktal

CluedIn

Dufrain
THE DATA COMPANY

Mathias Halkjær

Principal Architect, Data & Analytics
Fellowmind Denmark

Microsoft® Most Valuable Professional

@halkjaerm

linkedin.com/in/mhalkjaer

Fluxbi.com

FAVORITE STUFF:

Marc Lelijveld

Technical Evangelist | Solution Architect
Macaw Netherlands

Microsoft® Most Valuable Professional

@MarcLelijveld

linkedin.com/in/MarcLelijveld

Data-Marc.com

FAVORITE STUFF:

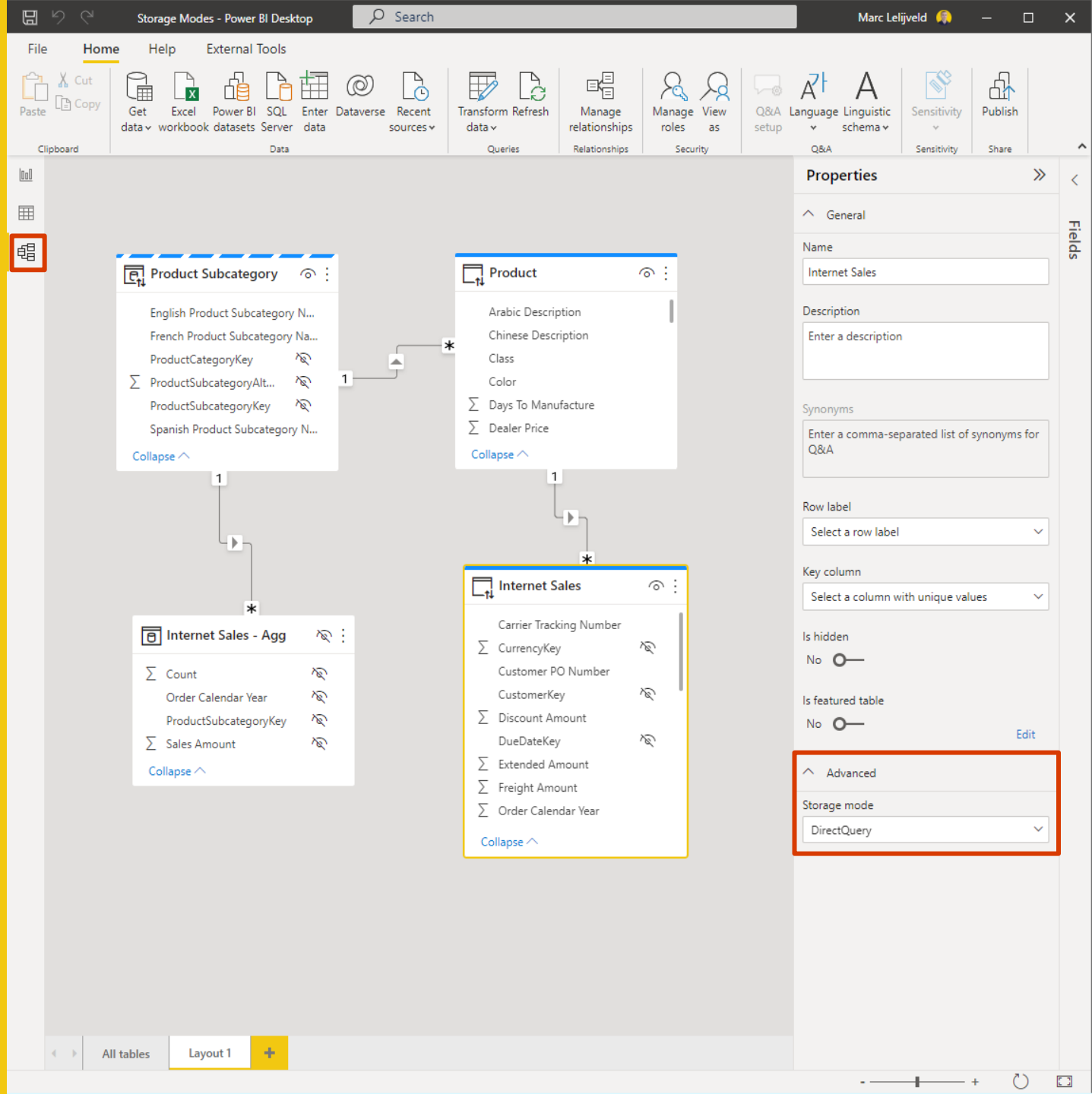
Storage modes

Mathias Halkjær - Marc Lelijveld

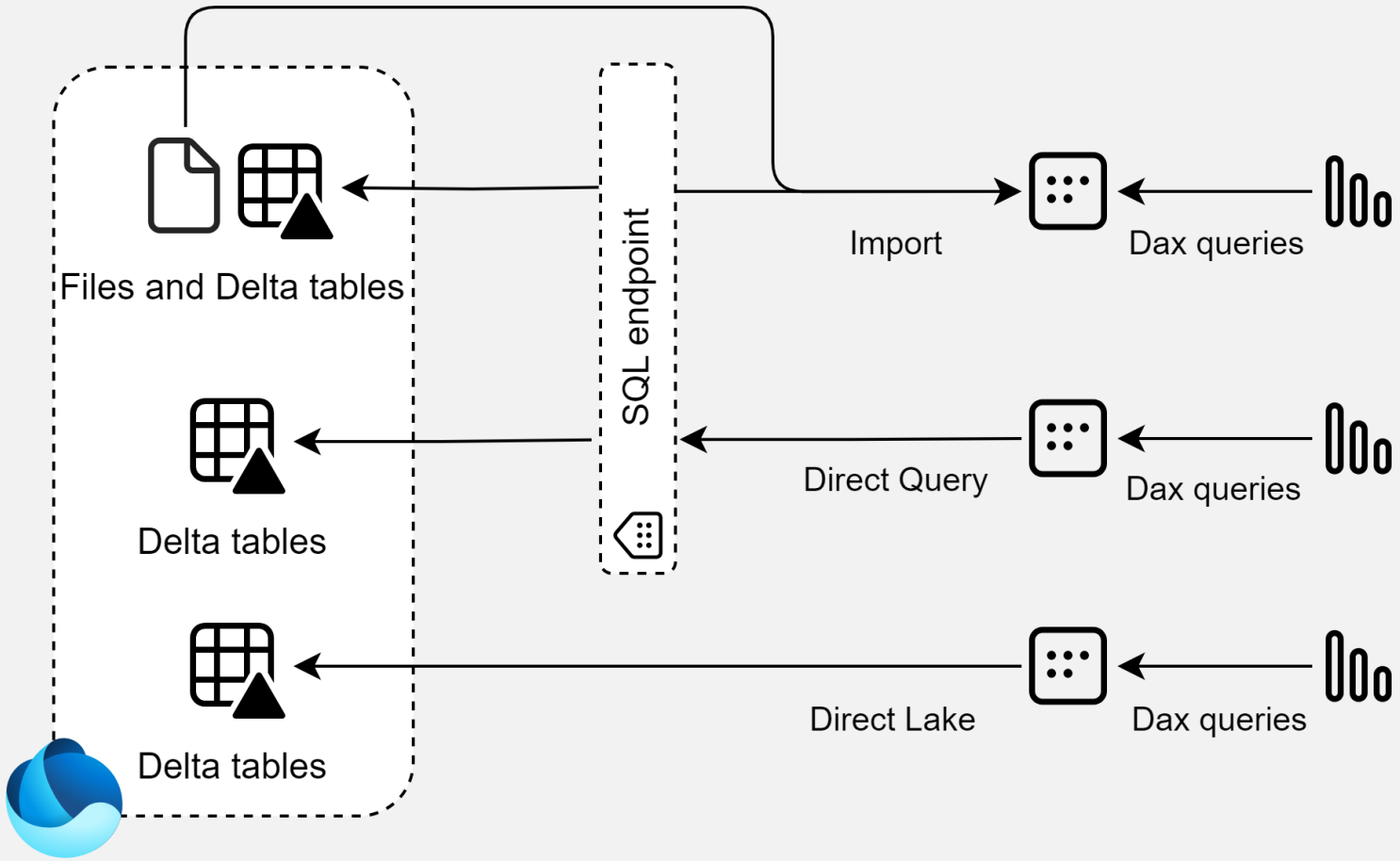
Different types of storage modes

Three familiar storage modes

- **Import** – data cached in the model
- **DirectQuery** – queries are submitted to the back-end data source
- **Dual** – can act in both above storage modes, depending on query context



Direct Lake



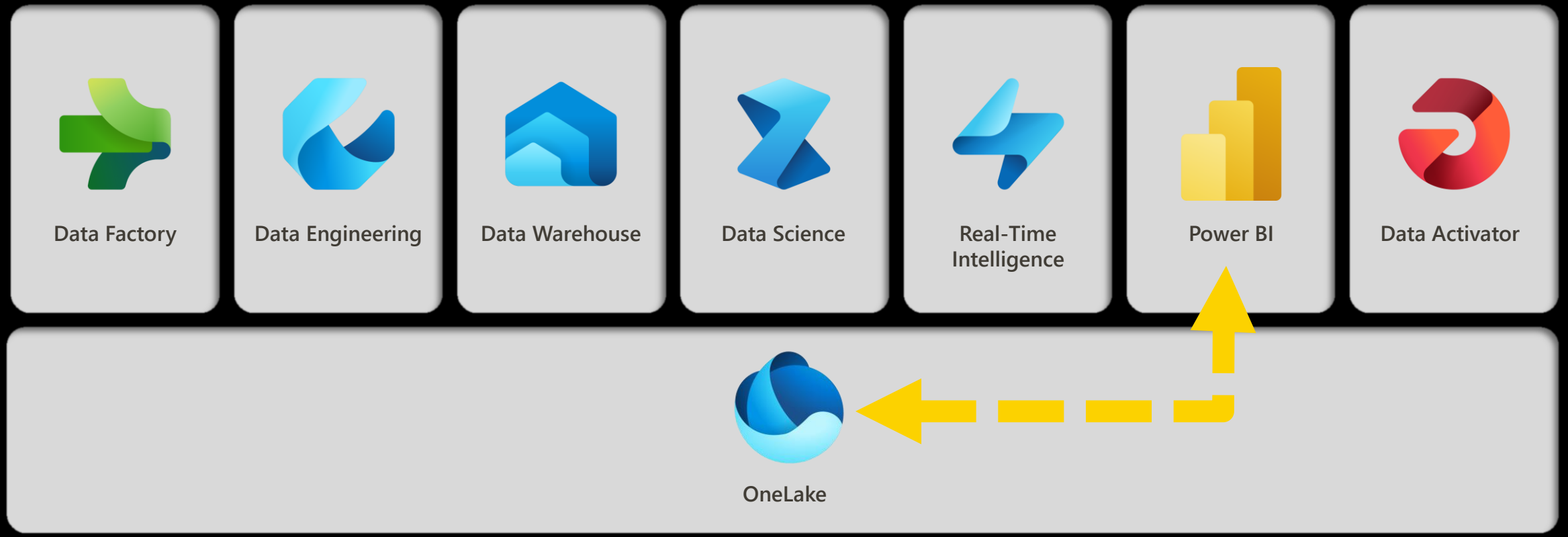
Latent & duplicative but fast

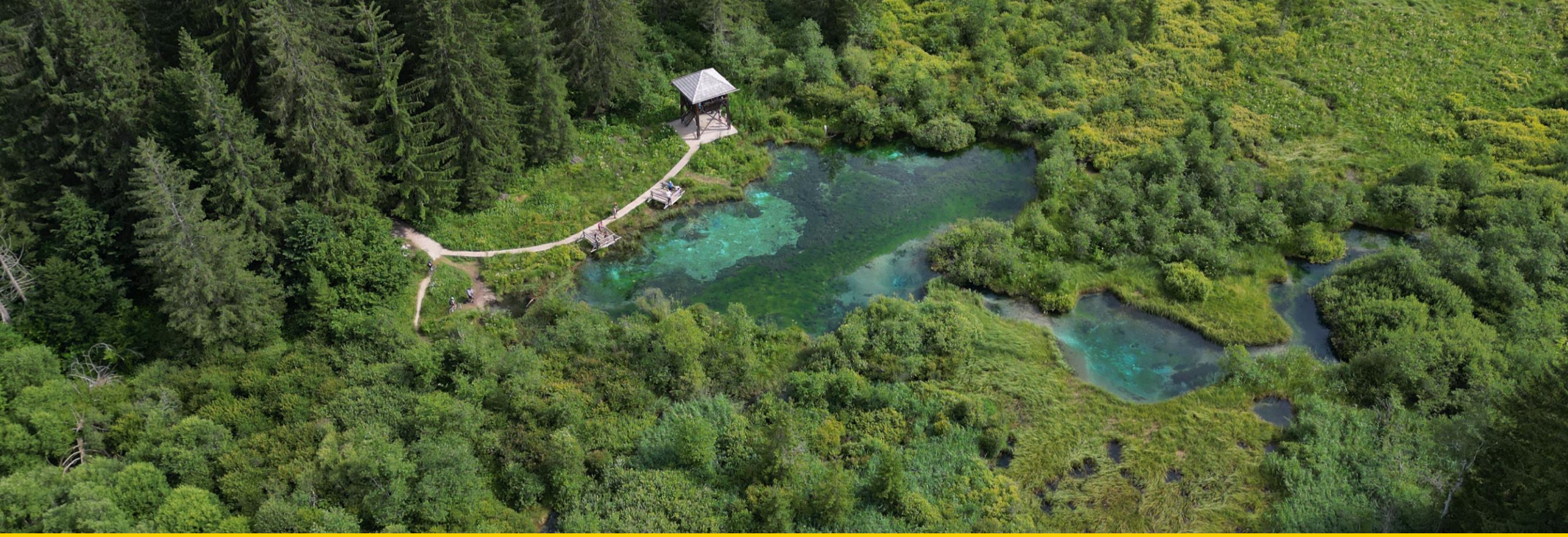
Slow, but real time

Considered being
Best of both worlds

Direct Lake is only applicable to Fabric

Fabric


Data Factory
Data Engineering
Data Warehouse
Data Science
Real-Time Intelligence
Power BI
Data Activator
OneLake



Build your data model benefitting Direct Lake

Mathias Halkjær - Marc Lelijveld

Data transformations

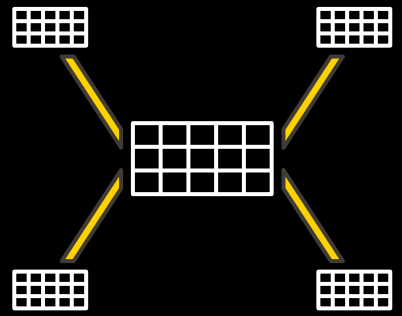
No Power Query or other data transformation capabilities*

Data transformations should be done as far upstream as possible

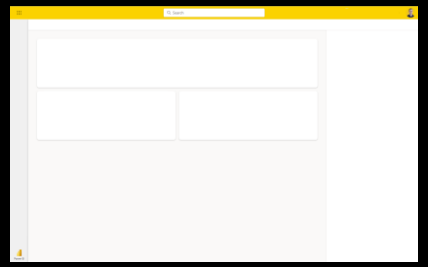
Data transformation directly in the Lakehouse unlocks “new” possibilities

Medallion architecture

Data modeling



General best practice to have a star schema still applies



Web (browser) experience only to develop new data models

You can edit from Desktop – though desktop acts like a browser

Demo

Internals & performance

Mathias Halkjær - Marc Lelijveld

Delta (Parquet)

Parquet (File format)

- Column-store
- Open industry standard
- Compressed & Encoded
- Parallelism

Enables fast bulk operations of large data volumes

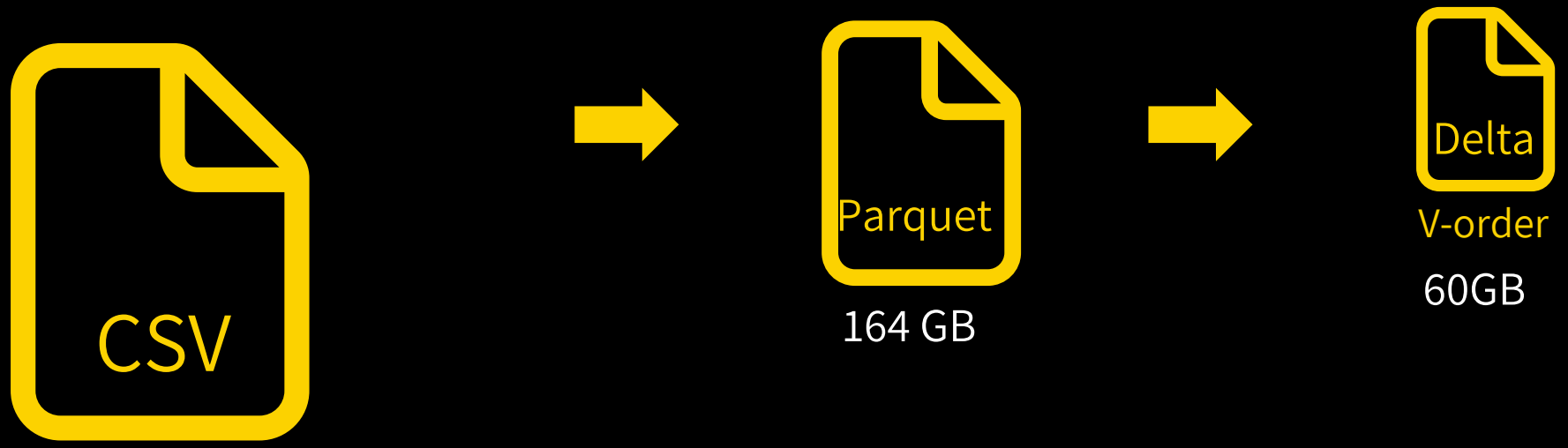
Delta (Storage layer management)

- Also, an open industry standard
- ACID transactions & schema enforcement
- Delete, update, merge
- Time-travel
- Optimized for querying, skipping and pruning

Brings warehouse reliability to the data lake

From Z-order to V-order

Yellow taxi (3 Billion rows)



x3.2

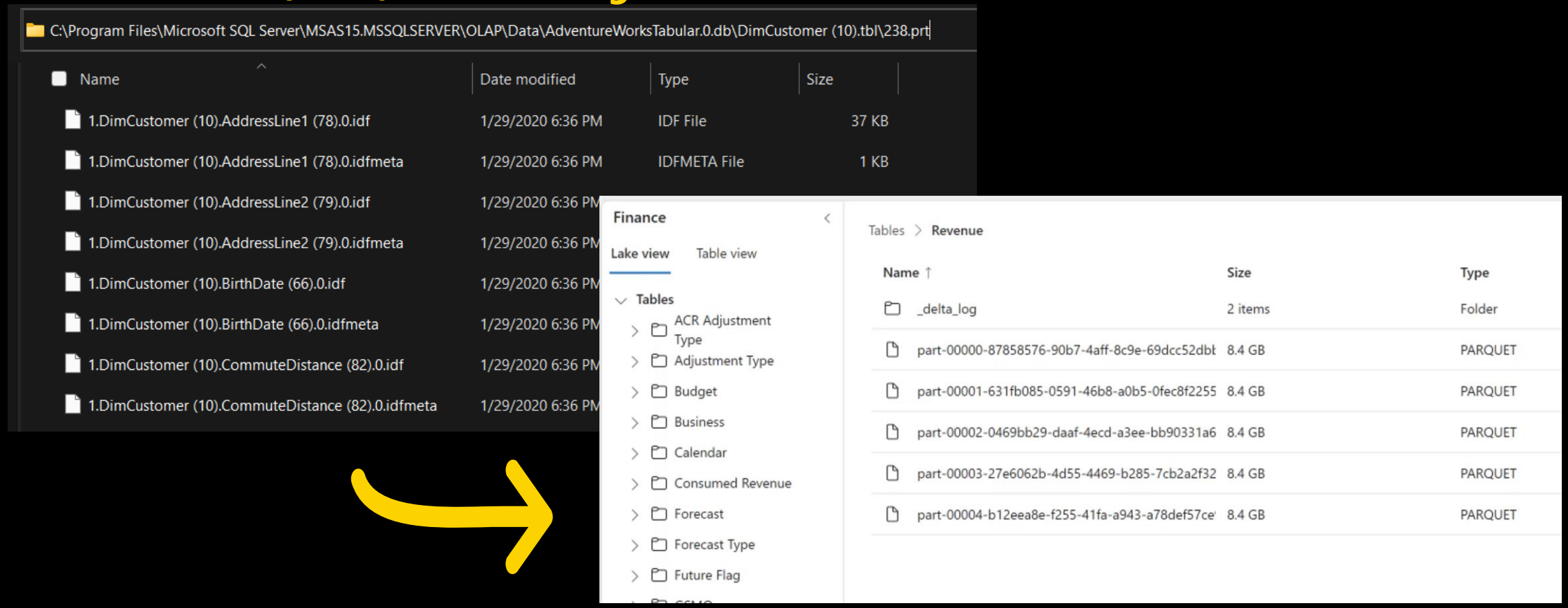
Less I/O for all * workloads

* Microsoft benchmark

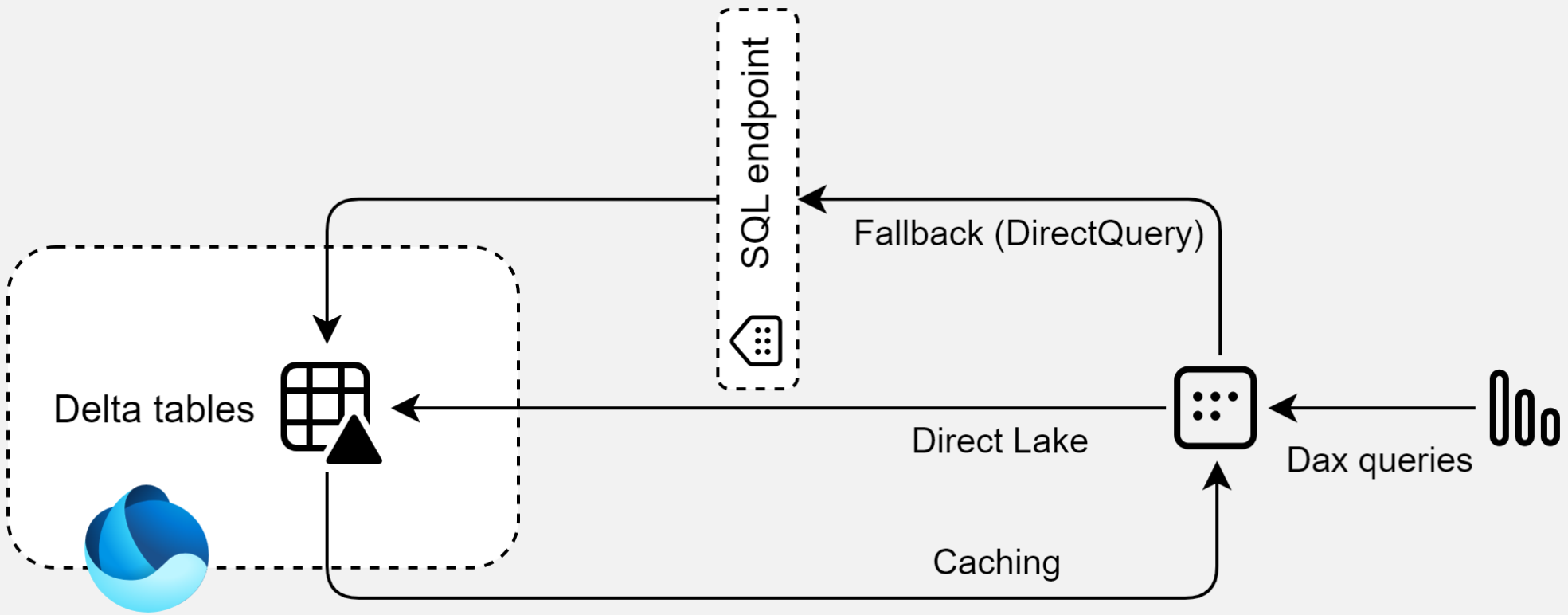
Data saving

The Analysis Services column-oriented storage using Delta Lake/Parquet open standard for Direct Lake

SSAS, AAS, Power BI large models



Fallback & Caching



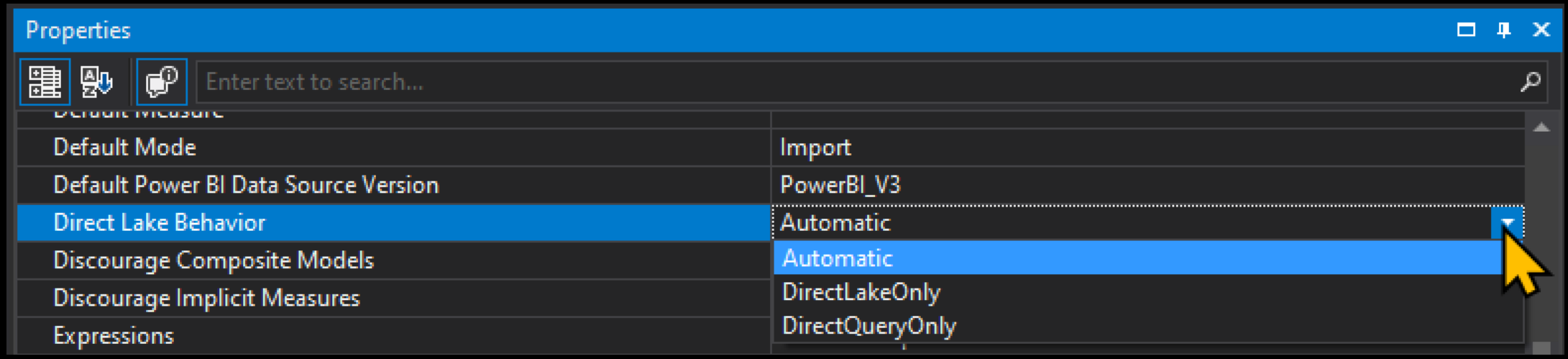
Fallback

When could fallback to DirectQuery happen?

- Special data types
- Large data volumes that does not fit the capacity size
- Composite models
- When you manually configure security
  Item level on lakehouse

Disable fallback

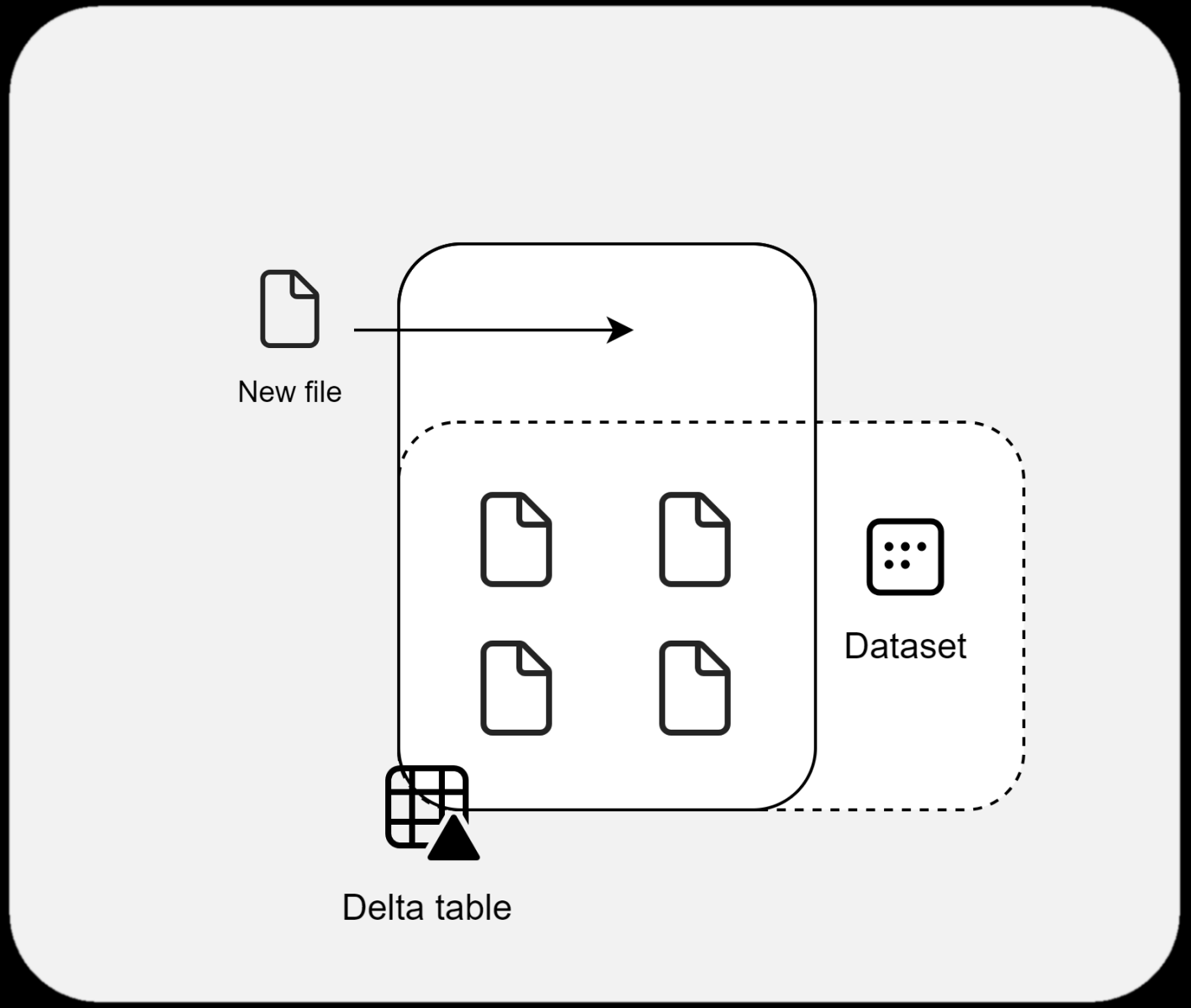
Using Tabular Editor (or any other tool using XMLA) you influence Direct Lake behavior. Decide if it is allowed to fall back to DirectQuery or not.



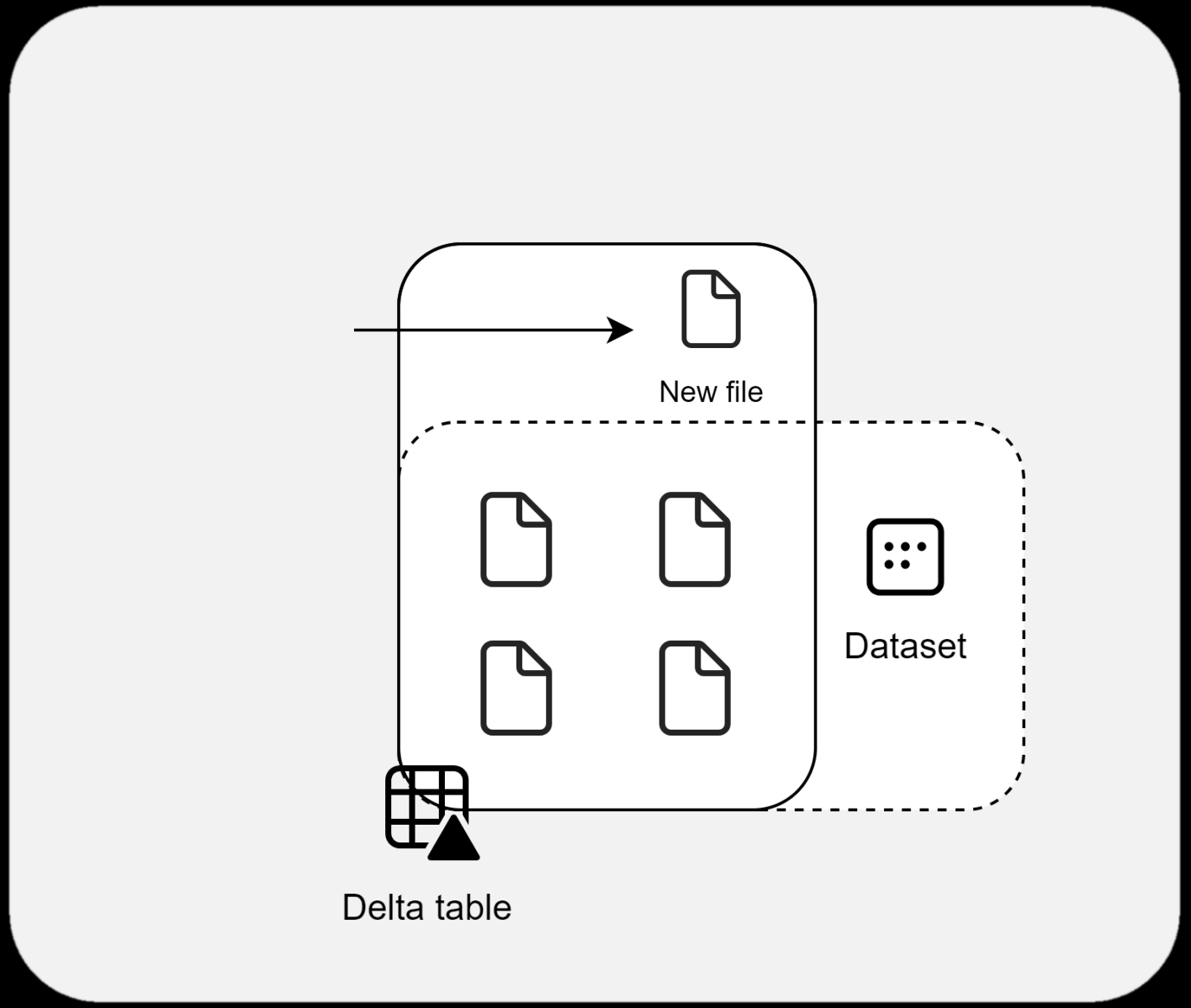
Introducing Framing

Metadata **refresh** which does not actually load the data, but only the delta table definitions.

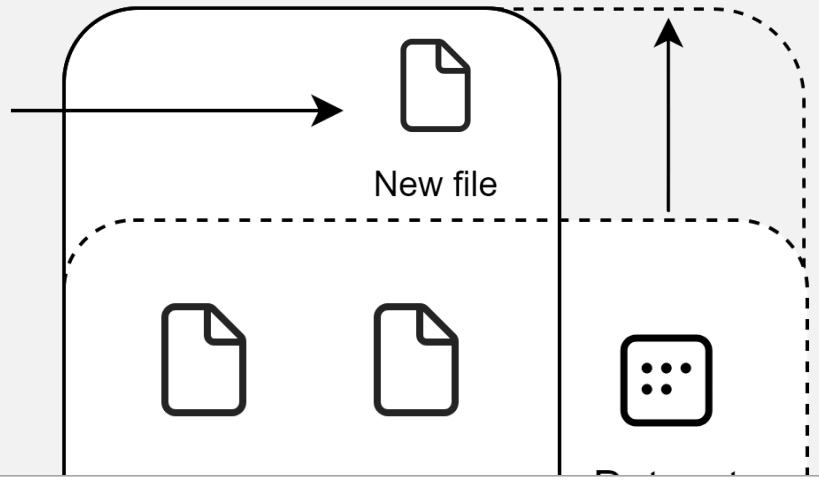
Framing



Framing



Framing



Refresh

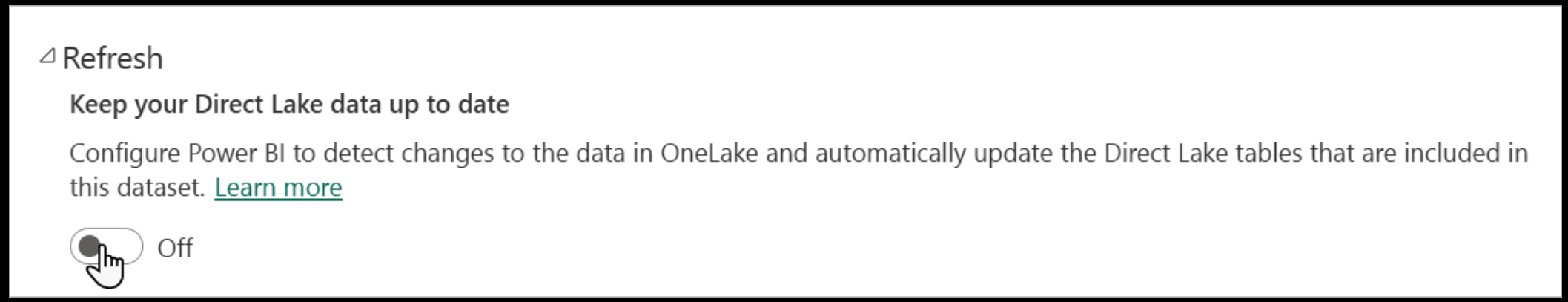
Keep your Direct Lake data up to date

Configure Power BI to detect changes to the data in OneLake and automatically update the Direct Lake tables that are included in this dataset. Learn more

Off

Automatic update (Refresh)

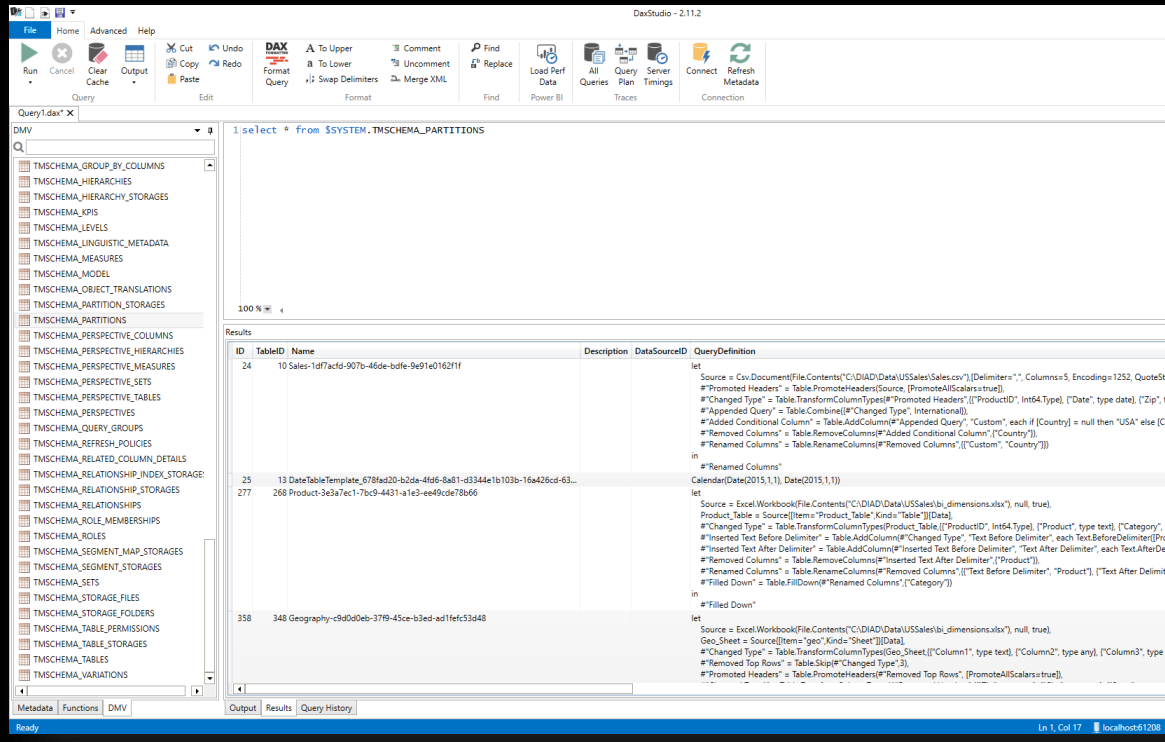
- Meta data updates
- Frequency of data updates (framing)
- Capacity impact with regards to caching + flushes the cache (partly)
- Date Integrity – keep data in sync (Fact updated but Dim not etc)



Dynamic Management Views

Analysis Services Dynamic Management Views (DMVs) are queries that return information about model objects, server operations, and server health.

- DB Schema = Database model
- DISCOVER = Operations & Sessions
- TM Schema = Tabular = Power BI / AAS
- MD Schema = MDX = Multidimensional



Demo

But wait...

Previously, all your data got evicted from cache after reframing.

But now;

- Data does not necessarily get evicted, if the segments in which the data was loaded did not change at the source.
- Behavior dependent on row groups (partitioning) of delta tables.

Forced eviction can be triggered by ProcessClear + ProcessFull commands.

Demo

Optimization

A good model – is key to get the most out of Direct Lake!

So, Semantic Link (Labs) will be your best friend to optimize your models!

run_model_bpa

Run Best Practice Analyzer scan for a semantic model, and display an HTML visualization, or return a pandas dataframe.

Python

```
run_model_bpa(dataset: str | UUI
'table', 'zip', 'none'] = 'html'
```

sempy_labs.directlake.show_unsupported_direct_lake_objects(*dataset: str | UUID, workspace: str | UUID | None = None*)→ Tuple[DataFrame, DataFrame, DataFrame]

Returns a list of a semantic model's objects which are not supported by Direct Lake based on official documentation.

Parameters:
- **dataset** (*str | uuid.UUID*) – Name or ID of the semantic model.
- **workspace** (*str | uuid.UUID, default=None*) – The Fabric workspace name or ID. Defaults to None which resolves to the workspace of the attached lakehouse or if no lakehouse attached, resolves to the workspace of the notebook.

Returns: 3 pandas dataframes showing objects in a semantic model which are not supported by Direct Lake.

Return type: pandas.DataFrame, pandas.DataFrame, pandas.DataFrame

Advanced patterns

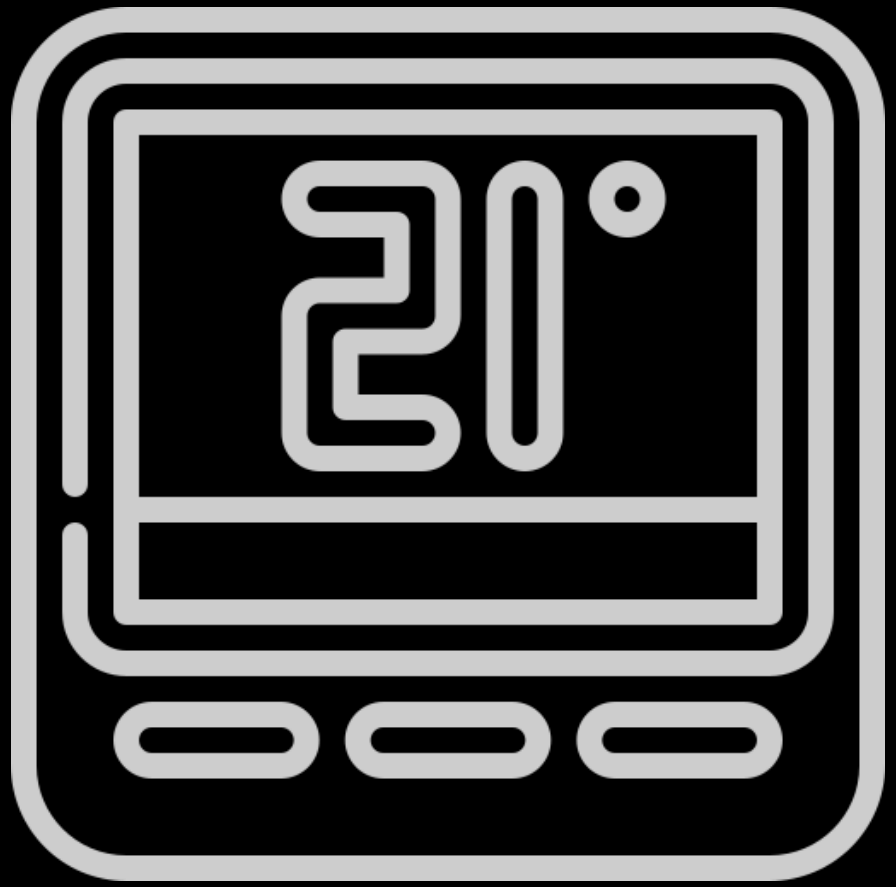
Mathias Halkjær - Marc Lelijveld

Temperature management

Keep it WARM!

Make sure your users are served optimally and avoid the capacity memory to be flushed.

- Either due to reframing
- Or by eviction due to capacity utilization



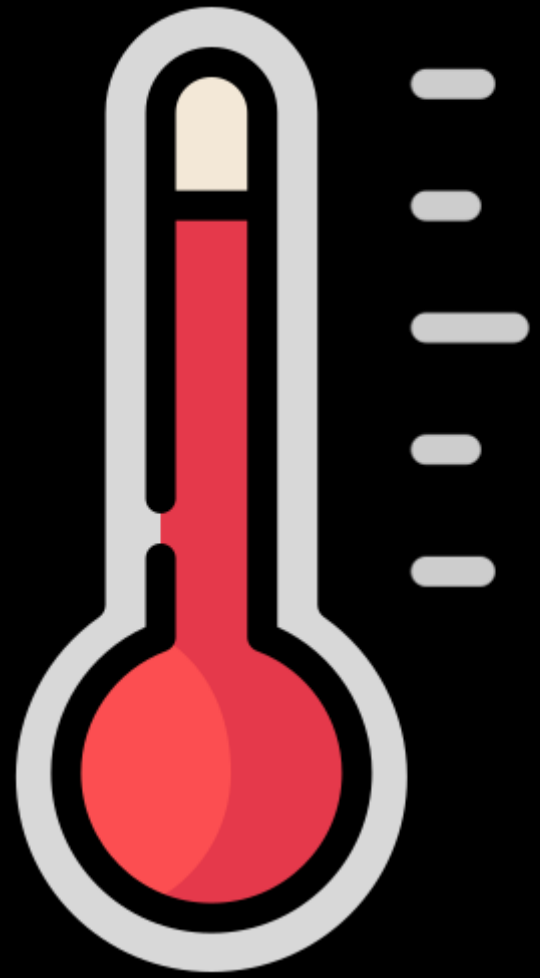
Temperature management

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**!

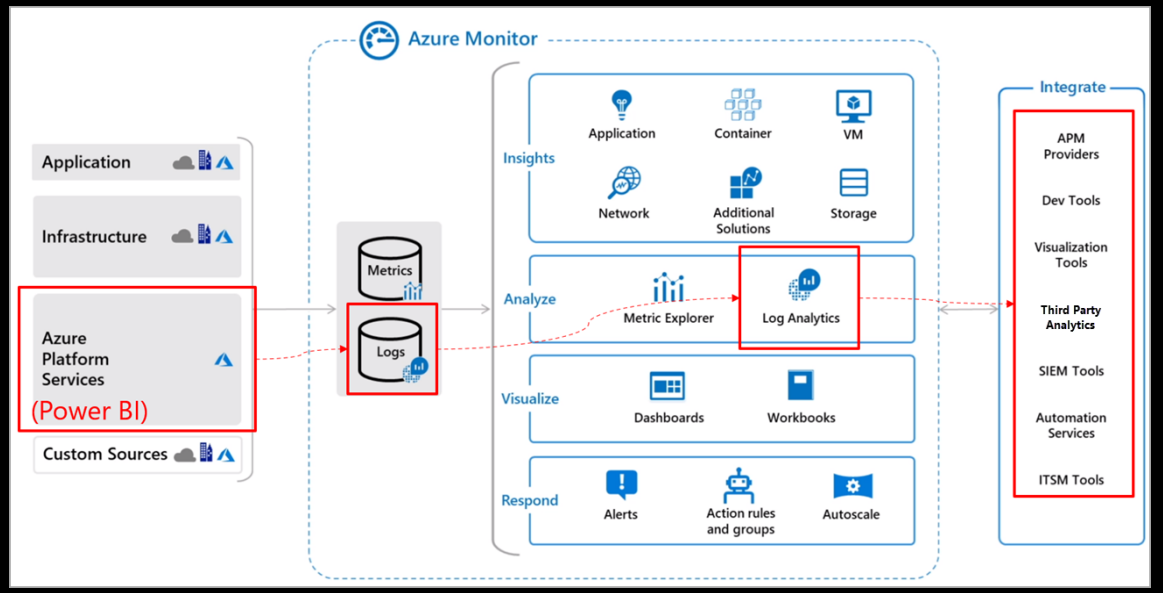


What should stay in memory?

Azure Monitor / Log Analytics

Or

Fabric Workspace Monitoring (preview)



Delivers a comprehensive solution for collecting, analyzing, and acting on telemetry from your cloud. It helps you understand how your applications are performing and proactively identifies issues affecting them and the resources they depend on.

Warm up your data using Semantic Link Labs

sempy_labs.directlake.warm_direct_lake_cache_isresident(*dataset: str | UUID, workspace: str | UUID | None = None*)→ DataFrame

Performs a refresh on the semantic model and puts the columns which were in memory prior to the refresh back into memory.

Parameters:
- **dataset** (*st*
- **workspac**
  ID. Defaul
  lakehouse
  notebook.

Returns: Returns a par
memory.

Return type: pandas.DataF

sempy_labs.directlake.warm_direct_lake_cache_perspective(*dataset: str | UUID, perspective: str, add_dependencies: bool = False, workspace: str | UUID | None = None*)→ DataFrame

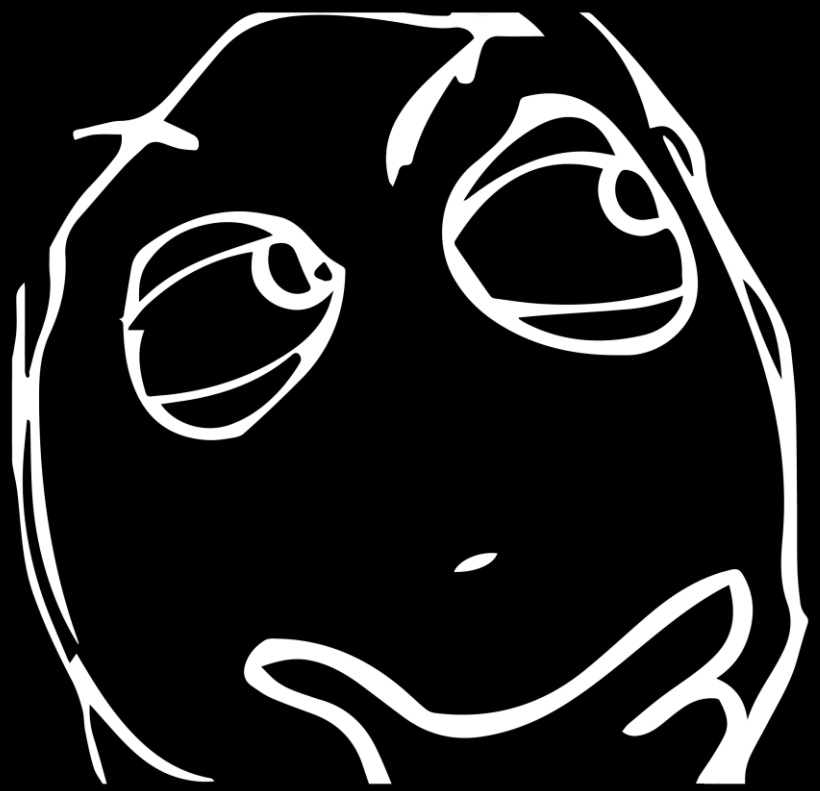
Warms the cache of a Direct Lake semantic model by running a simple DAX query against the columns in a perspective.

Parameters:
- **dataset** (*str | uuid.UUID*) – Name or ID of the semantic model.
- **perspective** (*str*) – Name of the perspective which contains objects to be used for warming the cache.
- **add_dependencies** (*bool, default=False*) – Includes object dependencies in the cache warming process.
- **workspace** (*str | uuid.UUID, default=None*) – The Fabric workspace name or ID. Defaults to None which resolves to the workspace of the attached lakehouse or if no lakehouse attached, resolves to the workspace of the notebook.

Returns: Returns a pandas dataframe showing the columns that have been put into memory.

Return type: pandas.DataFrame

Considerations



Should I change all my solutions to start using Direct Lake?

- Do you really need “real time” data?
- Consider impact on capacities when falling back to DQ
- Performance is better than DQ

Resources

Direct Lake generic documentation
https://learn.microsoft.com/en-us/power-bi/enterprise/Direct Lake-overview

Analyze performance for Direct Lake
https://learn.microsoft.com/en-us/power-bi/enterprise/directlake-analyze-qp

On-demand loading of Direct Lake Power BI datasets in Fabric
https://blog.crossjoin.co.uk/2023/07/02/on-demand-loading-of-direct-lake-power-bi-datasets-in-fabric/

Direct Lake Frequently Asked Questions
https://fabric.guru/power-bi-direct-lake-mode-frequently-asked-questions/

Direct Lake framing & warm-up
https://data-marc.com/exploring-direct-lake-framing-and-warm-up-data-using-semantic-link-in-fabric-notebooks/

Wrap up

Direct Lake…

- Only applicable when using MS Fabric
- No data is imported / copied
- On-demand loading
- Reads data from Lake / Parquet format – Delta is a must
- Performance dependent on capacity size/utilization
- Falls back to DirectQuery when limitations are hit!

Tell us what you think!