

A deep dive into Direct Lake



Big thank you to our great sponsors

Premium



Fellowwind

Fellowmind

ITPA
CONSULTING

ITPA consulting

data on
Power your data

dataon

Standard



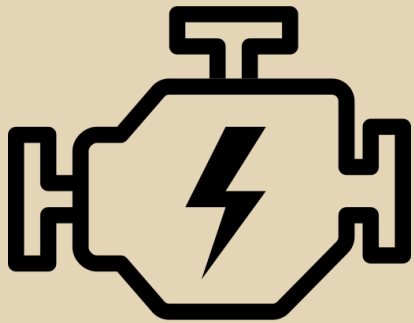
A|S

enversion



After this session

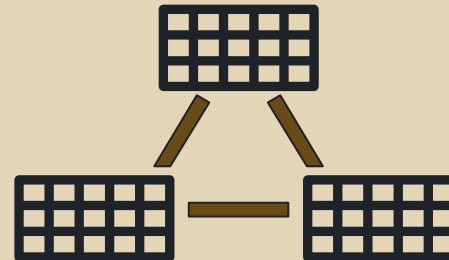
Understanding the engine
for Direct Lake



Analyzing performance
across different scenarios



Data modeling benefitting
from Direct Lake



Go next level with
advanced patterns



Mathias Halkjær

Principal Architect, Data & Analytics
Fellowmind Denmark



@halkjaerm



linkedin.com/in/mhalkjaer



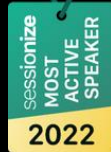
Fluxbi.com

FAVORITE STUFF:



Marc Lelijveld

Technical Evangelist | Solution Architect
Macaw Netherlands



@MarcLelijveld



[linkedin.com/in/MarcLelijveld](https://www.linkedin.com/in/MarcLelijveld)



Data-Marc.com

FAVORITE STUFF:





Storage modes

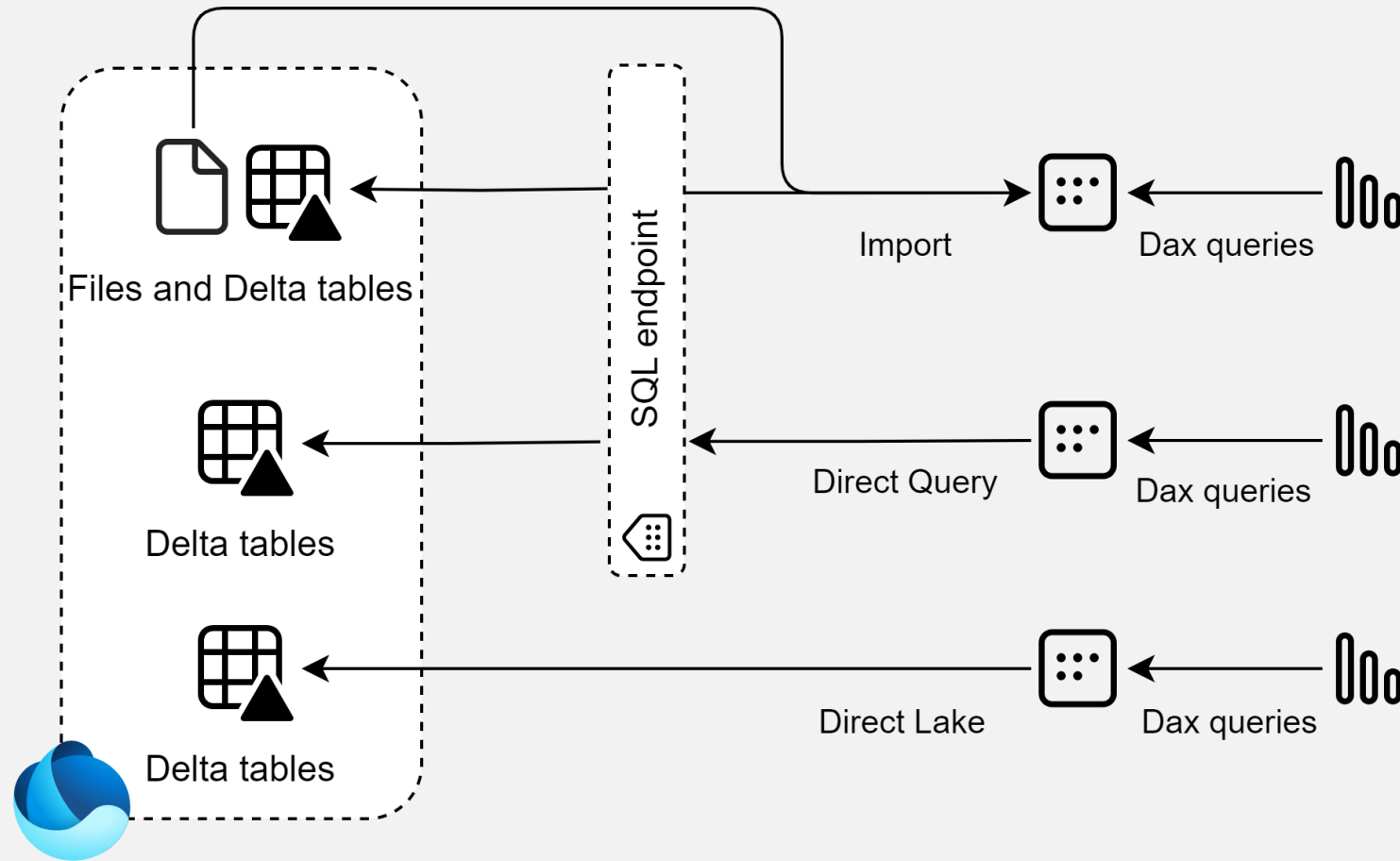
Different types of storage modes

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

The screenshot displays the Microsoft Power BI Desktop interface. The main workspace shows a data model with four tables: 'Product Subcategory', 'Product', 'Internet Sales', and 'Internet Sales - Agg'. The 'Internet Sales' table is highlighted with a yellow border. The 'Advanced' section of the Properties pane is highlighted with a red border, showing the 'Storage mode' dropdown set to 'DirectQuery'. The 'Properties' pane also shows other settings like 'Name', 'Description', 'Synonyms', 'Row label', and 'Key column'. The top ribbon includes 'File', 'Home', 'Help', and 'External Tools' tabs, with various icons for data sources, queries, relationships, and security.

Direct Lake



Latent & duplicative but fast

Slow, but real time

Best of both worlds

Direct Lake is only applicable to Fabric

Fabric



Data Factory



Synapse Data Engineering



Synapse Data Warehouse



Synapse Data Science



Synapse Real-Time Analytics



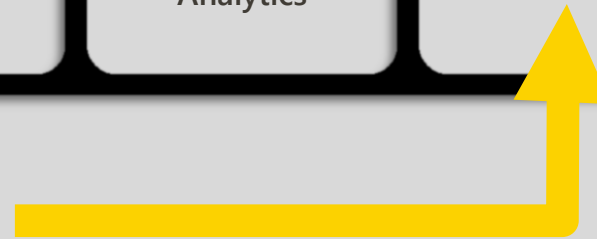
Power BI



Data Activator
(coming soon)



OneLake





Build your data model benefitting Direct Lake

Data transformations



No Power Query or other data transformation capabilities*



Data transformations should be done as far upstream as possible

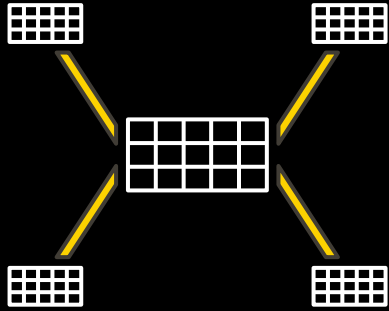


Data transformation directly in the Lakehouse unlocks “new” possibilities

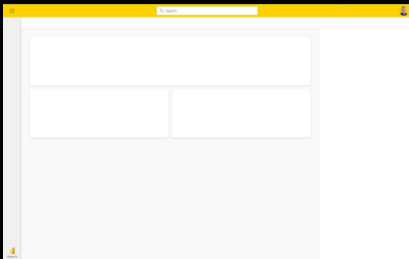
Medallion architecture



Data modeling



General best practice to have a star schema still applies



Web (browser) experience only to develop data models

Data modeling best practices unchanged

- Starschema all the things!
 - Avoid bi-directional or many-to-many relationships
 - Avoid limited relationships
 - Implement role-playing dimensions rather than duplication
 - Minimize redundant measure using calculation groups
 - Avoid ambiguous data models
- ... etcetera

Demo





Internals & performance

Delta (Parquet)



Parquet (File format)

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

Enables fast bulk operations of large data volumes

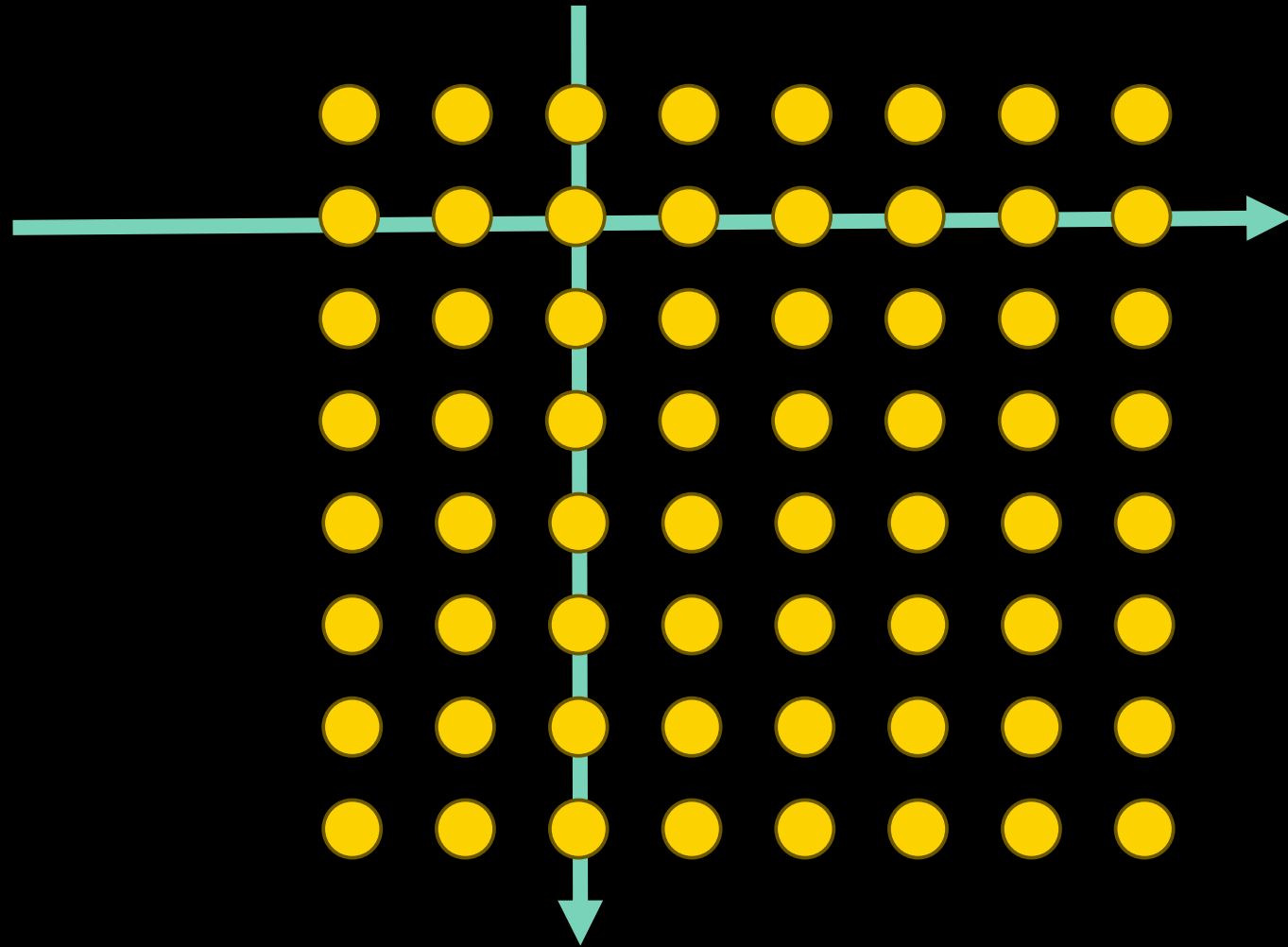
Delta (Storage layer management)

- 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

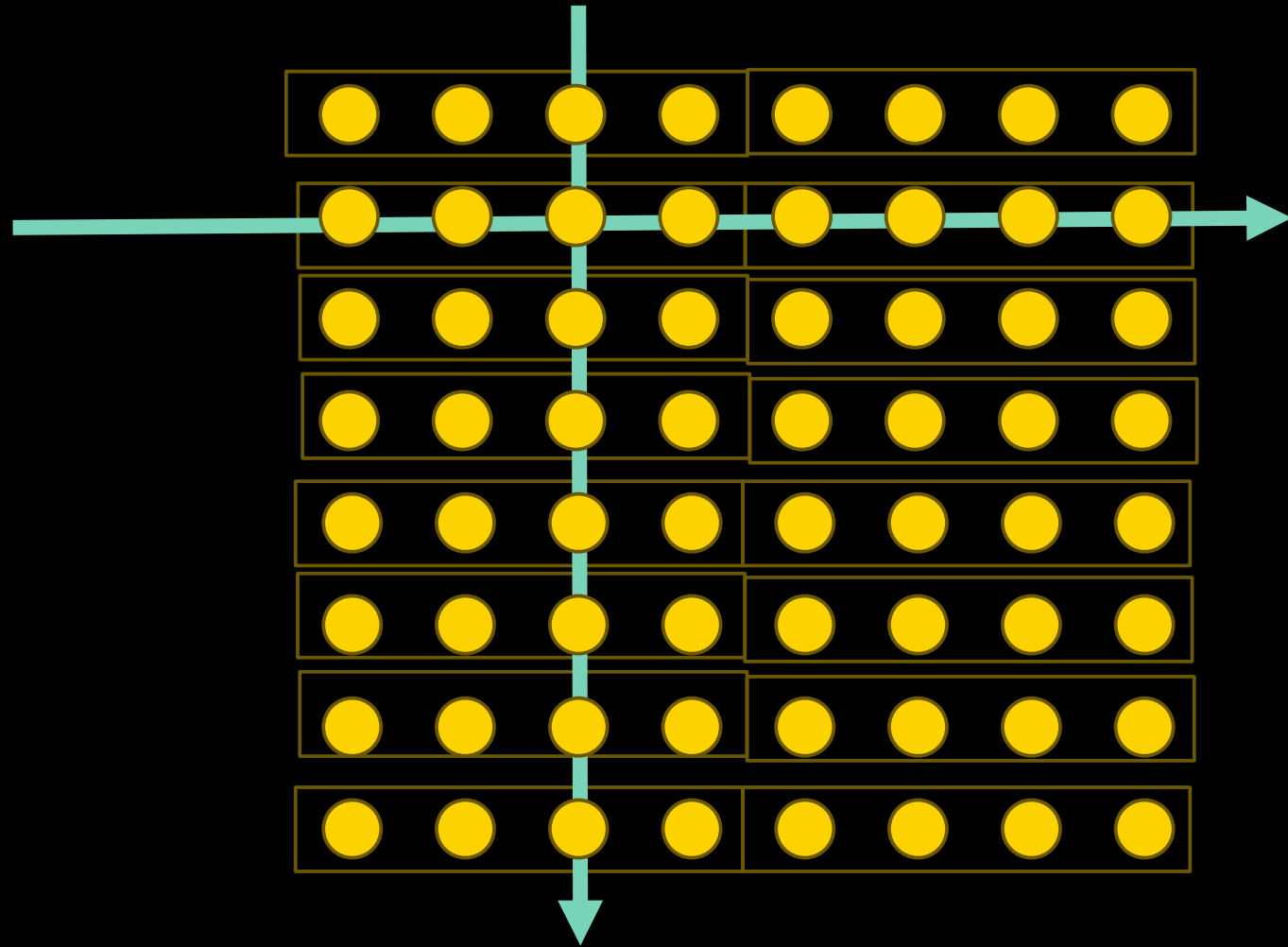
From Z-order to V-order

SELECT * FROM points WHERE x=3 or y=2



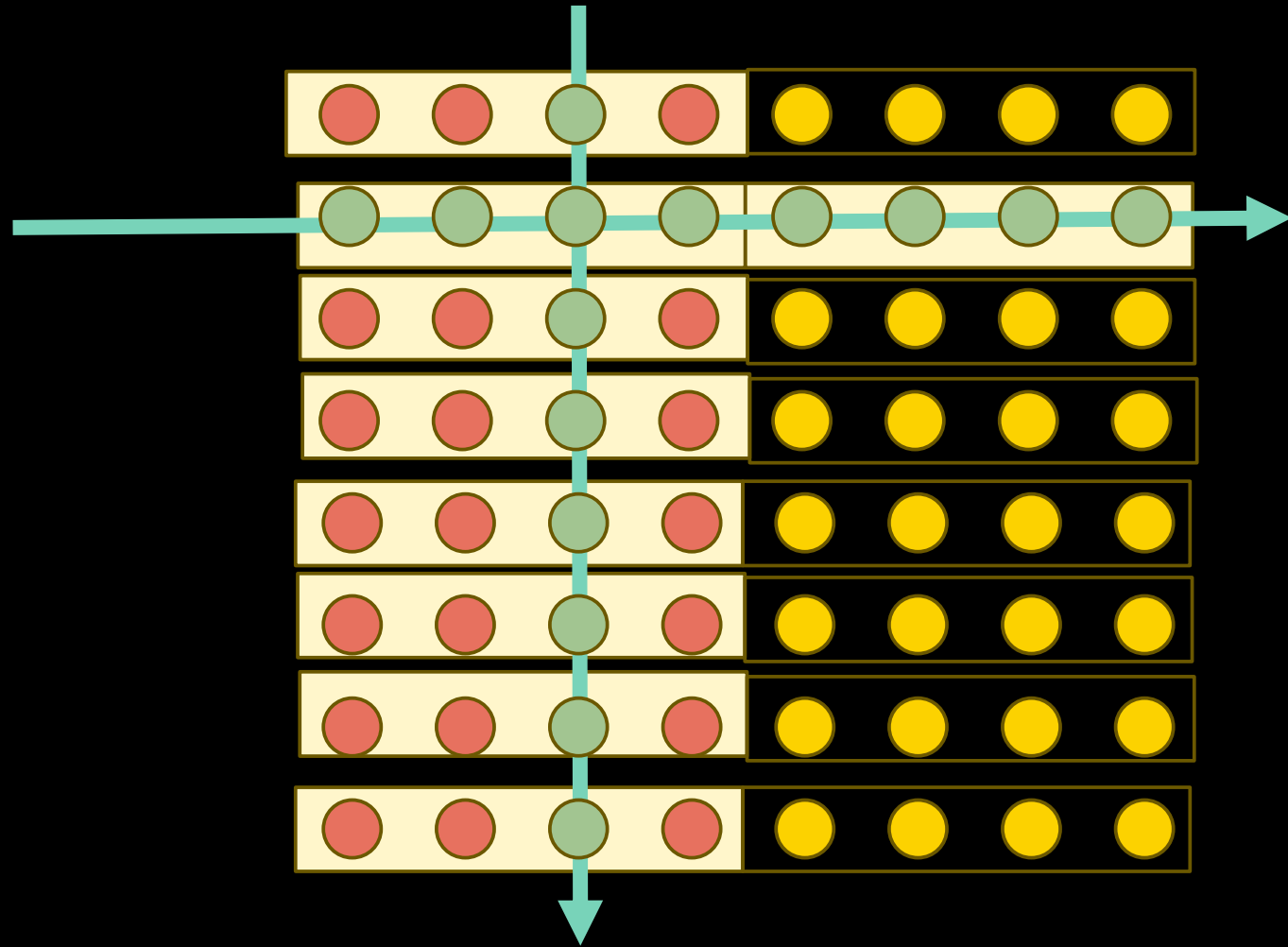
From Z-order to V-order

SELECT * FROM points WHERE x=3 or y=2



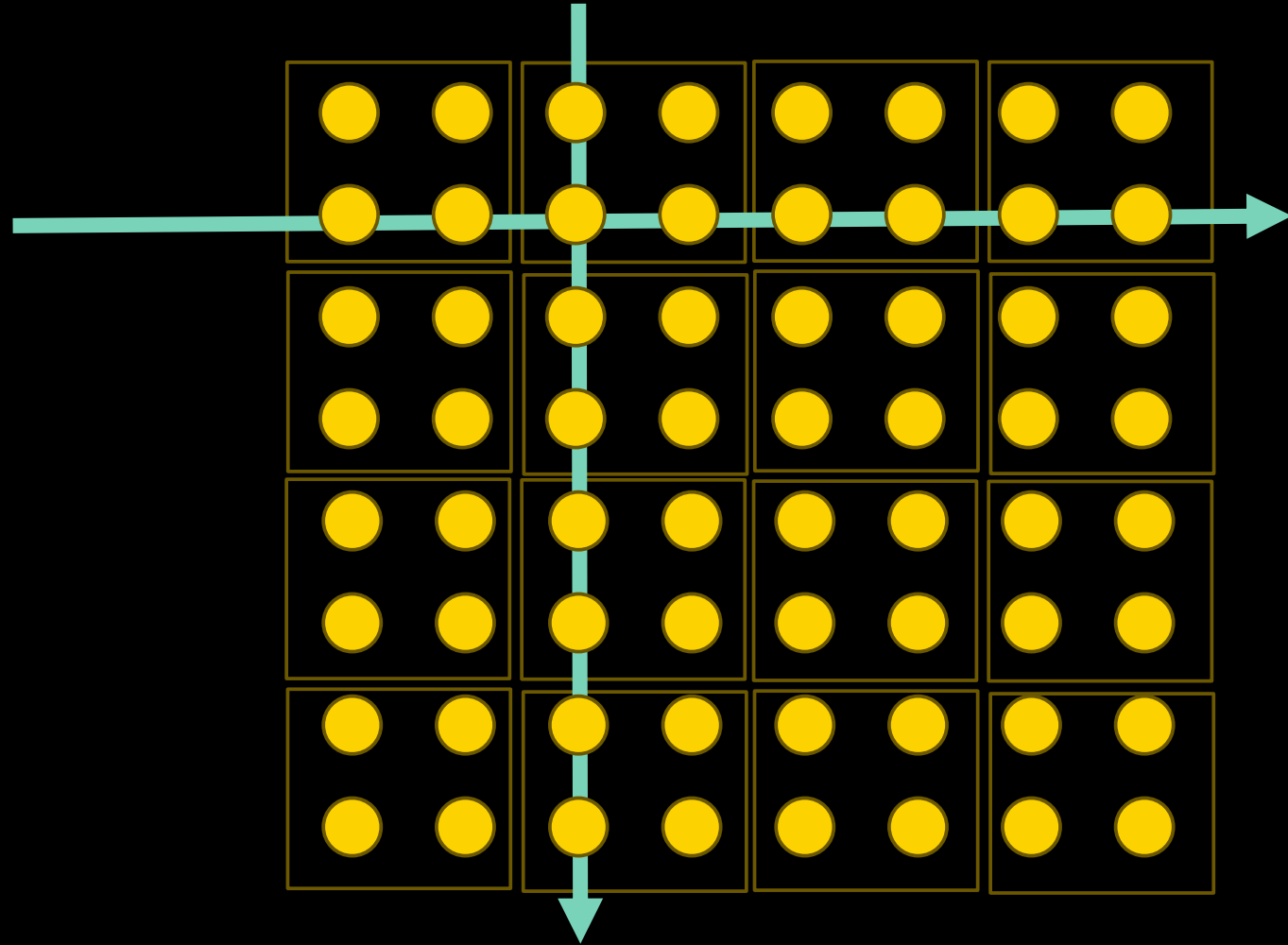
From Z-order to V-order

SELECT * FROM points WHERE x=3 or y=2



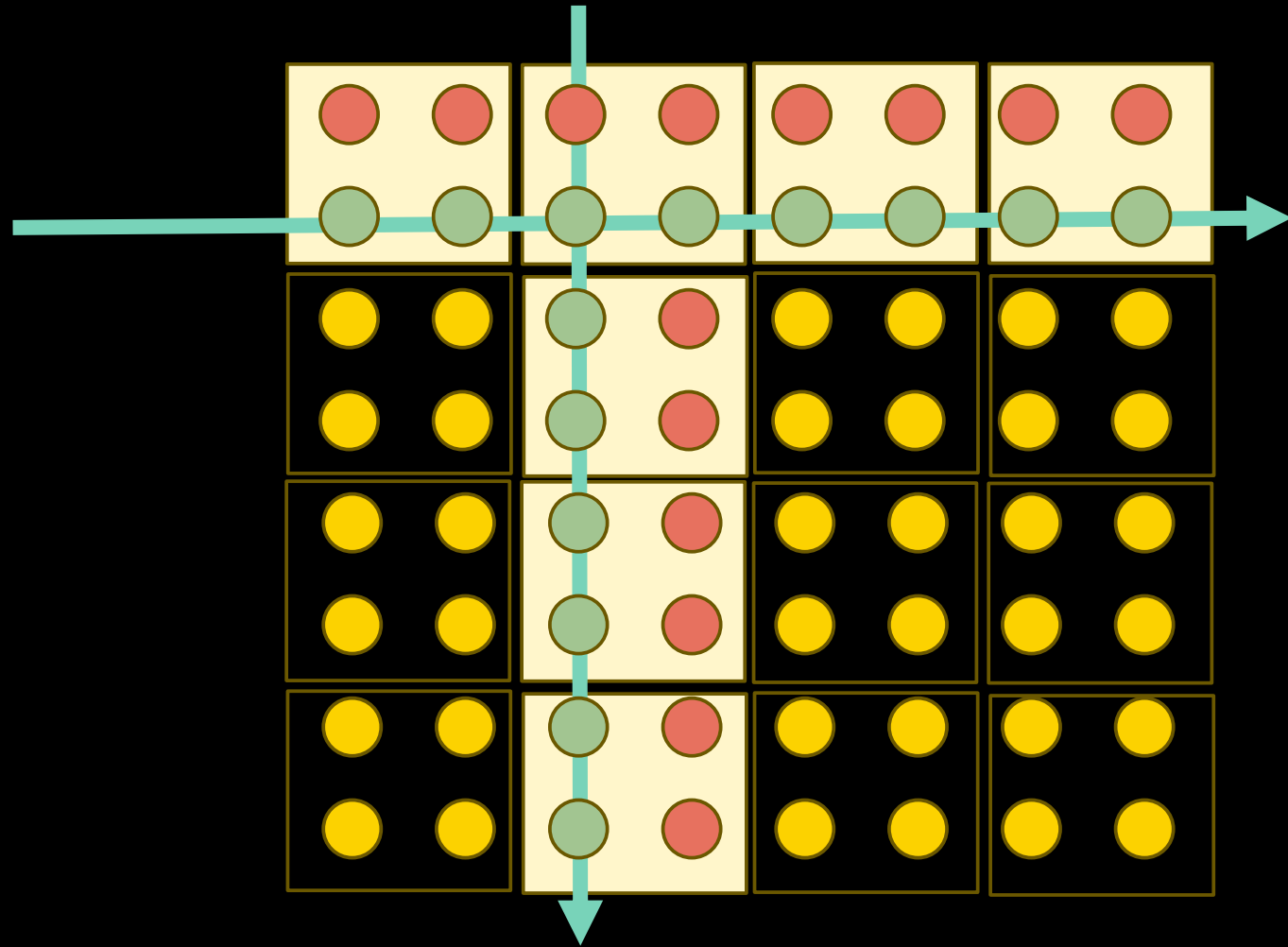
From Z-order to V-order

SELECT * FROM points WHERE x=3 or y=2



From Z-order to V-order

SELECT * FROM points WHERE x=3 or y=2

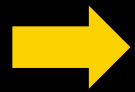


From Z-order to V-order

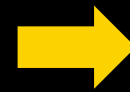
Yellow taxi (3 Billion rows)



416 GB



164 GB



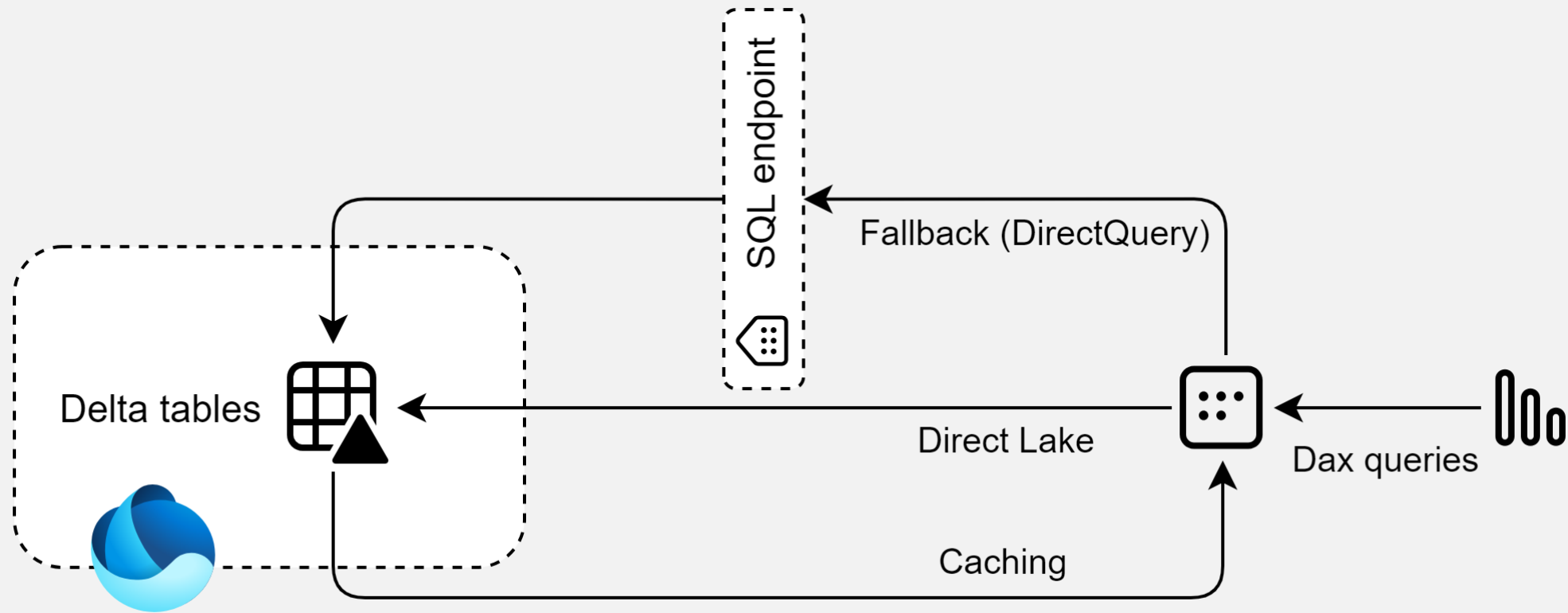
V-order
60GB

x3.2

Less I/O for all *
workloads

* Microsoft benchmark

Fallback & Caching

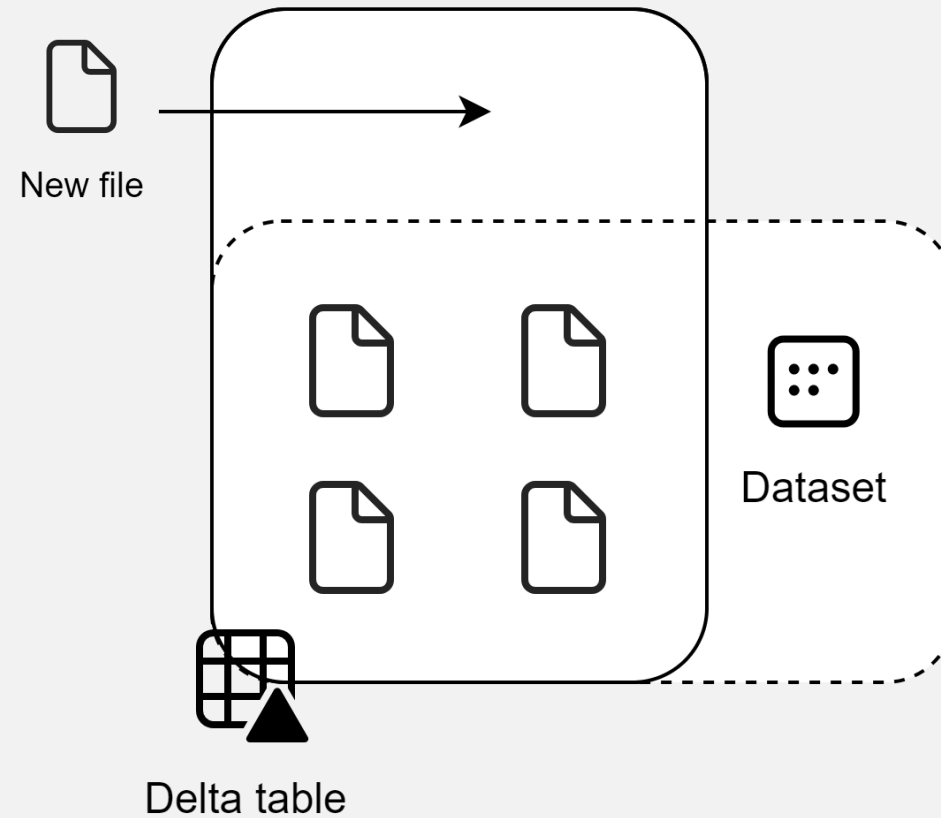


Fallback

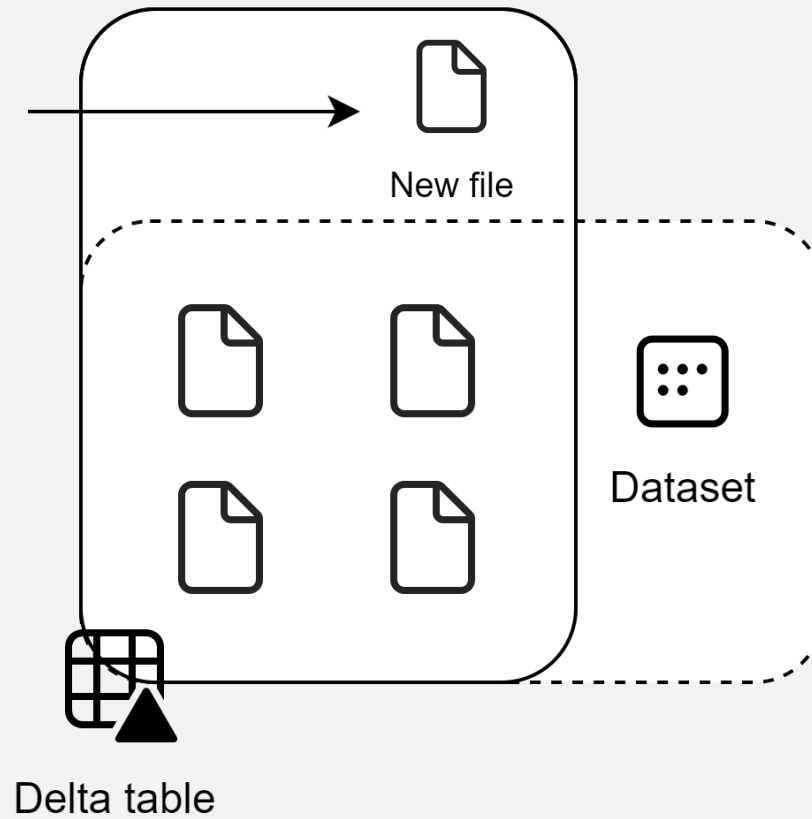
When could fallback to DirectQuery happen?

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

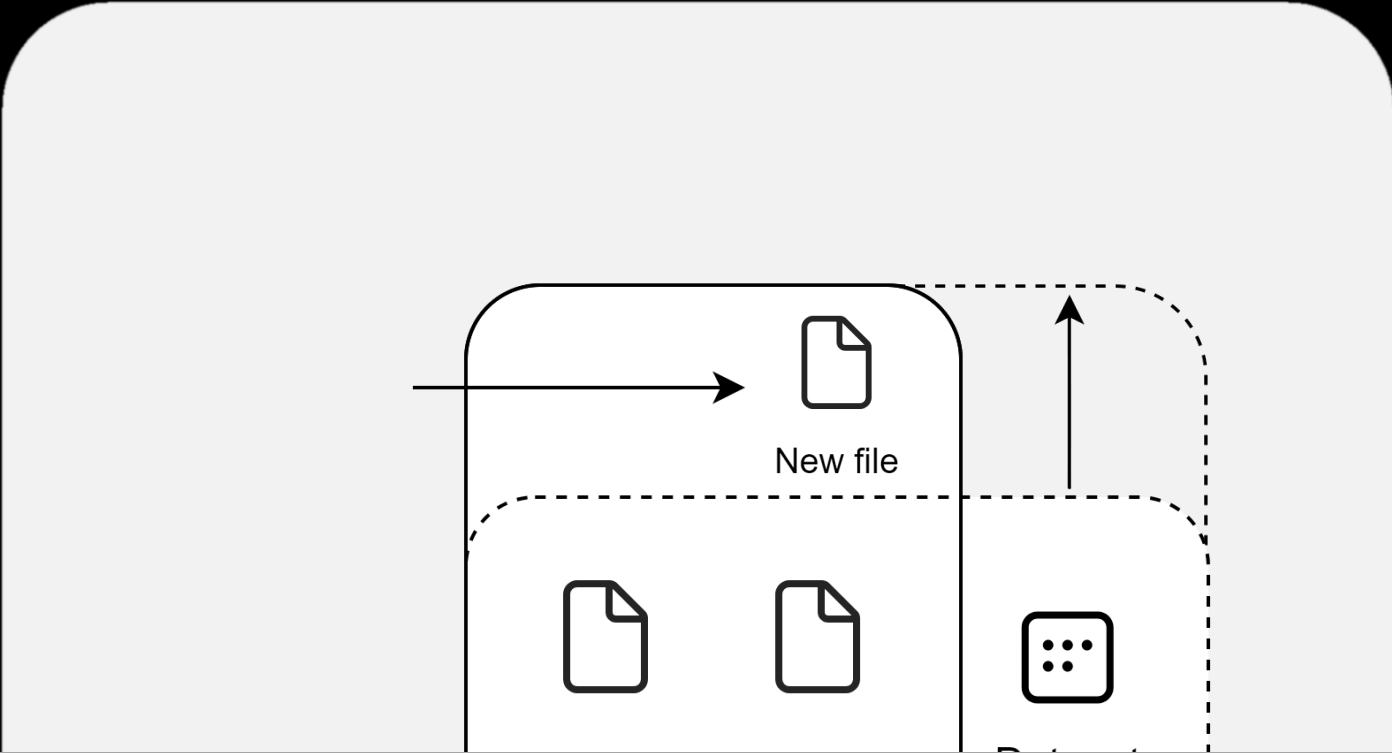
Framing



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](#)

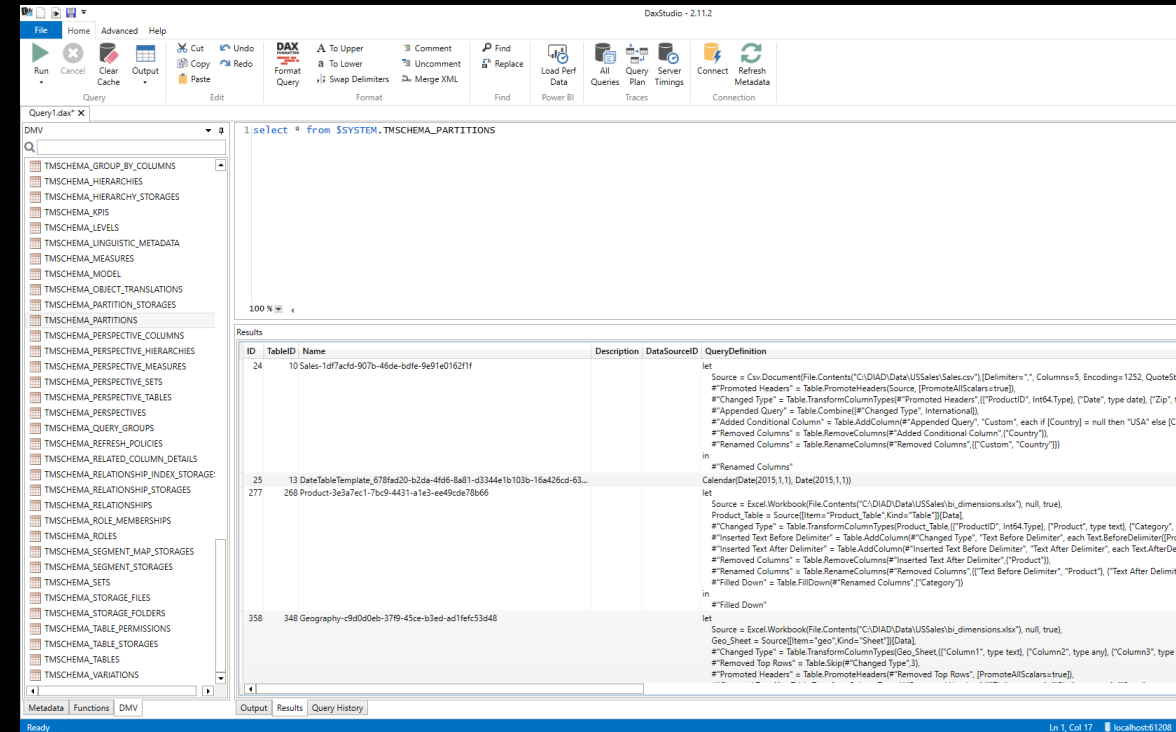


Off

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





Advanced patterns

XMLA

Talk with the back-end server, just like an Analysis Services server (read / write)

Server settings

Connection string

Data Source=powerbi://api.powerbi.com/v1.0/myorg/Dive%20into%20

Copy

XMLA Endpoint

Read Write ▾

Apply Discard

⊗ **Caution**

At this time, a write operation on a dataset authored in Power BI Desktop will prevent it from being downloaded back as a PBIX file. Be sure to retain your original PBIX file.

Introducing calculation groups

Benefits

- Reduce the number of redundant measures and grouping common measure expressions as calculation items
- Avoids duplicating logic in different measures
- Typical use cases are
 - Time-intelligence calculations (YTD / QTD / MTD / ...)
 - Format string change, like currency conversions

Limitations

- Can only be created from external tools in Power BI (Any tool using the XMLA endpoint such as Tabular Editor) – **but stay tuned....**
- Object level security on Calculation group items is not supported
- Smart narrative visuals in Power BI are not supported with Calculation Groups



Introducing calculation groups

Specific DAX expressions for Calculation Groups

- SELECTEDMEASURE()
- SELECTEDMEASURENAME()
- ISSELECTEDMEASURE()
- SELECTEDMEASUREFORMATSTRING()

Classic measure:

```
MTD =  
CALCULATE (  
    SUM ( Sales[SalesAmount] ),  
    DATESMTD ( DimDate[Date] )  
)
```

Dynamic measure context MTD with Calculation Group:

```
MTD =  
CALCULATE (  
    SELECTEDMEASURE (),  
    DATESMTD ( DimDate[Date] )  
)
```


Creating calculation groups over XMLA

The screenshot shows the Tabular Editor 3.10.1 interface. The main window displays an expression editor with the following code:

```
1 CALCULATE(  
2     SELECTEDMEASURE( ),  
3     USERRELATIONSHIP(  
4 )
```

The TOM Explorer on the right shows the following structure:

```
'Date'[DateKey] ↔-1 'InternetSales'[ShipDateKey]  
'InternetSales'[ProductKey] ↔-1 'Product'[ProductKey]
```

An error dialog box is overlaid on the screen with the following text:

Error Loading Data Model

Operation not supported for model with: ModifiedByXmlaEndpoint

Close

The background interface also shows the VertiPaq Analyzer section with the following information:

XMLA Edited

Total Size: 10,05 KB
Last Refresh: 21-9

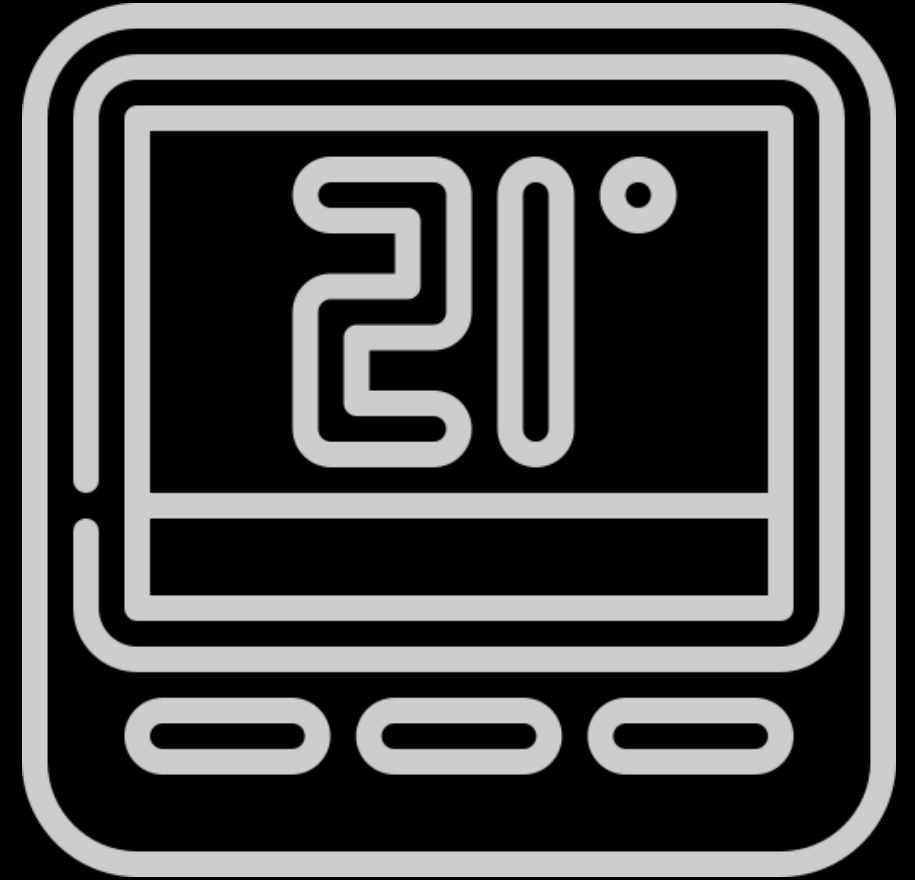
Compatibility Mode	Level	Tables	Columns
PowerBI	1604	6	70

The status bar at the bottom indicates: Model saved. No issues. AB Ln 4 Col 2 INS. Calculation Item "Calculate over shipdate". Dive into DirectLake.

Temperature management

Keep it WARM!

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



Eviction

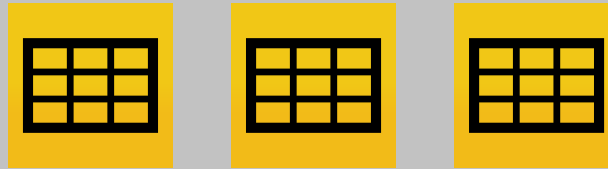
Capacity: F64 / P1

Memory: 25 GB

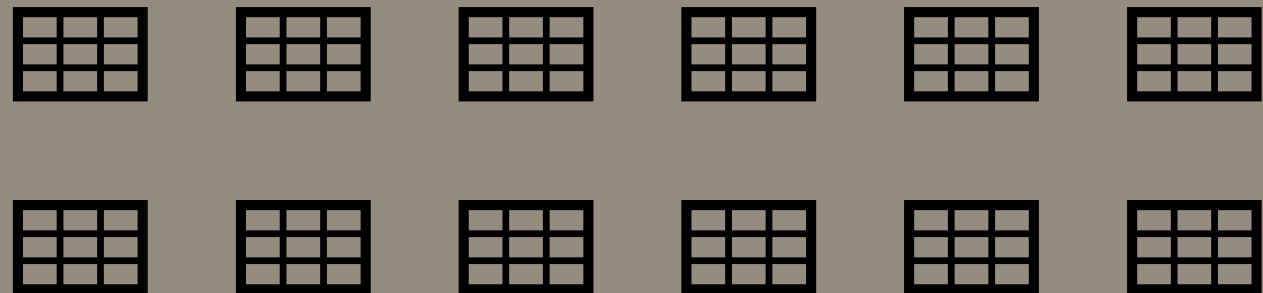
Capacity utilization:

Cool

Active memory



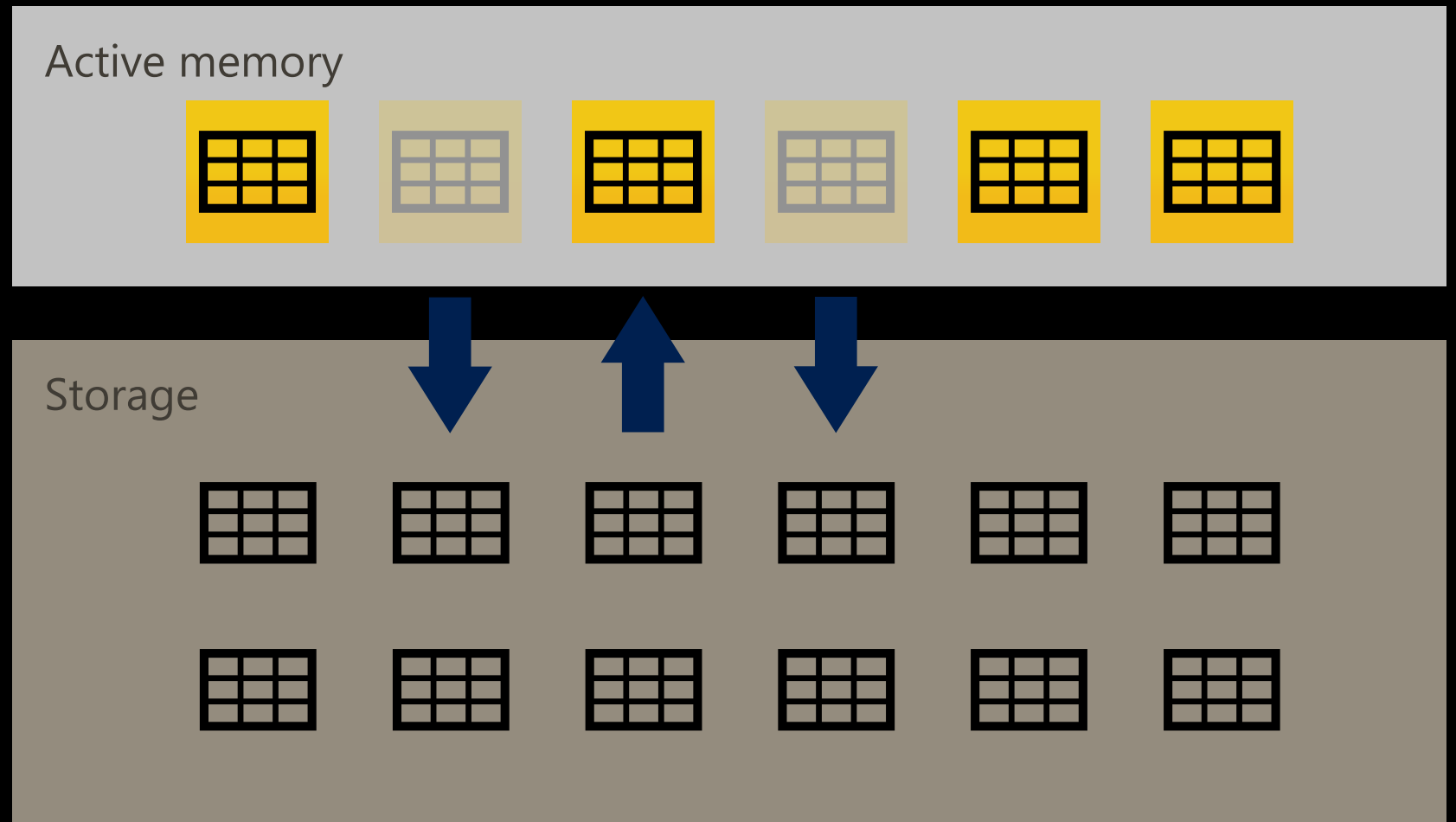
Storage



Eviction

Capacity: F64 / P1
Memory: 25 GB

Capacity utilization:
Warm



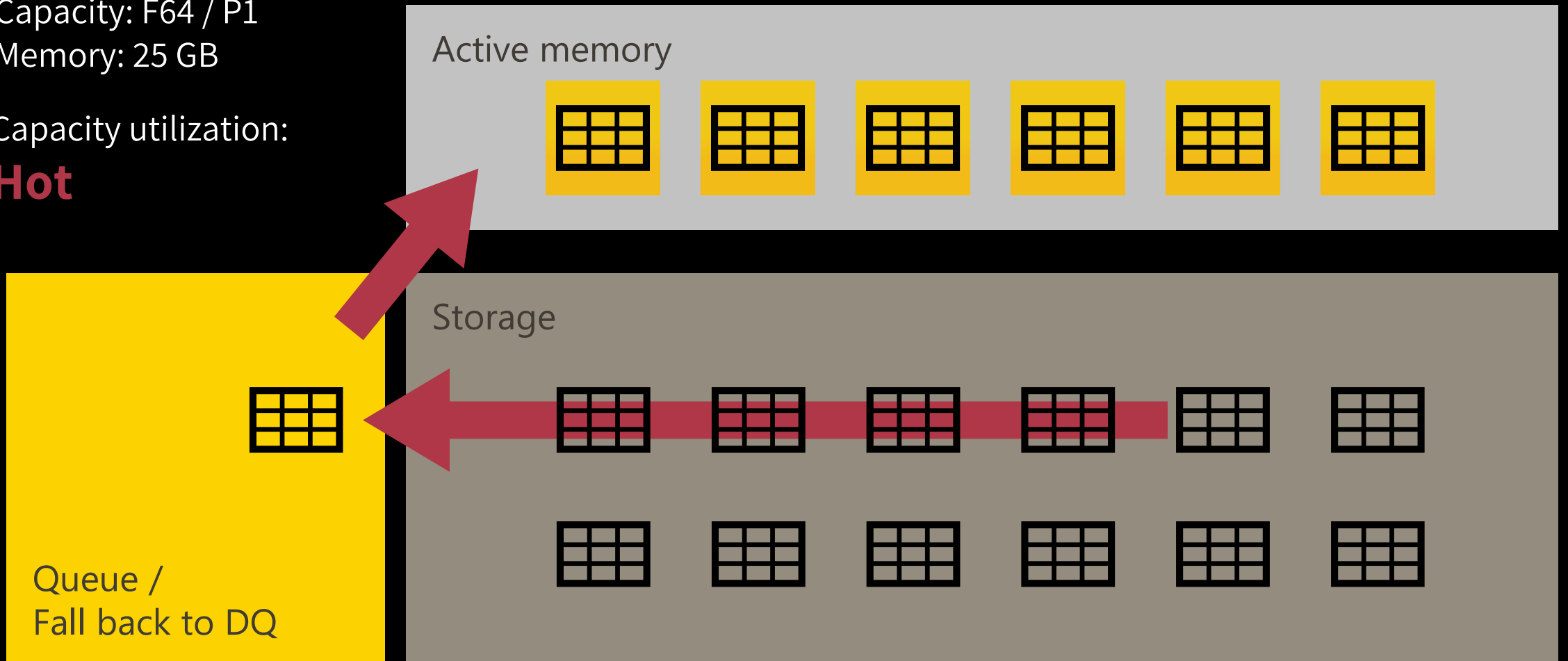
Eviction – queue / fall-back

Capacity: F64 / P1

Memory: 25 GB

Capacity utilization:

Hot



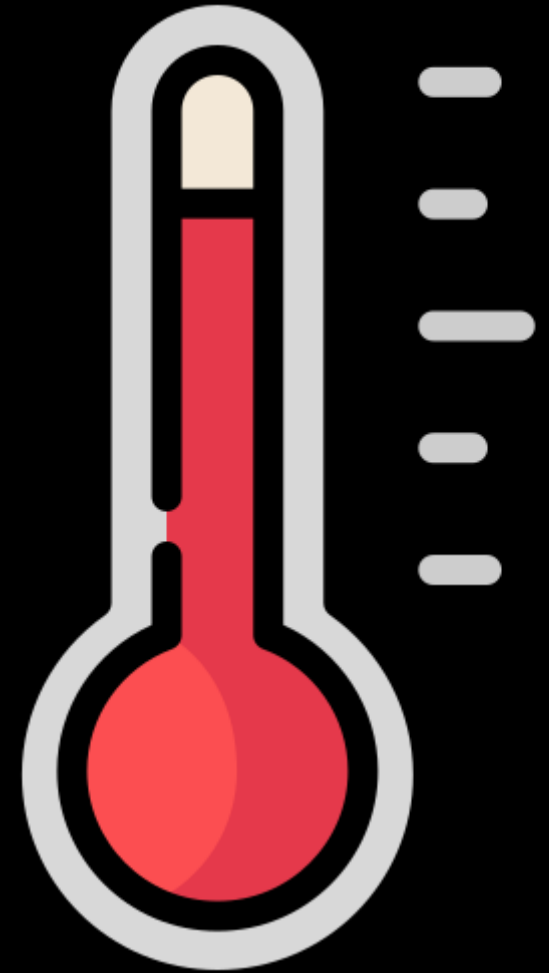
Temperature management

What will be evicted?

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

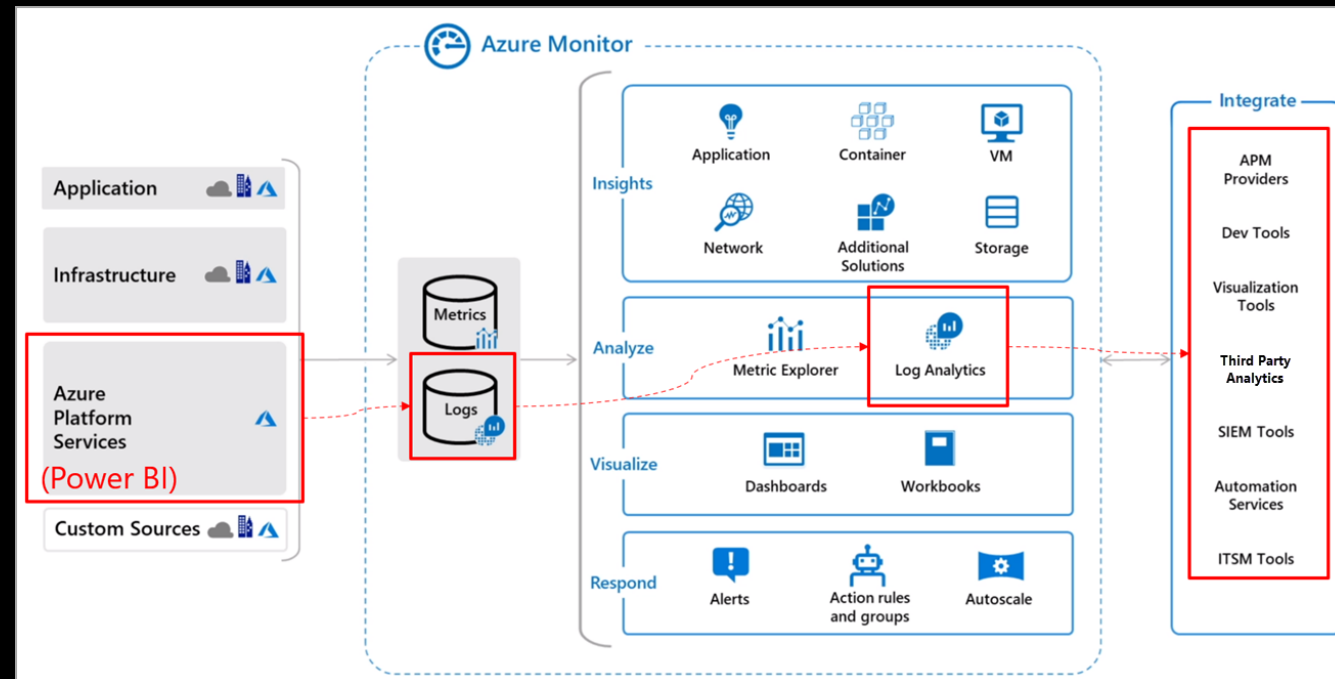
How can you influence that?

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



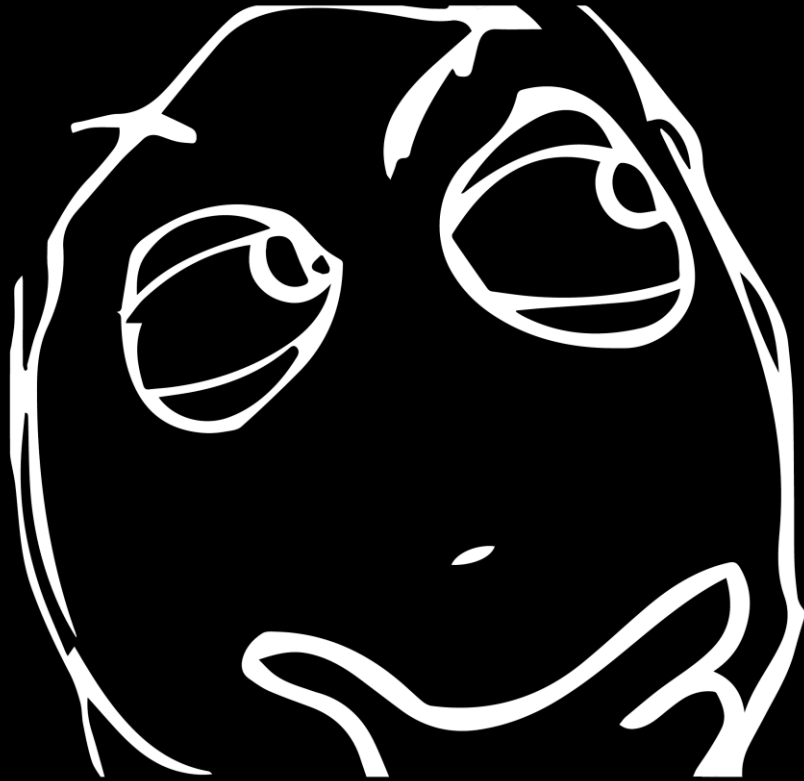
Wrap up

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!
- Consider implement advanced patterns for specific use cases

LET'S
RECAP...

Considerations



IT DEPENDS

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

- There is no OneSecurity yet
RLS / OLS on dataset level is possible
- Consider impact on capacities when falling back to DQ
- Performance is better than DQ
- It is in public preview

Resources

Direct Lake generic documentation

[https://learn.microsoft.com/en-us/power-bi/enterprise/Direct Lake-overview](https://learn.microsoft.com/en-us/power-bi/enterprise/Direct-Lake-overview)

Calculation groups for Direct Lake datasets

<https://powerbi.microsoft.com/en-us/blog/announcing-calculation-groups-for-direct-lake-datasets/>

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/>

Big thanks to Benni and Just for helping us and reference materials 😊