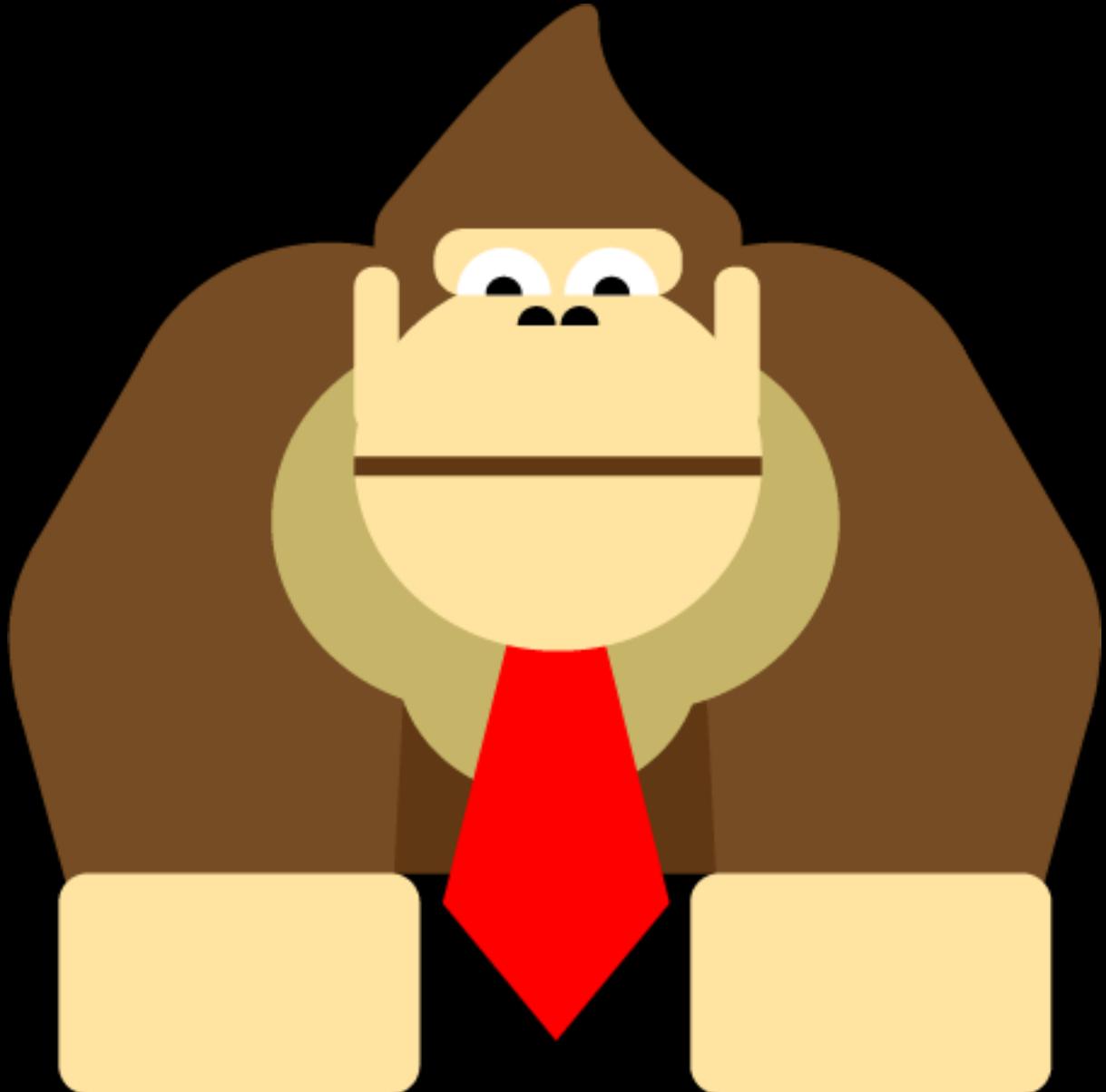




SQLBITS

Analytics at
scale with
Power BI and
Azure Synapse

Insert coins to start





No Wi-Fi hotspots please



Dave Ruijter

Solution Architect Data & Analytics
Blue Rocket Data Consulting & Solutions



-  dave@blue-rocket.it
-  [@DaveRuijter](https://twitter.com/DaveRuijter)
-  linkedin.com/in/DaveRuijter
-  ModernData.ai



Marc Lelijveld

Data & Analytics consultant
Macaw Netherlands



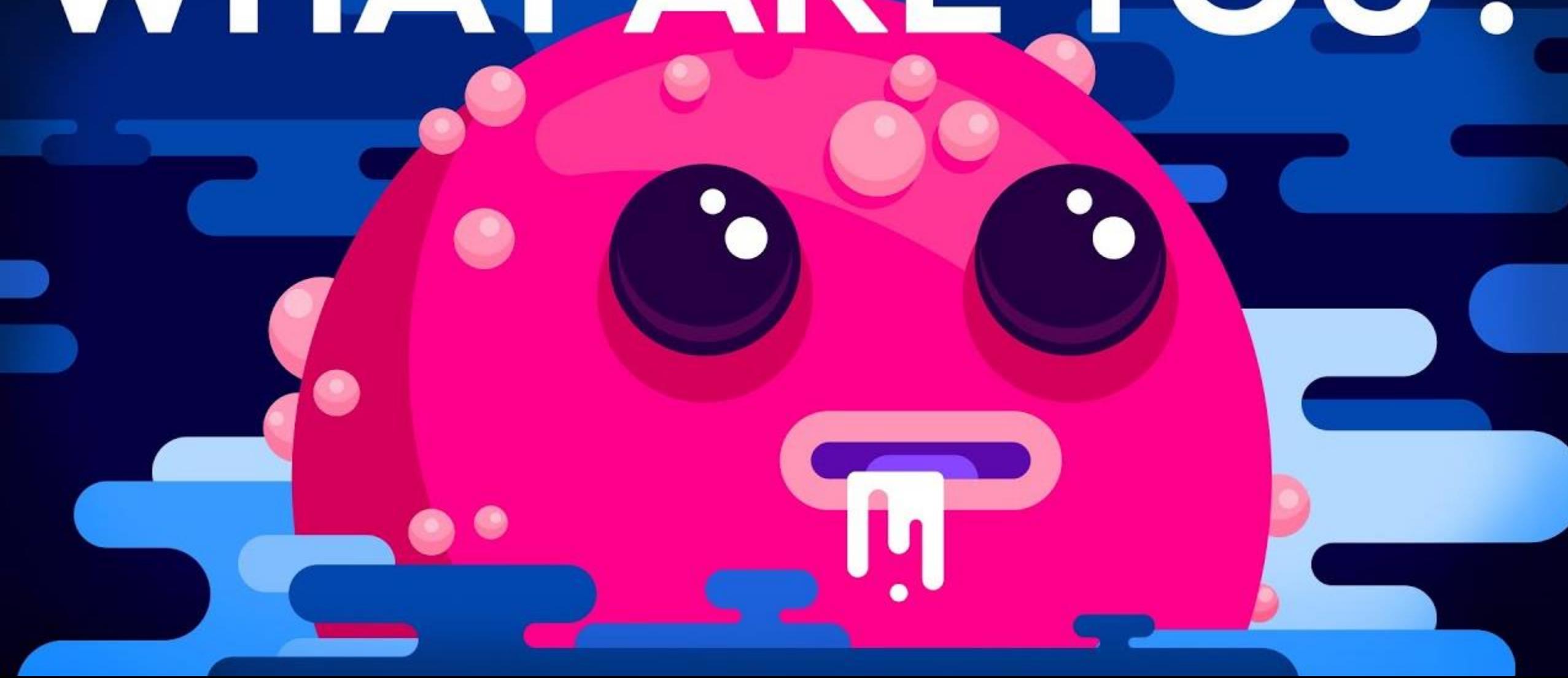
✉ marc.lelijveld@outlook.com

🐦 @MarcLelijveld

linkedin.com/in/MarcLelijveld

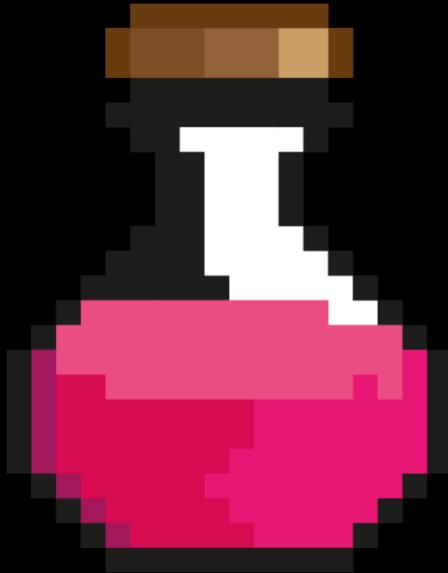
🌐 Data-Marc.com

WHAT ARE YOU?





Break times



Tea and coffee available all day

- 10:30 to 11:00 Refreshments with snacks
- 12:30 to 13:30 Lunch
- 15:00 to 15:30 Afternoon break with snacks

Agenda

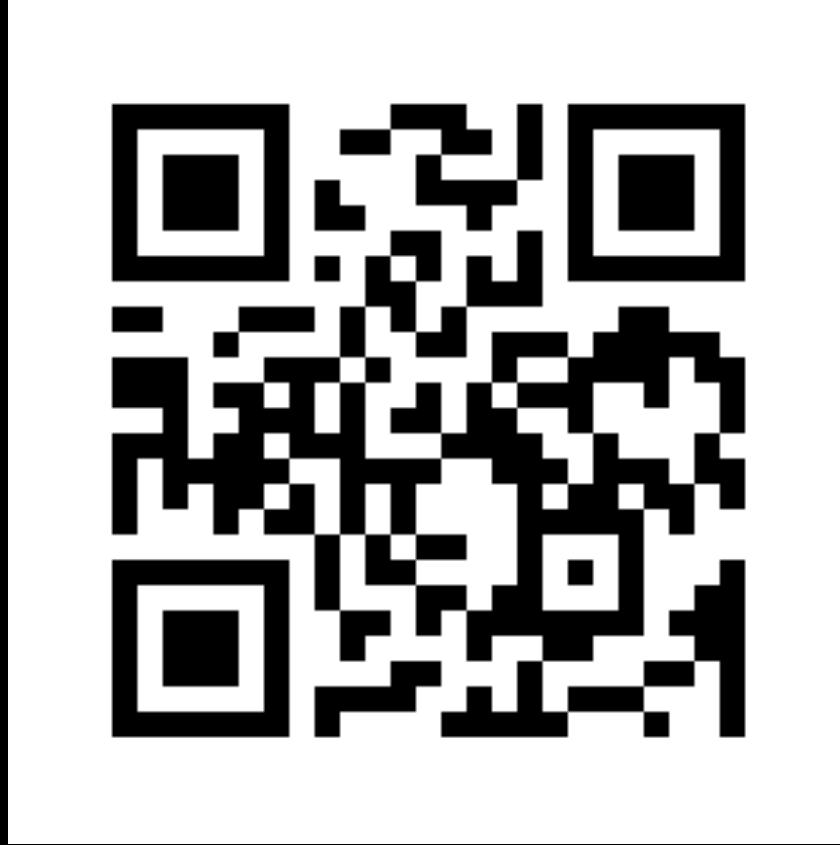


- Challenges
 - Data Platform
 - Power BI
- Deep Dive
 - Data Platform
 - Power BI
- Lunch
- Hybrid Tables & Incremental Refresh
- Refresh Challenges & Orchestration
- Backup & Restore
- Scaling
- Monitoring
- Wrap-up



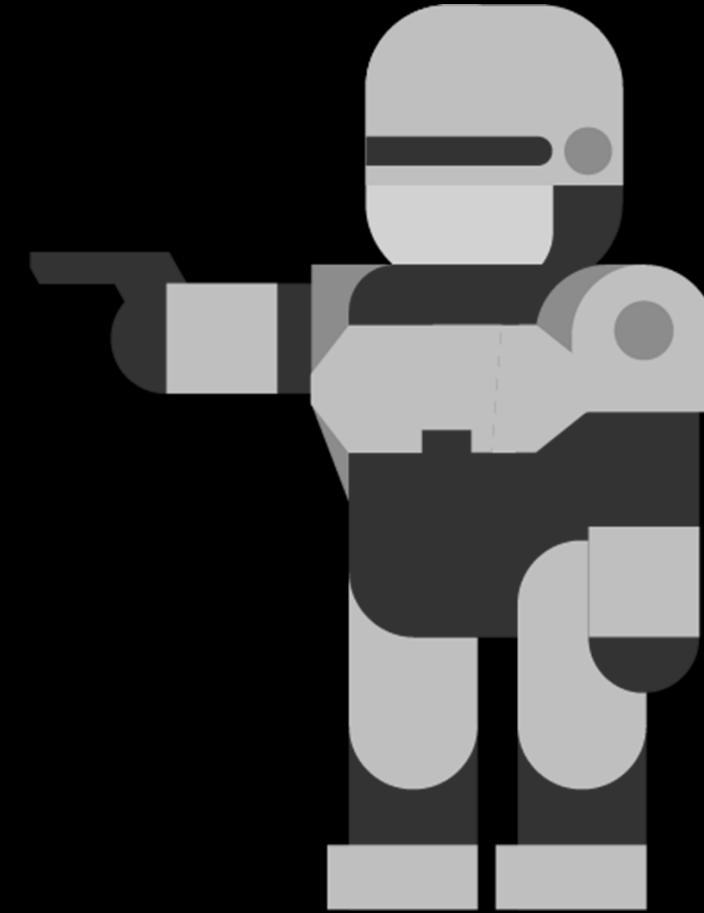
After this session

Design and implement	Orchestrate	Performance	Cost
Better design and implement complex data models, including hybrid tables, aggregations, and combined storage modes (import, DirectQuery , dual).	Orchestrate the end-to-end data processing, with a pipeline chain from data ingest in the data lake house to the incremental Power BI dataset refresh.	Use different techniques to identify performance bottlenecks in your solutions and how to solve those ("does it fold"?).	Implement a cost-efficient solution, that still meets the scalability demands.

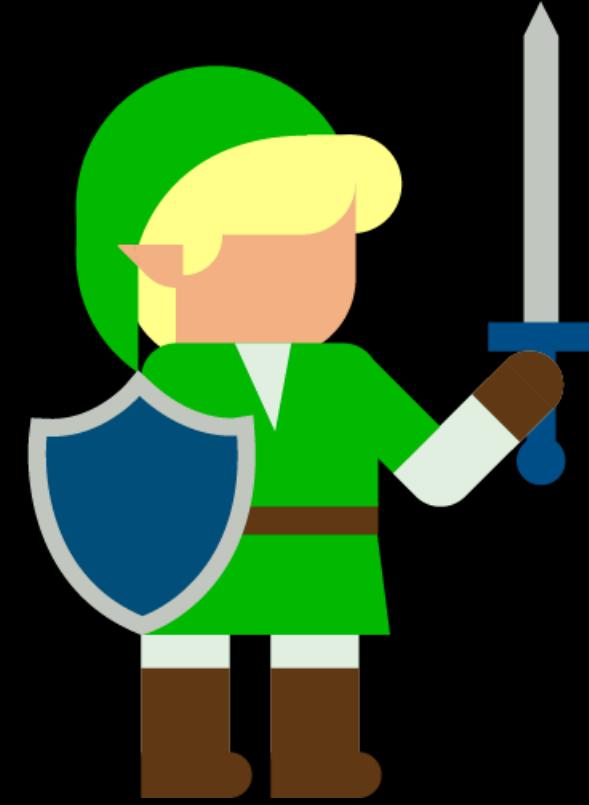


<https://sqlb.it/?6199>

Feedback
Feedback
Feedback

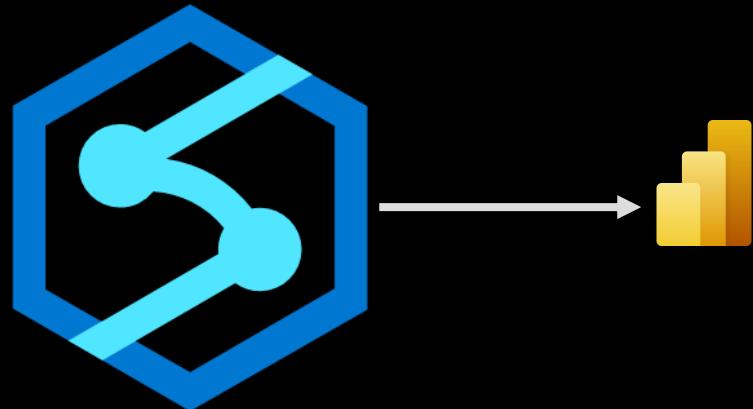


Solution challenges



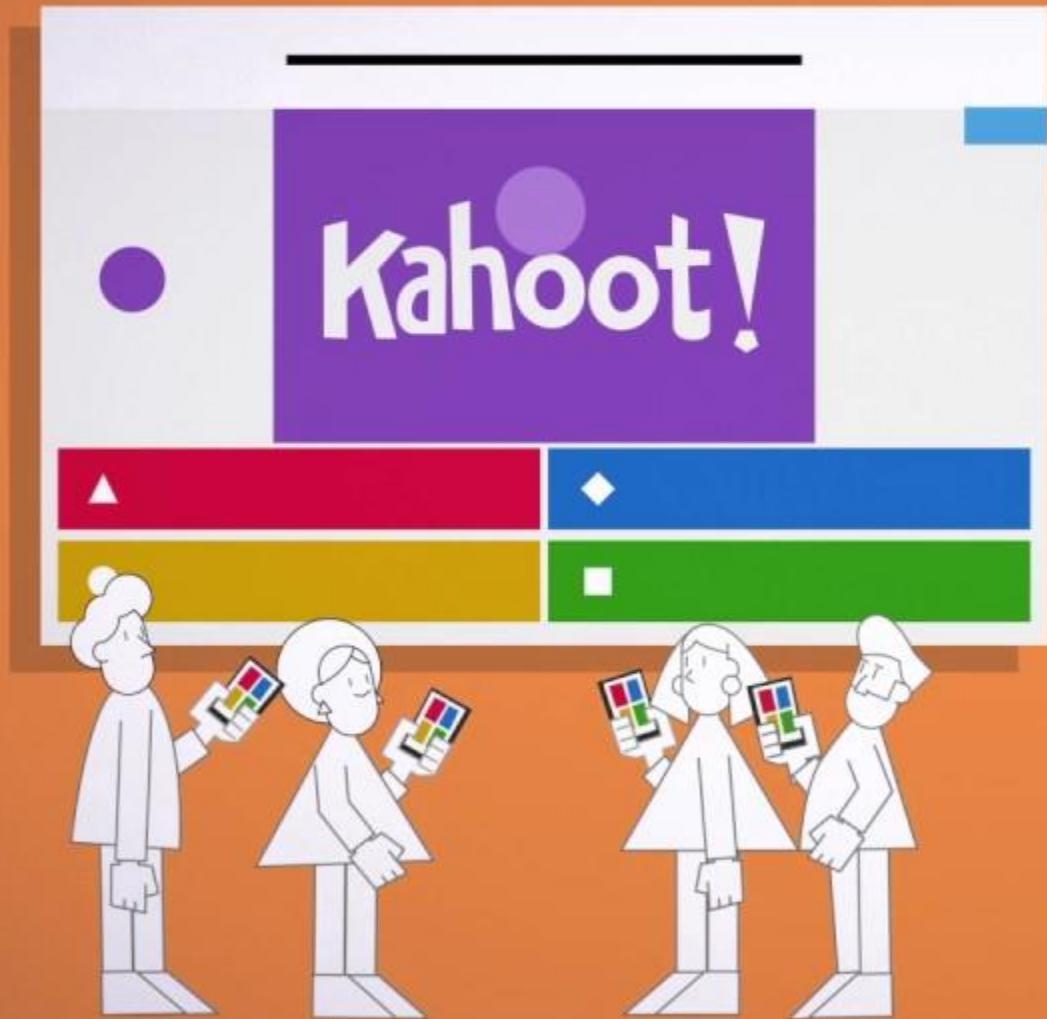
two separate worlds

Data Platform



Power BI







Data Platform Solution Challenges

- Power BI report can't handle the volume of data
- Showing near real-time data in Power BI report





Power BI report can't handle the volume

The screenshot shows the Power BI Desktop interface with the title bar "WWI Sales Model - full IMPORT - Power BI Desktop". The Home tab is selected in the ribbon. A central message box says "Unable to save document" and "Power BI Desktop ran out of memory trying to save the data model". The ribbon also displays "1bn" and "1 Sales". To the right, there are sections for Filters, Visualizations, Fields, and a list of fields including City, Customer, Date, Employee, Movements, Orders, Payment Method, Purchases, Sales, Stock Holdings, and Stock Item. The status bar at the bottom right shows "Storage Mode: Mixed".



Demo

Power BI report can't handle the volume of data



Should I put everything on
DirectQuery instead?





DirectQuery limitations

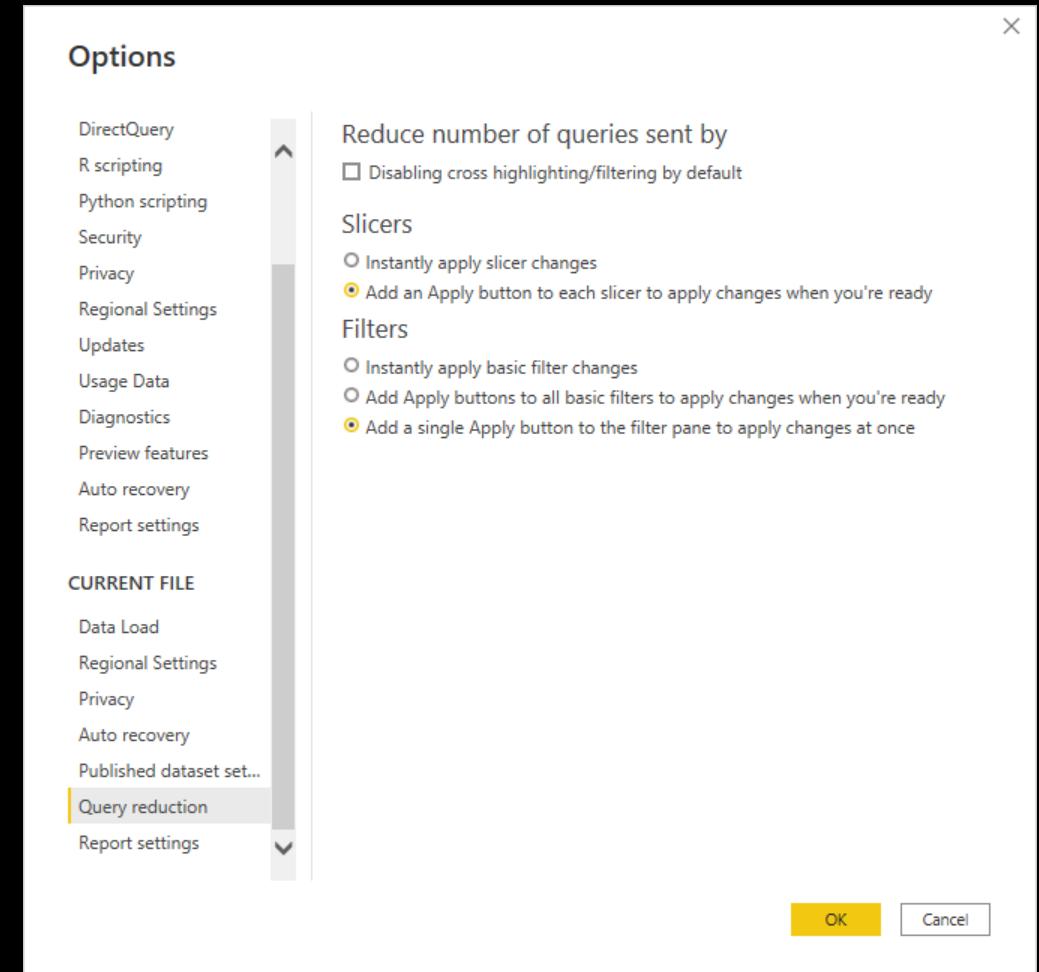
- Limited Power Query capabilities
- DirectQuery != streaming / live! Front-end still caches data
- No built-in date hierarchy (year/quarter/month/day)
- Lowest granularity data is seconds (no milliseconds)
- No parent-child support in DAX with *PATH()*
- Slower end user performance
- 1M row per query
- DAX limitations, only simple calculations possible

DirectQuery query reduction

Consider requesting to click **Apply** before queries are executed to the source

Applies to

- Slicers
- Filters (filter pane)





Showing near real-time data in
Power BI



Showing near real-time data in Power BI

- Refresh takes too long
- Poor end-user performance on DirectQuery
- Streaming datasets only allow one table
- Potentially queries are not foldable, therefore incremental refresh does not work (depending on source)



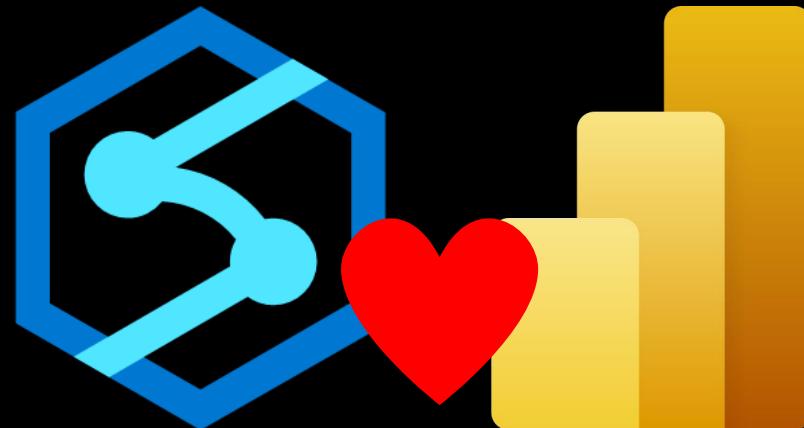
Demo

Showing near real-time data in Power BI report



Data Platform Solution Challenges

- Optimize Power BI model
- Use Aggregations
- Use Hybrid Table





Power BI Solution Challenges

- Loading data from various sources (flat files, databases, APIs)
- Some sources are manually maintained (like mapping tables)
- Data stored on decentralized storages, like SharePoint pages.
- Data from source systems are exported, rather than connected to analytical systems
- Store historical data in Power BI for trend analysis







Demo

Various manually maintained sources



OrderDate
Last ▾ 1 Select ▾
No filters applied

Product Category
All ▾

CountryRegion
 Canada
 United Kingdom
 United States

StateProvince, City
 Alberta
 Arizona
 British Columbia
 Brunswick

\$708,69K

\$ Sales

2.087

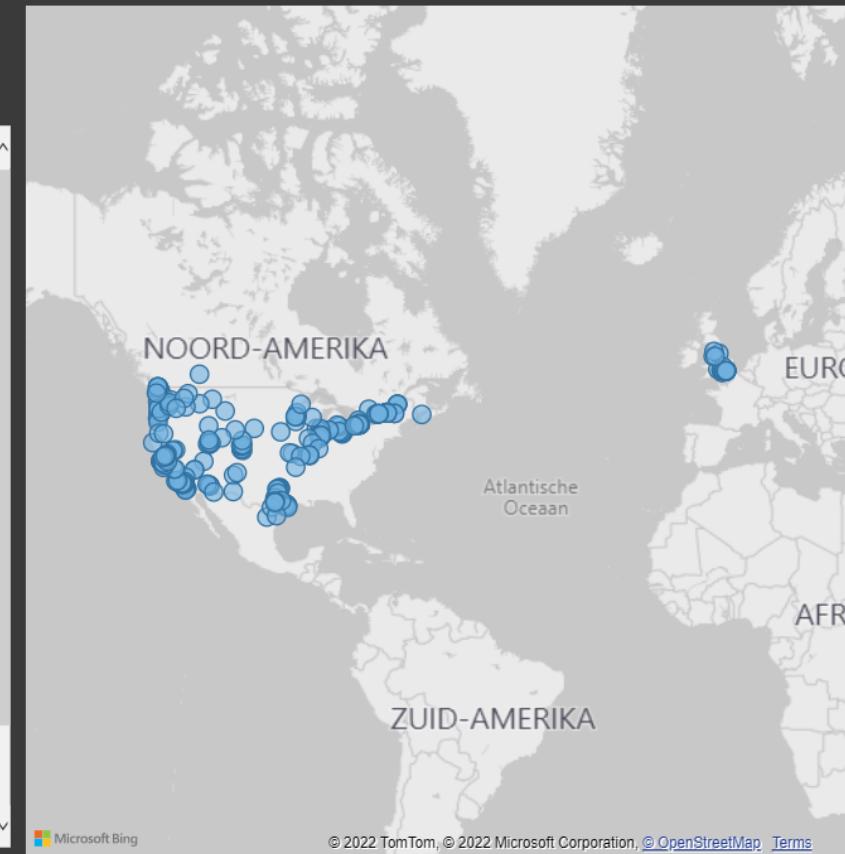
Order quantity

32

Orders

Product Category	# Order quantity	\$ Sales	% GT \$ Sales	\$ Price Avg per product
Touring Bikes	252	\$220.655,38	31,14%	\$875,62
Road Bikes	222	\$183.130,30	25,84%	\$824,91
Mountain Bikes	209	\$170.825,89	24,10%	\$817,35
Mountain Frames	128	\$54.949,60	7,75%	\$429,29
Road Frames	60	\$24.346,58	3,44%	\$405,78
Touring Frames	39	\$19.066,26	2,69%	\$488,88
Jerseys	230	\$7.017,88	0,99%	\$30,51
Vests	121	\$4.309,90	0,61%	\$35,62
Cranksets	22	\$3.968,87	0,56%	\$180,40
Shorts	80	\$3.299,80	0,47%	\$41,25
Pedals	84	\$2.996,50	0,42%	\$35,67
Helmets	124	\$2.523,88	0,36%	\$20,35
Bike Racks	32	\$2.304,00	0,33%	\$72,00
Hydration Packs	50	\$1.649,70	0,23%	\$32,99
Bottom Brackets	22	\$1.320,17	0,19%	\$60,01
Deraileurs	21	\$1.296,63	0,18%	\$61,74
Handlebars	27	\$1.192,97	0,17%	\$44,18
Saddles	39	\$1.010,30	0,14%	\$25,91
Gloves	57	\$837,56	0,12%	\$14,69
Brakes	13	\$830,70	0,12%	\$63,90
Socks	66	\$345,21	0,05%	\$5,23
Caps	52	\$277,36	0,04%	\$5,33
Cleaners	55	\$251,88	0,04%	\$4,58
Total	2.087	\$708.690,15	100,00%	\$339,57

\$ Sales by PostalCode

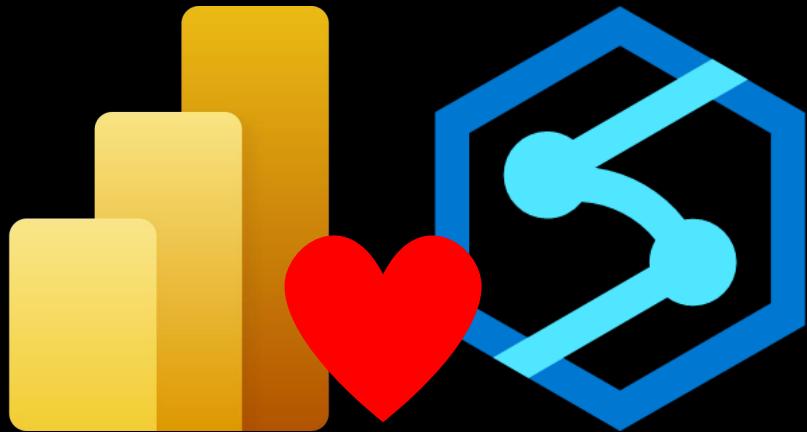


© 2022 TomTom, © 2022 Microsoft Corporation, [OpenStreetMap](#), [Terms](#)



Power BI Solution Challenges

- Use Synapse Analytics to ingest into a data lakehouse
- Use Layered approach: Bronze, Silver & Gold



Better together

Data platform deep dive





Improvement areas

- Ingesting data from APIs using Synapse Analytics
- Store (historical!) data in the delta lakehouse architecture



Ingesting data from APIs using Synapse Analytics

- Pipeline (same as ADF)
- Data Flow (same as ADF) *not to be confused with dataflows in PBI*
- Spark Notebook (4 languages available)
- Wrangling Dataflows (Same as ADF)



Hybrid data integration

Cloud native ETL/ELT

95+ connectors available

Secure connectivity to on-premise data sources, other clouds, and SaaS applications

Code-first and low/no code design interfaces

Schedule and Event based triggering





Code-free data wrangling

No/low-code data transformation

Excel-like interface is familiar to users

Transform data to desired shape completely visually

Operationalize into pipelines

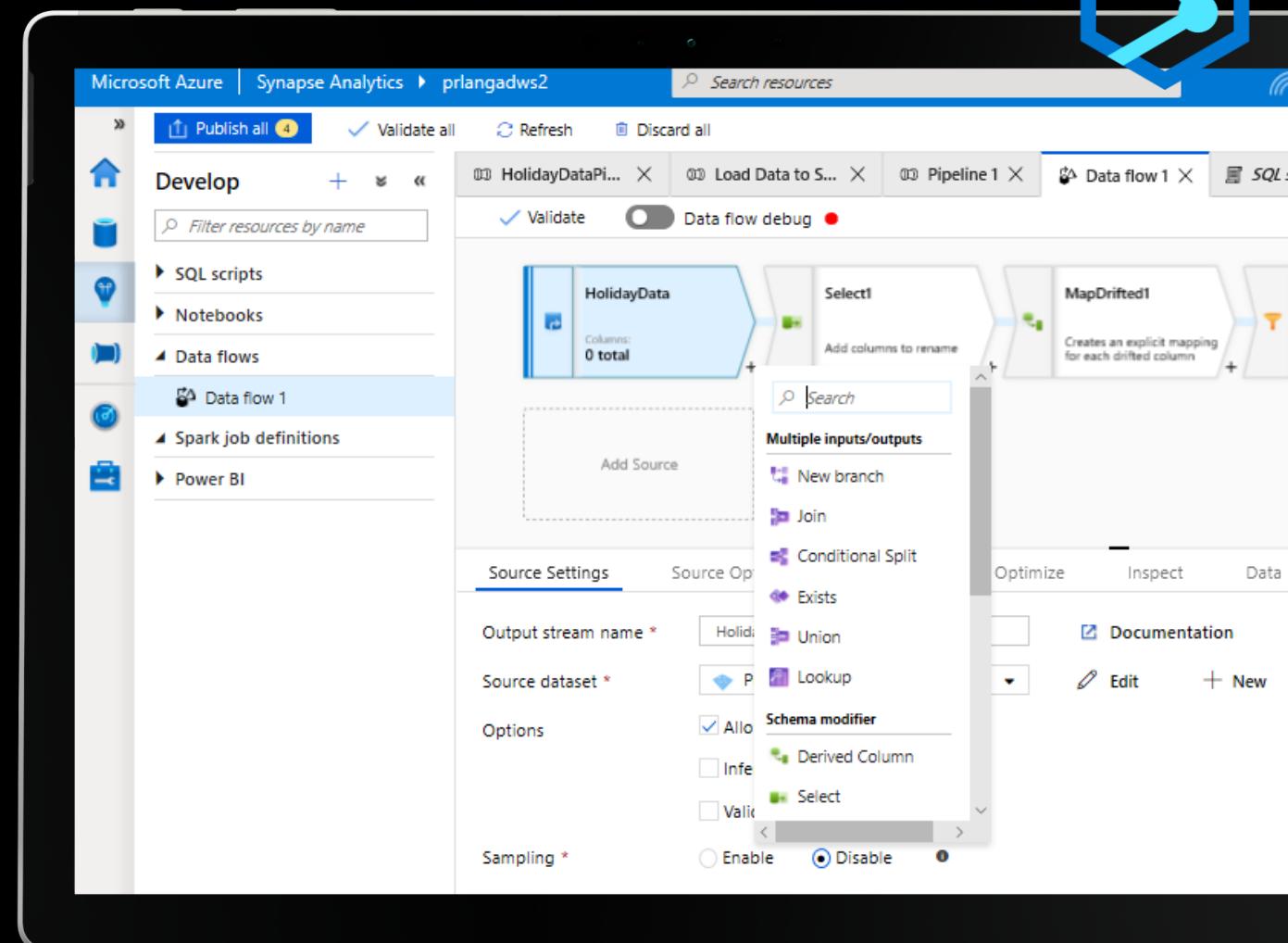
The screenshot shows the Microsoft Azure Synapse Analytics Power Query Editor interface. The top navigation bar includes 'Microsoft Azure', 'Synapse Analytics', 'wsazuresynapseanalytics', and a search bar. Below the navigation is a toolbar with various icons for validation, publishing, and discarding. The main workspace is titled 'PQSalesPrep' and shows a 'Settings' section with a note about Power Query M functions. The 'Home' tab is selected, displaying a ribbon of tools: Enter data, Options, Manage parameters, Refresh, Advanced editor, Properties, Choose columns, Remove columns, Manage columns, Keep rows, Remove rows, Reduce rows, Sort, Split column, Group by, Replace values, and Transform. To the right of the ribbon is a data preview pane showing a table with 17 rows and 8 columns. The columns are labeled: ab storeId, ab productCode, 12 quantity, 1.2 logQuantity, ab advertising, ab price, ab weekStarting, and ab id. The data consists of various surface.go entries with different quantities and prices. At the bottom of the preview pane, it says 'Columns: 8 Rows: 99+'. The overall interface is clean and modern, designed for data wrangling and transformation.

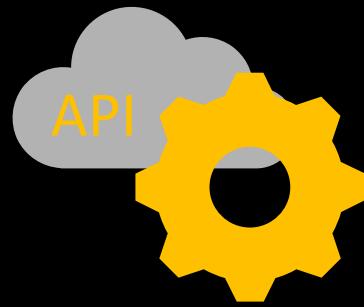


Hybrid data integration

Data Flows

not to be confused with dataflows in Power BI





— Demo API ingest



Store (historical!) data in the lake

- The Layered approach: Bronze, Silver & Gold
- Keep original raw data, build up history -> **bronze**
- Cleanse and refine data, standard file format -> **silver**
- Aggregate, prepare, transform, merge, make start schema -> **gold**



Store (historical!) data in the lake

- SQL The Layered approach:
Bronze, Silver & Gold

- SQL Keep original raw data, build up history -> **bronze**
- SQL Cleanse and refine data, standard file format -> **silver**
- SQL Aggregate, prepare, transform, merge, make start schema -> **gold**

Bronze



Raw data
All history,
system
replayable

Silver



Apply
metadata
Protect data
(GDPR)

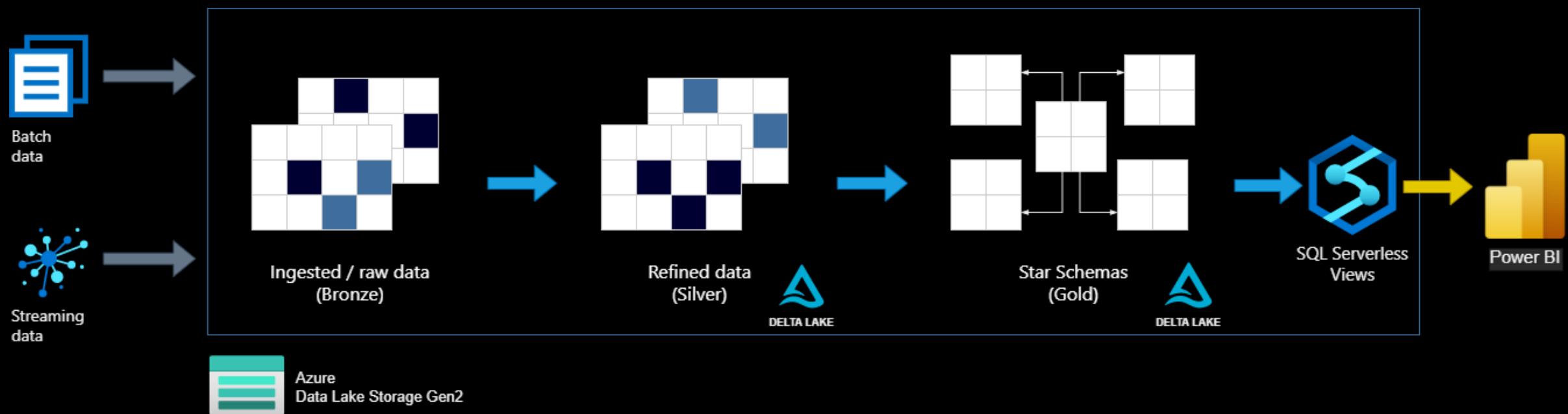
Gold



Current &
historical
view



Using Delta Lakehouse with layered approach





Warm-up time of Serverless SQL pools

SQL requests

Refresh Edit columns

Filter by keyword

Local time : **Last 24 hours**

Status : **All**

Pool : **Built-in**

Add filter

Showing 1 - 4 of 4 items

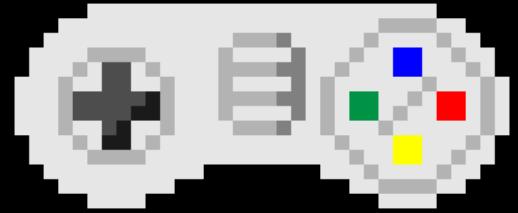
Request ID ↑	Request content ↑	Submit time ↑	Duration	Data processed
44196195	SELECT TOP (1000... More	3/6/22, 10:45:14 AM	14 sec	5.12 GB
44190548	SELECT TOP (1000... More	3/6/22, 10:43:13 AM	16 sec	4.34 GB
44187683	SELECT TOP (1000... More	3/6/22, 10:42:32 AM	14 sec	6.12 GB
44178591	SELECT TOP (1000... More	3/6/22, 10:39:55 AM	20 sec	6.11 GB

Sales Territory	\$ Sales
External	\$12,540,565,587.63
Far West	\$130,724,597,536.11
Great Lakes	\$137,062,180,086.18
Mideast	\$170,472,095,945.05
New England	\$58,853,530,841.09
Plains	\$154,405,133,753.92
Rocky Mountain	\$72,387,209,160.70
Southeast	\$258,110,199,285.25
Southwest	\$141,628,285,606.20
Total	\$1,136,183,797,802.13



Best practices for serverless SQL pools

-  Azure AD Pass-through Authentication performance <= shared access signature credentials
-  Colocate
-  Same region
-  Convert large CSV and JSON files to Parquet
-  Try to optimize storage layout by using partitioning and keeping your files in the range between 100 MB and 10 GB
-  Use appropriate data types (smallest, integer-based, varchar)
-  Use filename and filepath functions to target specific partitions



Demo Data Platform
Lakehouse architecture

Better together

Power BI deep dive





Improvement areas

- Data model
- Query Folding
- Aggregations
- Storage modes
- Hybrid tables



But before we start changing our solution, let's measure...

- Refresh durations
- Model Size
- Vertipaq Analyzer
- Performance Analyzer
- Query folding applied?

Perfect E2E.pbix			
Total Size	Last Data Refresh	Analysis Date	
74,89 MB	1-3-2022 20:05:54 +01:00	1-3-2022 20:05:55 +01:00	
Compatibility	Tables	Columns	Server
1550	7	124	localhost:60032



Performance analyzer in Power BI Desktop

The screenshot shows the 'Performance analyzer' window in Power BI Desktop. It has a toolbar with 'Performance analyzer' (highlighted), 'Sync slicers', 'Start recording', 'Refresh visuals', 'Stop', 'Clear', and 'Export'. A red box highlights the 'Duration (ms)' column header in a table. The table lists various tasks with their execution times:

Name	Duration (ms)
0.0%	2279
Simple Image	1440
Net Sales vs "What If" Analysis	4331
OneNote	2391
"What If" Analysis Forecast	2543
Changed a slicer	50
What If...	-
Return Rate	128
Net Sales (Forecast)	1028
Extra Profit	1618
Card	2046
"What If" Analysis Forecast	1425
Returns	1890
OneNote	2249
Button	1155
Last Refresh: Jun 30th, 2019 / ...	1723
\$30,772	1722
+17.1%	1723
What If...	398
"What If" Analysis Forecast	2107
Simple Image	1528
What If...	1715
"What If" Analysis Forecast	388
Simple Image	2721

Learn more about optimizing your report's performance on our [support site](#). Find help tuning your report from specialist Power BI partners on [AppSource](#).

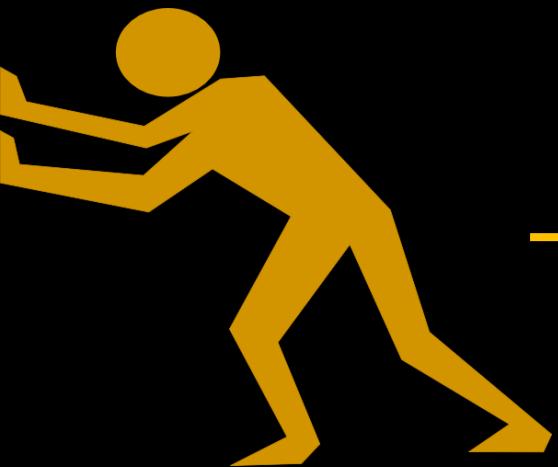
- DAX Query
- Visual Display
- Other
 - Preparing queries
 - Waiting for other visuals to complete
 - Other background processes



Vertipaq analyzer

See where your data volume is

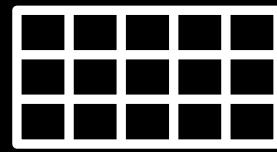
VertiPaq Analyzer Metrics									
Tables		Cardinality	Table Size	Col Size	Data	Dictionary	Hier Size	Encoding	Data Type
↳ Sales Agg		1.933.444	55.428.208	55.422.984	29.854.056	21.688.736	3.880.192	Many	-
Total Including Tax		119.940	55.428.208	11.252.400	4.973.680	5.319.152	959.568	HASH	Double
Tax Amount		119.137	55.428.208	11.241.312	4.972.888	5.315.288	953.136	HASH	Double
Total Excluding Tax		118.952	55.428.208	11.241.120	4.973.680	5.315.776	951.664	HASH	Double
Profit		113.830	55.428.208	11.139.328	4.975.824	5.252.816	910.688	HASH	Double
Invoice Date Key		1.444	55.428.208	3.109.056	3.013.344	84.112	11.600	HASH	DateTime
Delivery Date Key		1.443	55.428.208	3.109.016	3.013.344	84.072	11.600	HASH	DateTime
Count Rows		9.722	55.428.208	2.356.128	1.974.104	304.200	77.824	HASH	Int64
Customer Key		403	55.428.208	1.929.700	1.916.016	10.420	3.264	HASH	Int64
Salesperson Key		101	55.428.208	44.804	41.176	2.780	848	HASH	Int64
RowNumber-2662979B-1795-4F74-8F37-6A1BA8059B61		0	55.428.208	120	0	120	0	VALUE	Int64
↳ Date		3.287	23.045.462	22.981.430	73.096	22.805.342	102.992	Many	-
↳ Customer		403	6.436.260	6.436.260	2.448	6.419.380	14.432	Many	-
↳ Employee		213	2.157.356	2.157.356	1.120	2.151.460	4.776	Many	-
↳ StockItem		0	8.576	8.576	160	8.416	0	HASH	-
↳ Sales		0	8.392	8.352	176	8.176	0	HASH	-
↳ sales_model Employee		0	8.352	8.352	176	8.176	0	HASH	-
↳ City		0	6.496	6.496	112	6.384	0	HASH	-



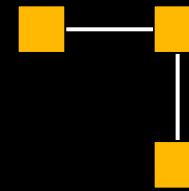
— Performance & Vertipaq Analyzer



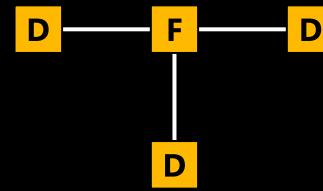
Care about the data model!



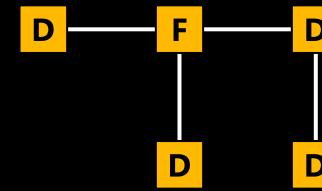
Flat



Normalized
(nth normal form)



Star



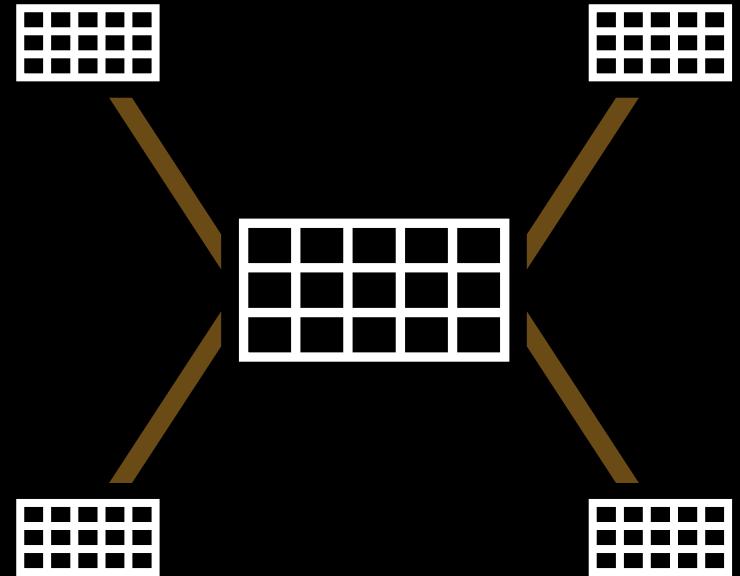
Snowflake



Star schema all the things!

Facts

- Contains numerical information about a business process or items to be aggregated
- Aggregations provide totals, averages, etc.
Power BI implements these using Measures
- Usefulness limited without context
Context is provided by dimensions that slice the data



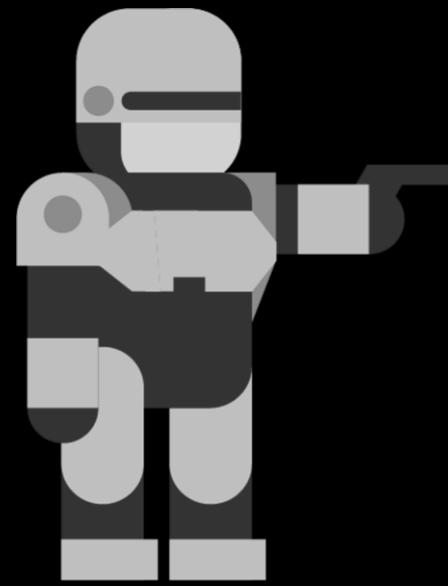
Dimensions

- Contains descriptive information that define how a fact should roll up.
- Examples: Date, Month, Customer, Geography, Product, Payment type.
- Without dimensions there is no context.
- Also called: Lookup table on steroids.



Care about the data model

	Flat table	Normalized	Star schema	Snowflake
Performance for analytics	Low	Medium	High	High
Development effort	Low	High	High	High
Query volume and complexity	Low volume Low complexity	High volume Low complexity	Low volume High complexity	Low volume High complexity
Intended for	No database	CRM / ERP / Applications	Analytical systems / data warehouses	Analytical systems / data warehouses
Compression	Row	Row	Column	Column



LUNCH BREAK





Aggregations

Benefits

- Report visualizations are faster
- Balanced architecture by combining DirectQuery and Import storage modes

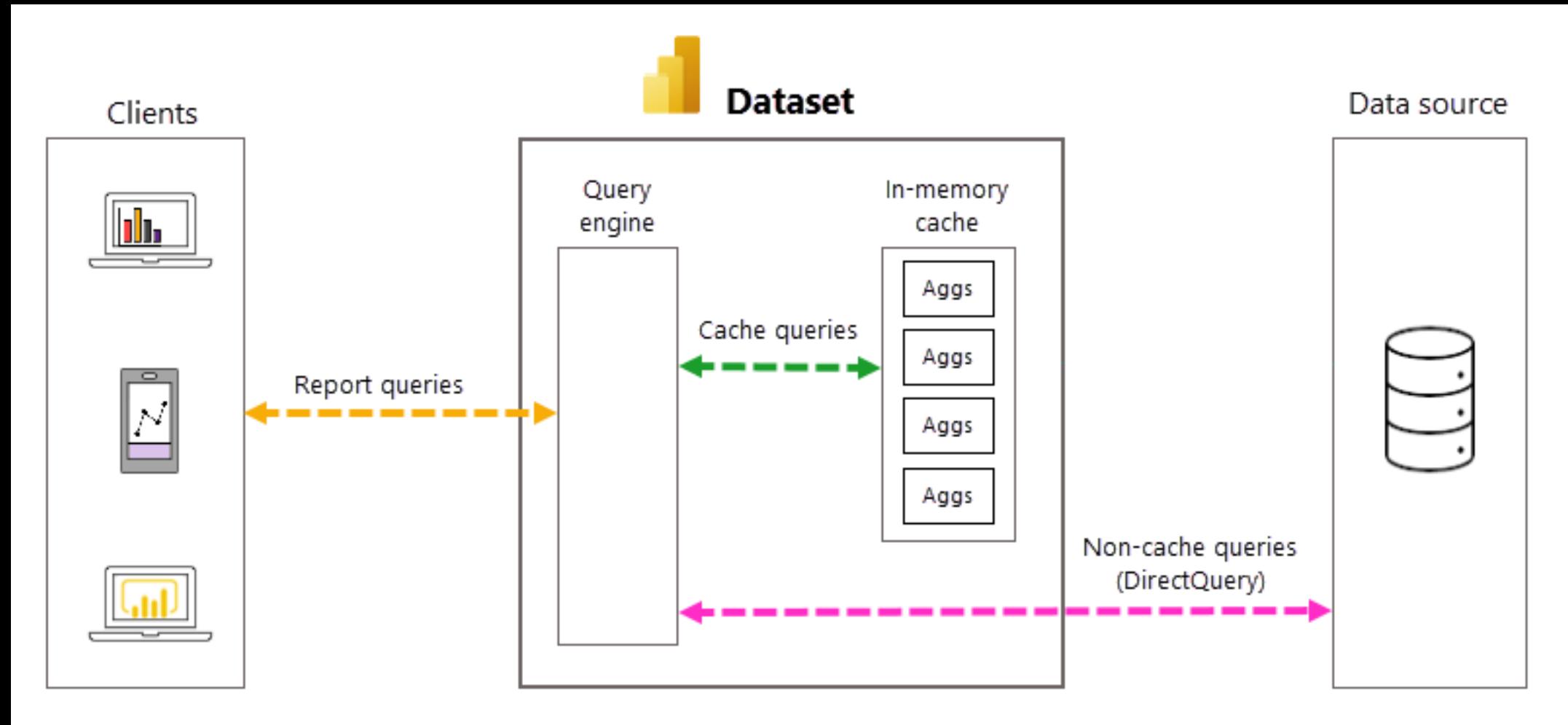
Store data at a higher level of granularity than the original table

The following aggregations are available:
count, groupby, max, min, sum, and count table rows

- Aggregated data is stored in-memory (imported), details are accessed through DirectQuery
- You can create the aggregated table in the Data Transformations (Power Query) or in your source (preferred)



Aggregations





Aggregations

Aggregation, that hit based on relationships, require *regular* relationships.

Regular relationships include the following storage mode combinations, where both tables are from a single source group:

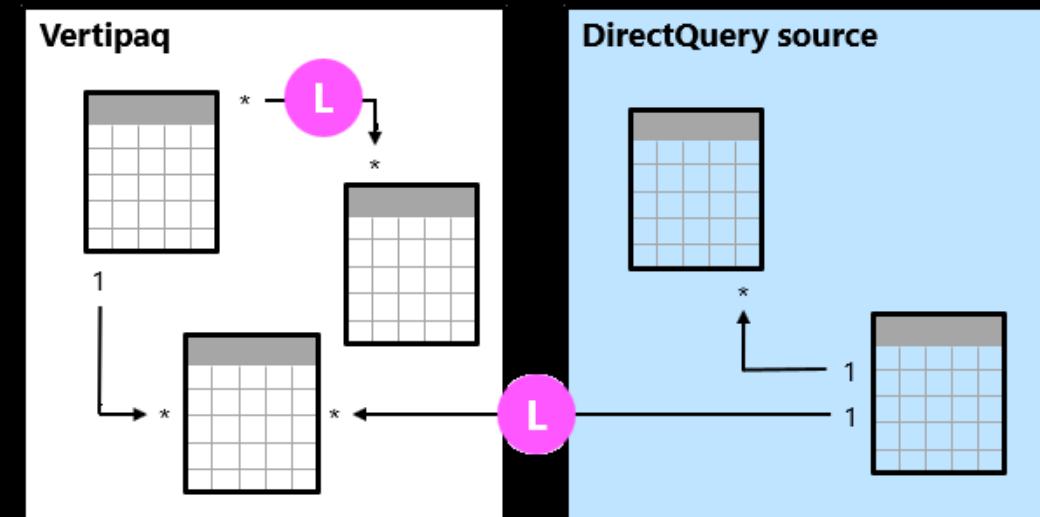
Table on the many side	Table on the 1 side
Dual	Dual
Import	Import or Dual
DirectQuery	DirectQuery or Dual



Relationships and storage modes

A model relationship is **limited** when there's no guaranteed "one" side. It can be the case for three reasons:

- The relationship uses a Many-to-many cardinality type (even if one or both columns contain unique values)
- The storage mode combination is Import and DirectQuery
- The relationship is cross source group





Impact of limited relationships

Cross source group relationships have performance implications.

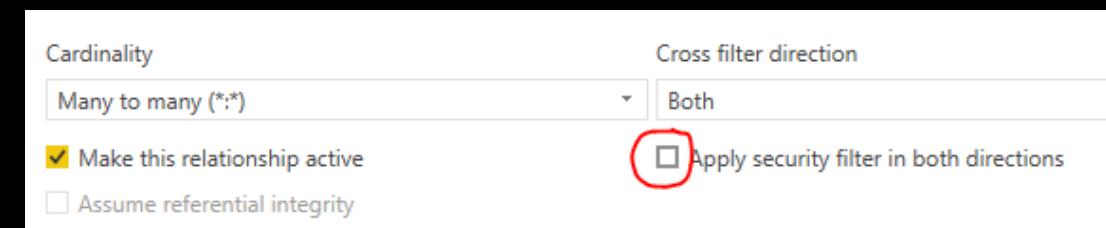
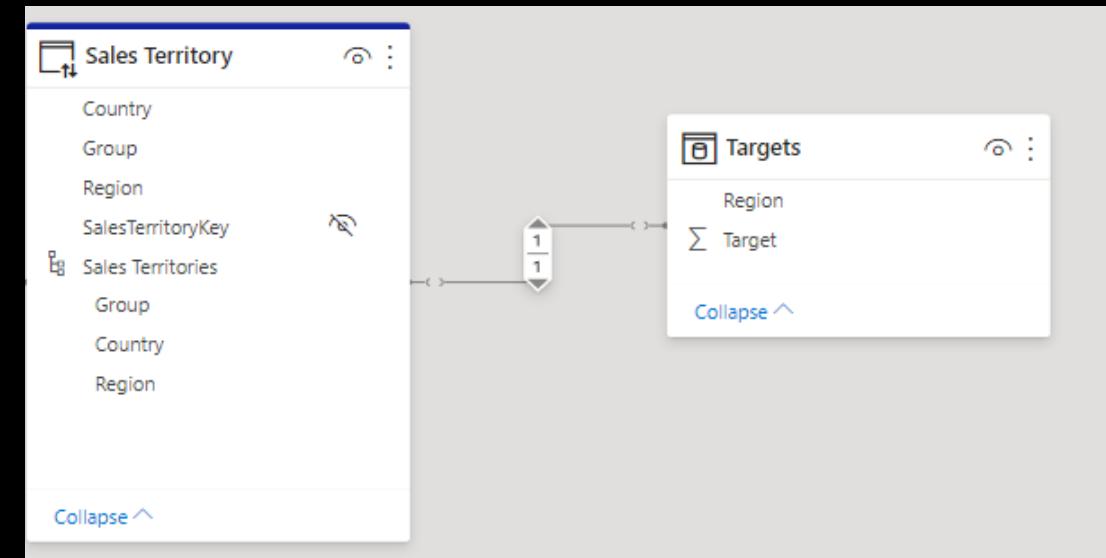
Limited optimization: joins are done on-demand for DirectQuery.

No blank rows: table joins are achieved by using INNER JOIN.

- Blank rows are not added for referential integrity violations

Additional restrictions:

- RELATED DAX function cannot be used to retrieve the 'one' side of the relationship
- Enforcing RLS requires you to check the following checkbox





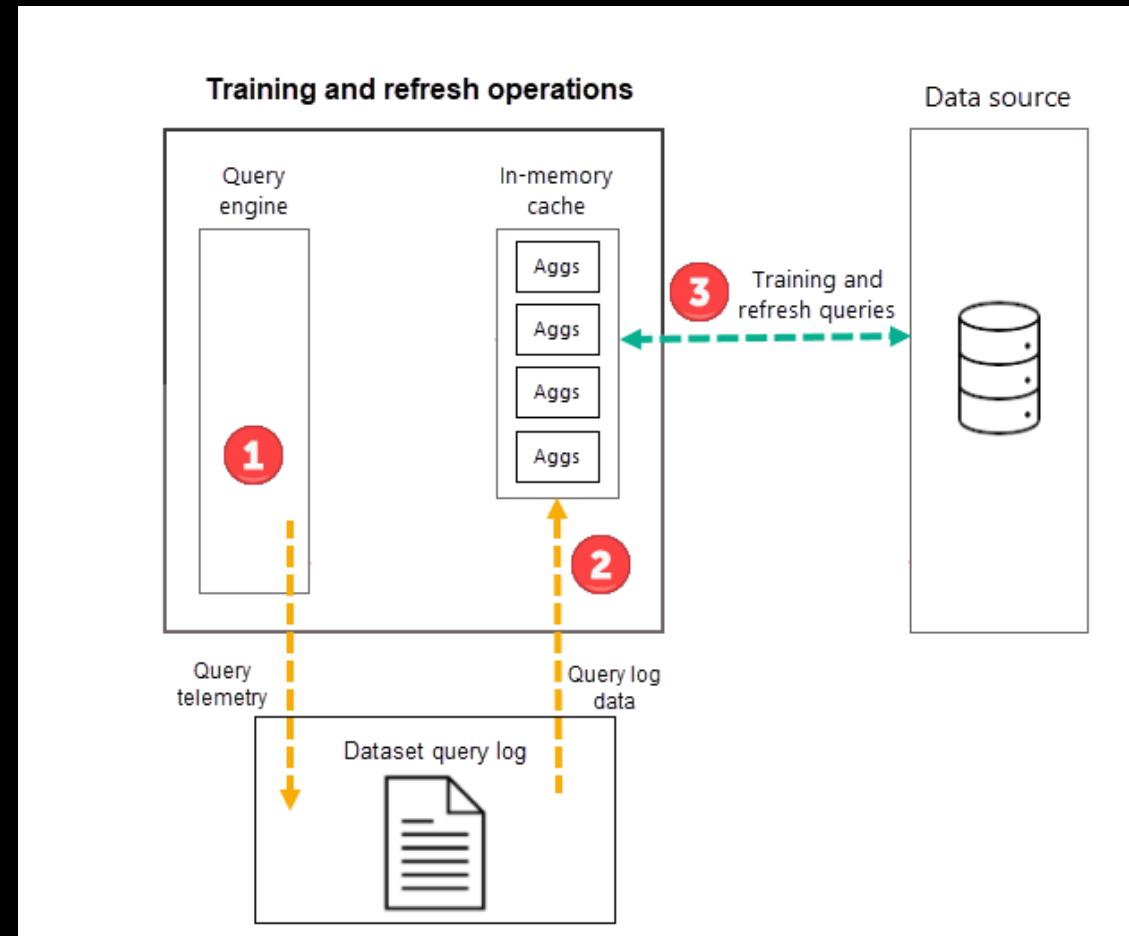
Automatic Aggregations

Power BI Premium per User, Premium Capacity and Embedded datasets

Automatic aggregations based on Query logs (7 days)

Supported sources during preview:

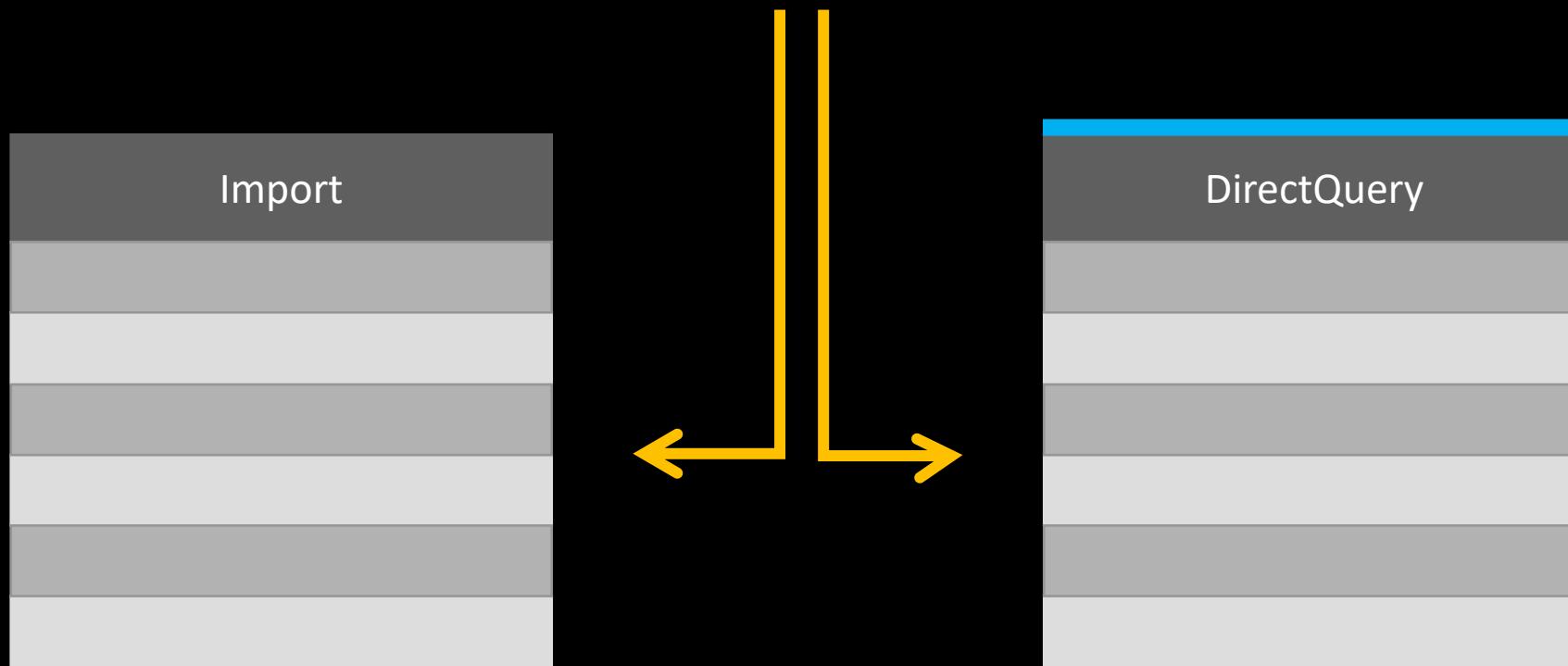
- Azure SQL Database
- Azure Synapse Dedicated SQL pool
- Google BigQuery
- Snowflake

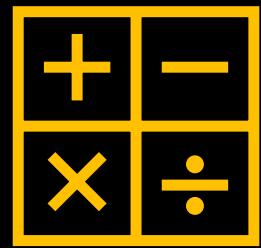


Aggregated data

Detailed data

{ *Query* }





Demo Aggregations



Query Folding



Generate SQL queries



Push queries back to
the source



Improve performance
in Power Query



Query folding – supported sources

- Relational data sources like SQL Server, Oracle...
- OData sources (SharePoint lists...)
- Active Directory
- Exchange
- ...



Query folding – supported operations

- Filtering (both rows and columns)
- Joins
- Aggregation (Group By)
- Pivot and Unpivot
- Numeric Calculations
- Simple transformations (Uppercase / Lowercase)

SQL requests

Refresh Edit columns

Filter by keyword Local time : Last 24 hours Status : All Pool : Built-in Add filter

Showing 1 - 94 of 94 items

Request ID ↑	Request content ↑	Submit time ↑	Duration	Data processed	Submitter ↑
11198235	*** Global stats qu... More	3/1/22, 9:56:40 PM	2 sec	10.00 MB	Marc@Data-marc.com
11198526	SELECT TOP (1000... More	3/1/22, 9:56:40 PM	4 sec	3.29 GB	Marc@Data-marc.com
11196299	SELECT TOP (1000... More	3/1/22, 9:56:09 PM	18 sec	6.64 GB	Marc@Data-marc.com
11033321	SELECT TOP (1000... More	3/1/22, 8:51:57 PM	3 sec	3.29 GB	Marc@Data-marc.com
11031891	*** Global stats qu... More	3/1/22, 8:51:36 PM	2 sec	10.00 MB	Marc@Data-marc.com
11032227	SELECT TOP (1000... More	3/1/22, 8:51:36 PM	5 sec	3.41 GB	Marc@Data-marc.com
11028238	*** Global stats qu... More	3/1/22, 8:51:00 PM	22 sec	1.85 GB	Marc@Data-marc.com
11029080	*** Global stats qu... More	3/1/22, 8:51:00 PM	30 sec	1.62 GB	Marc@Data-marc.com
11030543	*** Global stats qu... More	3/1/22, 8:51:00 PM	30 sec	10.00 MB	Marc@Data-marc.com
11030806	SELECT TOP (1000... More	3/1/22, 8:51:00 PM	36 sec	10.00 MB	Marc@Data-marc.com
11026584	*** Global stats qu... More	3/1/22, 8:50:57 PM	16 sec	351.00 MB	Marc@Data-marc.com
11025192	*** Global stats qu... More	3/1/22, 8:50:47 PM	2 sec	11.00 MB	Marc@Data-marc.com
11025794	SELECT TOP (3502)... More	3/1/22, 8:50:47 PM	3 sec	10.00 MB	Marc@Data-marc.com
11004240	select [].[Customer]... More	3/1/22, 8:44:08 PM	1 sec	10.00 MB	Marc@Data-marc.com
11004313	select [].[Employee]... More	3/1/22, 8:44:08 PM	1 sec	10.00 MB	Marc@Data-marc.com
11003771	select [rows].[Invoice]... More	3/1/22, 8:44:05 PM	15 sec	37.47 GB	Marc@Data-marc.com
10988500	select top 1000 [row]... More	3/1/22, 8:38:51 PM	15 sec	37.41 GB	Marc@Data-marc.com
10987176	select top 1000 [row]... More	3/1/22, 8:38:33 PM	15 sec	37.41 GB	Marc@Data-marc.com
10982220	*** Global stats qu... More	3/1/22, 8:37:42 PM	7 sec	83.00 MB	Marc@Data-marc.com
10982965	*** Global stats qu... More	3/1/22, 8:37:42 PM	15 sec	910.00 MB	Marc@Data-marc.com

Home

Data

Develop

Integrate

Monitor

Manage

Analytics pools

- SQL pools
- Apache Spark pools
- Data Explorer pools (preview)

Activities

- SQL requests
- KQL requests
- Apache Spark applications
- Data flow debug

Integration

- Pipeline runs
- Trigger runs
- Integration runtimes

SQL requests

 Refresh  Edit columns

Filter by keyword

Local time : Last 24 hours

Status : All

Showing 1 - 94 of 94 items

Request ID ↑	Request content ↑	Submit time ↑
11198235	*** Global stats qu... More	3/1/22, 9:56:40 PM
11198526	SELECT TOP (1000... More	3/1/22, 9:56:40 PM
11196299	SELECT TOP (1000... More	3/1/22, 9:56:09 PM
11033321	SELECT TOP (1000... More	3/1/22, 8:51:57 PM
11031891	*** Global stats qu... More	3/1/22, 8:51:36 PM
11032227	SELECT TOP (1000... More	3/1/22, 8:51:36 PM
11028238	*** Global stats qu... More	3/1/22, 8:51:00 PM
11029080	*** Global stats qu... More	3/1/22, 8:51:00 PM
11030543	*** Global stats qu... More	3/1/22, 8:51:00 PM
11030806	SELECT TOP (1000... More	3/1/22, 8:51:00 PM
11026584	*** Global stats qu... More	3/1/22, 8:50:57 PM
11025192	*** Global stats qu... More	3/1/22, 8:50:47 PM
11025794	SELECT TOP (3502)... More	3/1/22, 8:50:47 PM
11004240	select [__].[CustomerKey] as [Customer Key], [__].[CityKey] as [City Key], [__].[CustomerName] as [Customer Name], [__].[Address] as [Address], [__].[PostalCode] as [Postal Code], [__].[Phone] as [Phone], [__].[Fax] as [Fax], [__].[Email] as [Email], [__].[SalesPersonKey] as [Salesperson Key], [__].[SalesTerritoryKey] as [Sales Territory], [__].[SalesTerritoryName] as [Sales Territory Name], [__].[SalesQuota] as [Sales Quota], [__].[CommissionRate] as [Commission Rate], [__].[LastModifiedBy] as [Last Modified By], [__].[LastModifiedDate] as [Last Modified Date], [__].[Rowguid] as [Rowguid], [__].[ModifiedDate] as [Modified Date]	3/1/22, 8:44:08 PM
11004313	select [__].[EmployeeID] as [Employee ID], [__].[EmployeeName] as [Employee Name], [__].[Title] as [Title], [__].[TitleOfCourtesy] as [Title Of Courtesy], [__].[BirthDate] as [Birth Date], [__].[HireDate] as [Hire Date], [__].[Address] as [Address], [__].[PostalCode] as [Postal Code], [__].[Phone] as [Phone], [__].[Fax] as [Fax], [__].[Email] as [Email], [__].[ReportTo] as [Report To], [__].[CommissionRate] as [Commission Rate], [__].[LastModifiedBy] as [Last Modified By], [__].[LastModifiedDate] as [Last Modified Date], [__].[Rowguid] as [Rowguid], [__].[ModifiedDate] as [Modified Date]	3/1/22, 8:44:08 PM
11003771	select [__].[rows].[InvoiceDateKey] as [Invoice Date Key], [__].[rows].[InvoiceDate] as [Invoice Date], [__].[rows].[InvoiceLineCount] as [Invoice Line Count], [__].[rows].[InvoiceTotalAmount] as [Invoice Total Amount], [__].[rows].[InvoiceTotalExcludingTax] as [Invoice Total Excluding Tax], [__].[rows].[InvoiceTotalIncludingTax] as [Invoice Total Including Tax], [__].[rows].[InvoiceType] as [Invoice Type], [__].[rows].[LastModifiedBy] as [Last Modified By], [__].[rows].[LastModifiedDate] as [Last Modified Date], [__].[rows].[Rowguid] as [Rowguid], [__].[rows].[ModifiedDate] as [Modified Date]	3/1/22, 8:44:05 PM
10988500	select top 1000 [row].[InvoiceDateKey] as [Invoice Date Key], [row].[InvoiceDate] as [Invoice Date], [row].[InvoiceLineCount] as [Invoice Line Count], [row].[InvoiceTotalAmount] as [Invoice Total Amount], [row].[InvoiceTotalExcludingTax] as [Invoice Total Excluding Tax], [row].[InvoiceTotalIncludingTax] as [Invoice Total Including Tax], [row].[InvoiceType] as [Invoice Type], [row].[LastModifiedBy] as [Last Modified By], [row].[LastModifiedDate] as [Last Modified Date], [row].[Rowguid] as [Rowguid], [row].[ModifiedDate] as [Modified Date]	3/1/22, 8:38:51 PM
10987176	select top 1000 [row].[InvoiceDateKey] as [Invoice Date Key], [row].[InvoiceDate] as [Invoice Date], [row].[InvoiceLineCount] as [Invoice Line Count], [row].[InvoiceTotalAmount] as [Invoice Total Amount], [row].[InvoiceTotalExcludingTax] as [Invoice Total Excluding Tax], [row].[InvoiceTotalIncludingTax] as [Invoice Total Including Tax], [row].[InvoiceType] as [Invoice Type], [row].[LastModifiedBy] as [Last Modified By], [row].[LastModifiedDate] as [Last Modified Date], [row].[Rowguid] as [Rowguid], [row].[ModifiedDate] as [Modified Date]	3/1/22, 8:38:33 PM
10982220	*** Global stats qu... More	3/1/22, 8:37:42 PM
10982965	*** Global stats qu... More	3/1/22, 8:37:42 PM

Request content

11198526

```
SELECT
TOP (1000001) *
FROM
(
SELECT [t5].[Sales Territory],SUM([t4].[Profit])
AS [a0]
FROM
(
select [__].[SaleKey] as [Sale Key],
[__].[CityKey] as [City Key],
[__].[CustomerKey] as [Customer Key],
[__].[BillToCustomerKey] as [Bill To Customer Key],
[__].[StockItemKey] as [Stock Item Key],
[__].[InvoiceDateKey] as [Invoice Date Key],
[__].[DeliveryDateKey] as [Delivery Date Key],
[__].[SalespersonKey] as [Salesperson Key],
[__].[WWIInvoiceID] as [WWIInvoice ID],
[__].[Description] as [Description],
[__].[Package] as [Package],
[__].[Quantity] as [Quantity],
[__].[UnitPrice] as [Unit Price],
[__].[TaxRate] as [Tax Rate],
[__].[TotalExcludingTax] as [Total Excluding Tax],
[__].[TaxAmount] as [Tax Amount],
[__].[Profit] as [Profit],
[__].[TotalIncludingTax] as [Total Including Tax],
[__].[TotalDryItems] as [Total Dry Items],
[__].[TotalChillerItems] as [Total Chiller Items],
[__].[LineageKey] as [Lineage Key]
from [sales_model].[Sales] as [__]
where [__].[InvoiceDateKey] >= convert(date, '2013-01-01') and [__].[InvoiceDateKey]
) AS [t4]
INNER JOIN

(
select [__].[CityKey] as [City Key],
[__].[WWICityID] as [WWICity ID],
[__].[City] as [City],
```

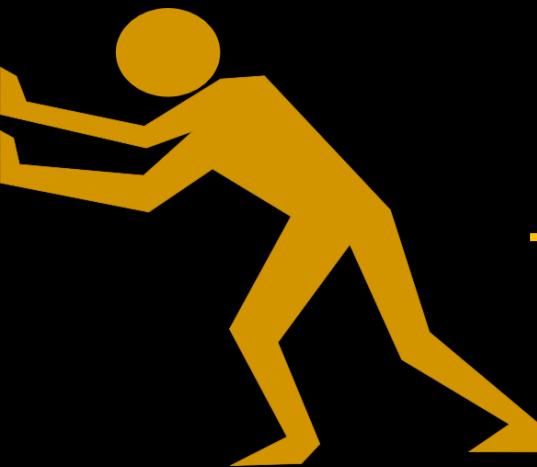
DAX Studio trace

The screenshot shows the DaxStudio interface version 2.17.3. The top ribbon has tabs for File, Home, Advanced, Help, Layout, and Traces. The Traces tab is selected. The toolbar includes Run, Cancel, Clear Cache, Output, Query Builder, View, Cut, Undo, Copy, Redo, Format Query, To Upper, To Lower, Comment, Uncomment, Merge XML, Find, Replace, Load Perf Data, All Queries, Query Plan, Server Timings, Connect, Refresh Metadata, Power BI, and Traces.

The main window has a left sidebar with a tree view of tables: City, Customer, Date, Employee, Sales, Sales Agg, Sales Hybrid, and StockItem. Below the tree is a search bar. The main area displays XML code for a DAX query:

```
<Batch Transaction="true" xmlns="http://schemas.microsoft.com/analysisservices/2003/engine">
<Refresh xmlns="http://schemas.microsoft.com/analysisservices/2014/engine">
<DatabaseID>3771129f-bce3-4f2e-b0e4-91af44829145</DatabaseID>
<MaxParallelism>2</MaxParallelism>
<Partitions>
<xss:schema xmlns:xss="http://www.w3.org/2001/XMLSchema" xmlns:sql="urn:schemas-microsoft-com:xml-sql">
<xss:element>
<xss:complexType>
<xss:sequence>
<xss:element type="row"/>
</xss:sequence>
</xss:complexType>
</xss:element>
<xss:complexType name="row">
<xss:sequence>
<xss:element name="ID" type="xs:unsignedLong" sql:field="ID" minOccurs="0"/>
<xss:element name="ID_Table" type="xs:string" sql:field="ID_Table" minOccurs="0"/>
<xss:element name="ID_Partition" type="xs:string" sql:field="ID_Partition" minOccurs="0"/>
<xss:element name="RefreshType" type="xs:long" sql:field="RefreshType" minOccurs="0"/>
</xss:sequence>
</xss:complexType>
</xss:schema>
<row xmlns="urn:schemas-microsoft-com:xml-analysis:rowset">
<ID>123</ID>
<RefreshTimes>1</RefreshTimes>
</row>
</xss:rowset>
</xss:partitions>
</xss:batch>
</xss:refresh>
</batch>
```

The bottom section shows a trace table with columns: StartTime, Type, Duration, User, Database, and Query. The table lists numerous entries from different users (Power BI Service, DemoUser@Data-marc.com) performing various operations like XMLA requests, DAX queries, and XML refreshes. The trace table has a yellow header bar stating "Trace is not currently active, click on the All Queries button in the ribbon to resume tracing".



Demo Query Folding



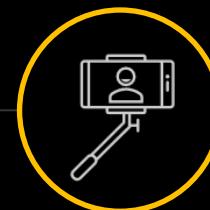


Power BI Licensing

Power BI Free



Personal use only



Self-Service

Power BI Pro



Sharing &
Collaborating



Organization
perspective

Shared resources in the cloud



Power BI Licensing

Power BI Premium

Now also Premium Per User



Flexibility to license
by capacity



Greater scale
and performance



Unifying self-service and
enterprise BI



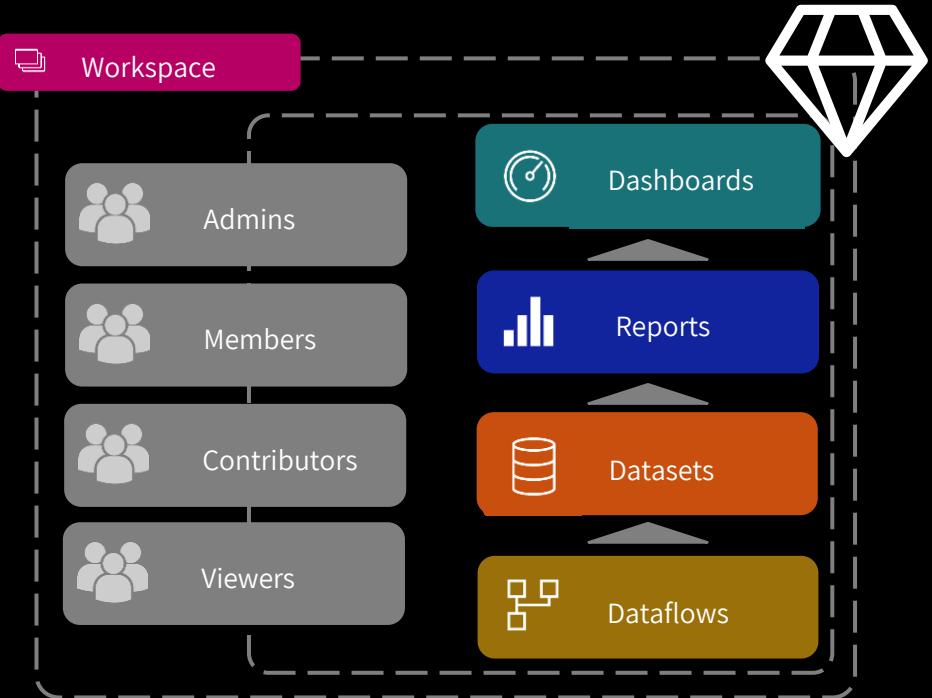
Extending on-premises
capabilities

Dedicated resources in the cloud



Power BI Premium

- Dedicated resources in the cloud (vs. shared resources on Pro)
- Bunch of awesome extra (enterprise) features
- Large scale distribution to viewers who don't need a per-user license
... much more!





Power BI Premium

Power BI Pro	Power BI Premium		
Per user	Per user (preview)	Per capacity	Auto-scale (preview)
€ 8,40 Per user / month	€ 16,90 Per user / month	€ 4.212,30 Per capacity / month	€ 71,68 Per vCore / 24 hours
License individual users with modern, self-service analytics to visualize data with live dashboards and reports, and share insights across your organization.	License individual users to accelerate access to insights with advanced AI, unlock self-service prep for big data, and simplify data management and access at enterprise scale.	License your organization with capacity to accelerate access to insights with advanced AI, unlock self-service prep for big data, and simplify data management and access at enterprise scale—without per-user licenses for content consumers.	Respond to occasional usage spikes in Power BI Premium capacity by automatically adding one vCore at a time per 24-hour period as your organization's needs change with the flexibility of an Azure subscription.

- Also included in Microsoft 365 E5

- Includes all features available in Power BI Pro
- Brings additional Premium functionality

- Users still require Power BI Pro license for publishing content to the Premium capacity
- Enable Auto-scale with Azure subscription for automated scaling of capacity size

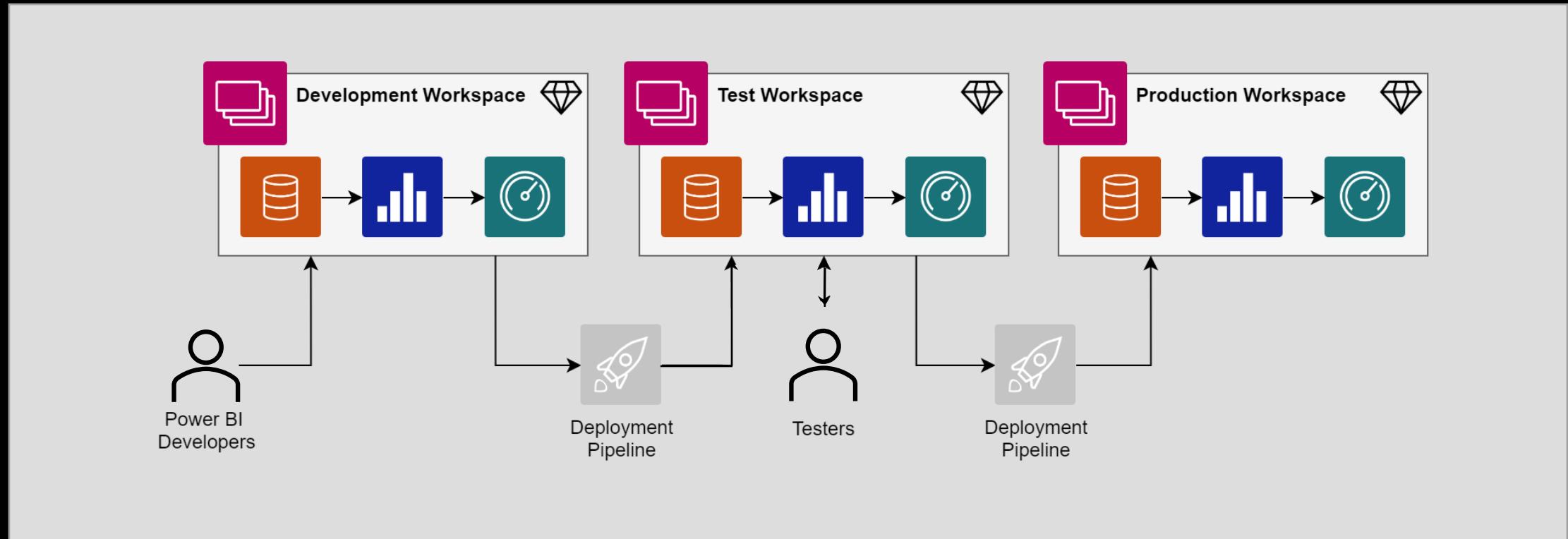


Premium specific features

- **Paginated Reports**
- **Share with free users**
(P sku only! Not in A sku!)
- **Auto-scale**
- **Support for larger dataset sizes**
- **48x daily refresh**
(and Automatic Page Refresh)
- **Extra dataflows features**
- **Enhanced embedding scenarios**
- **Bring your own key (BYOK)**
- **Hybrid tables**
- **Multi-geo support**
- **XMLA endpoints**
- **Paginated Reports**
- **Deployment Pipelines** (extra slide on this)
- **AI workloads**

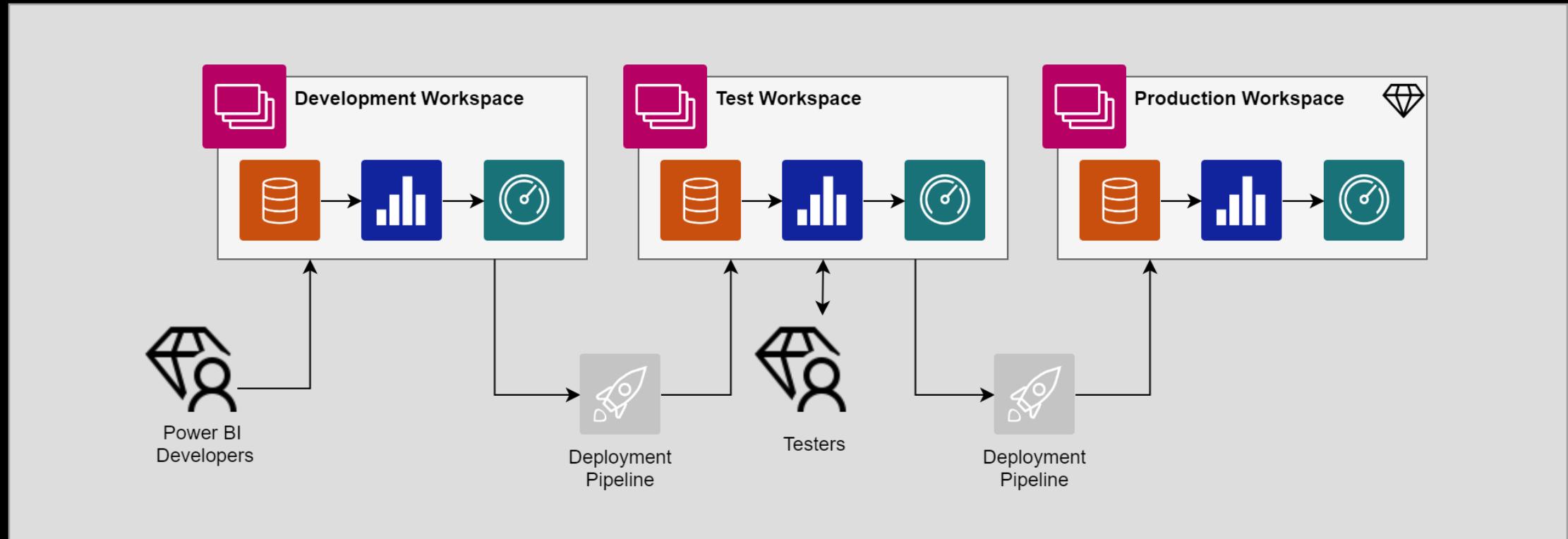


Lower capacity utilization with DTAP





Lower capacity utilization with DTAP



Hybrid tables & incremental refresh

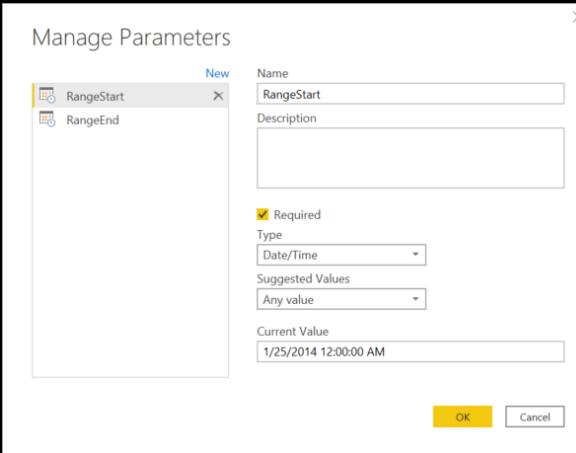
Incremental refresh

- Fewer refresh cycles for fast-changing data – DirectQuery mode gets the latest data updates as queries are processed without requiring a high refresh cadence.
- Refreshes are faster - Only the most recent data that has changed needs to be refreshed.
- Refreshes are more reliable - Long-running connections to volatile data sources aren't necessary. Queries to source data run faster, reducing potential for network problems to interfere.
- Resource consumption is reduced - Less data to refresh reduces overall consumption of memory and other resources in both Power BI and data source systems.
- Enables large datasets - Datasets with potentially billions of rows can grow without the need to fully refresh the entire dataset with each refresh operation.
- Easy setup - Incremental refresh policies are defined in Power BI Desktop with just a few tasks. When published, the service automatically applies those policies with each refresh.

Incremental refresh

- Incremental refresh is supported for Power BI Premium, Premium per user, **Power BI Pro**, and Power BI Embedded datasets.
- Getting the latest data in **real time** with DirectQuery is **only supported for Power BI Premium**, Premium per user, and Power BI Embedded datasets.

Incremental refresh config



The screenshot shows the 'CustomerPONumber' table in Power BI Desktop. A context menu is open over the 'OrderDate' column, with the 'Custom Filter...' option highlighted by a red box. Other options in the menu include Sort Ascending, Sort Descending, Clear Sort, Clear Filter, Remove Empty, and Date/Time Filters.

Incremental refresh and real-time data

Refresh large tables faster with incremental refresh. Plus, get the latest data in real time with DirectQuery (Premium only). [Learn more](#)

These settings will apply when you publish the dataset to the Power BI service. Once you do that, you won't be able to download it back to Power BI Desktop. [Learn more](#)

- Select table**
FactInternetSales
- Set import and refresh ranges**
 Incrementally refresh this table
Archive data starting Years before refresh date
Data imported from 12/21/2016 to 12/18/2021.
Incrementally refresh data starting Days before refresh date
Data will be incrementally refreshed from 12/18/2021 to 12/21/2021.
- Choose optional settings**
 Get the latest data in real time with DirectQuery (Premium only) [Learn more](#)
 Only refresh complete days [Learn more](#)
 Detect data changes [Learn more](#)
- Review and apply**
Archived
5 years before refresh date 3 days before refresh date Refresh date



Hybrid tables

- Live / Realtime data in Power BI
- Combines different storage modes on partition level in a single table
- Goes hand-in-hand with Incremental Refresh

Granularity	Name	Row Count
Year	2011	295,489,717
Year	2012	297,678,498
Year	2013	295,575,442
Year	2014	292,477,875
Year	2015	297,780,469
Year	2016	294,060,081
Year	2017	300,419,682
Year	2018	296,541,108
Year	2019	292,787,420
Year	2020	299,273,979
Quarter	2021Q1	74,135,277
Month	2021Q104	24,939,498
Day	2021Q10501	820,805
Day	2021Q10502	826,885
Day	2021Q10503	821,043
Day-DirectQuery	2021Q10504-DQ	271,110
Total		3,063,898,887

Archived: Import

Incremental refresh: Import

Real time: DirectQuery



Hybrid tables

- Implementation with Incremental Refresh
- Customizable via 3rd party tooling like Tabular Editor

>> Limitation: Only 1 DQ partition per table allowed at the moment.

Incremental refresh and real-time data

These settings will apply when you publish the dataset to the Power BI service. Once you do that, you won't be able to download it back to Power BI Desktop. [Learn more](#)

1. Select table
Sales Agg

2. Set import and refresh ranges
 Incrementally refresh this table
Archive data starting Years before refresh date
Data imported from 3/2/2012 to 2/20/2022.
Incrementally refresh data starting Days before refresh date
Data will be incrementally refreshed from 2/20/2022 to 3/2/2022.

3. Choose optional settings
 Get the latest data in real time with DirectQuery (Premium only) [Learn more](#)
 Only refresh complete days [Learn more](#)
 Detect data changes [Learn more](#)

4. Review and apply

Archived Incremental Refresh Real time

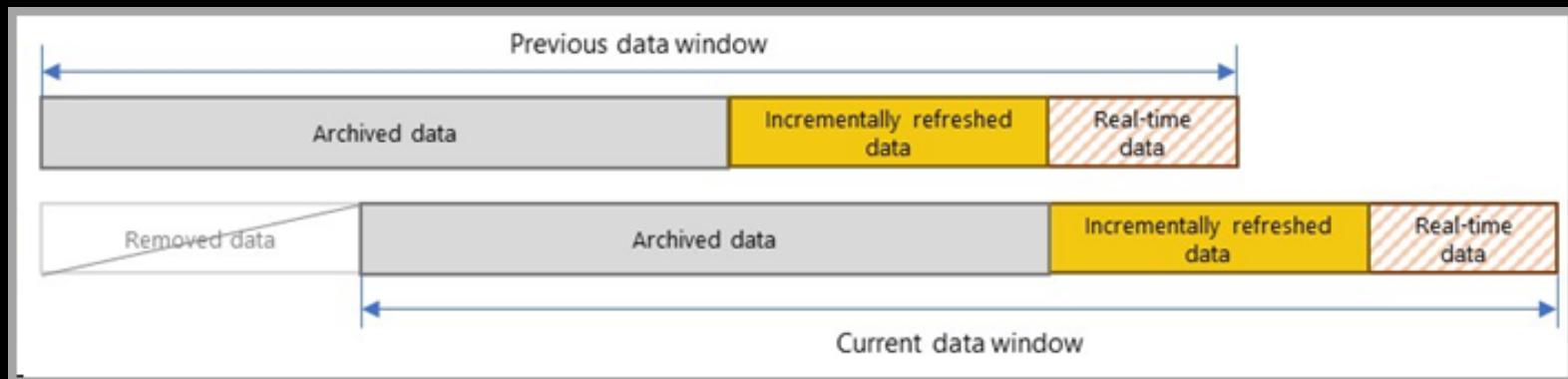
10 years before refresh date 10 days before refresh date Refresh date

[Apply](#) [Cancel](#)



Hybrid tables – what challenge does it solve?

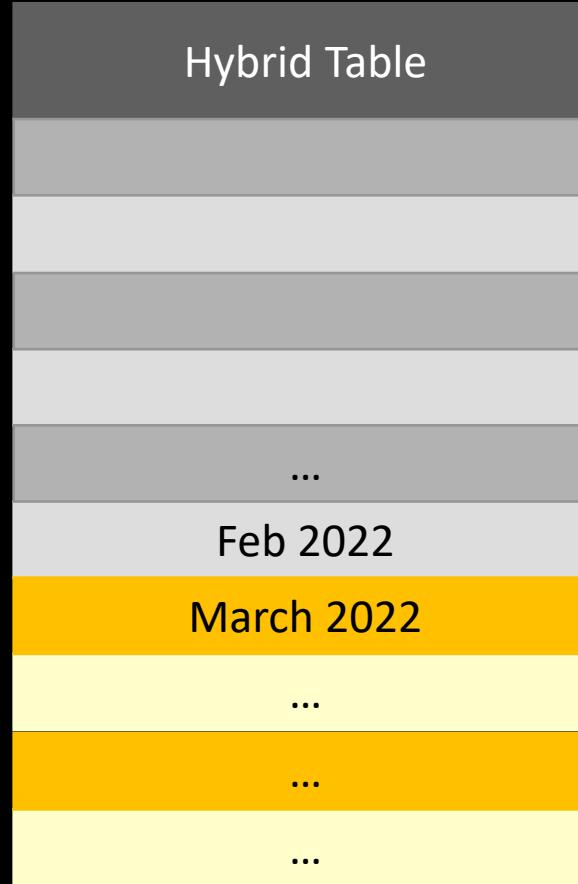
- SQL Realtime scenarios without full tables on DQ mode
- SQL No complex refresh mechanisms needed with partition refresh and queries over XMLA
- SQL No more multiple tables and complex DAX to combine to achieve the same goal





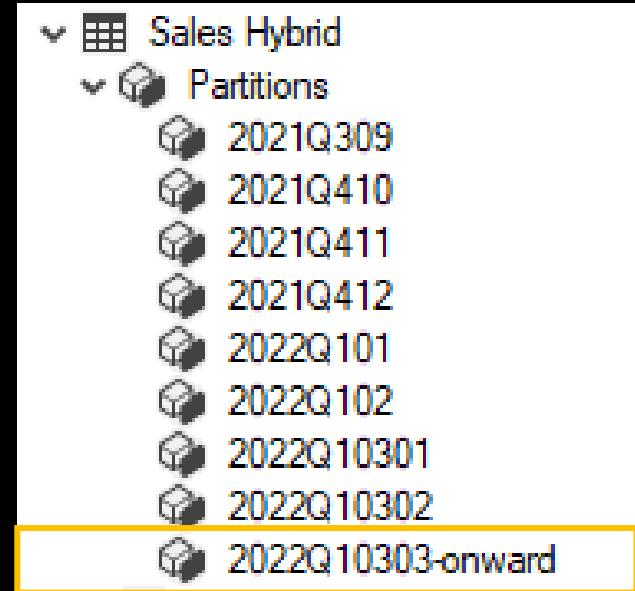
Hybrid tables – Keep in mind that...

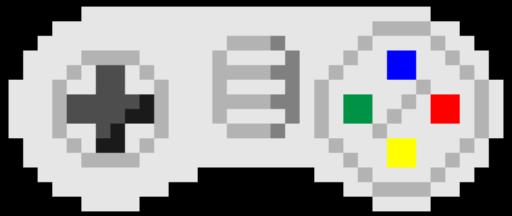
- Premium feature
- DAX restrictions for DirectQuery apply
- Limited Power Query capabilities (due to DQ)
- Requires Large Dataset Format (storage) in workspace
- Performance hit on upstream data sources



} Import

} DirectQuery →





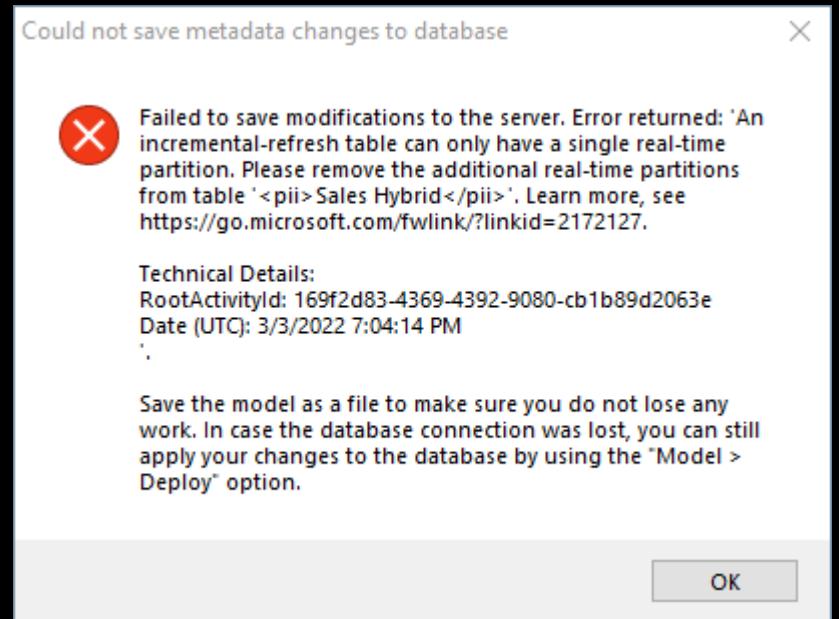
Demo Hybrid Tables
Latest data in real time



Can I change partition storage modes?

You cannot adjust tables with incremental refresh policies applied. However, there are other options to consider.

Options	Default
Data View	DirectQuery
Mode	PolicyRange
Source Type	





Manually setup partitioning

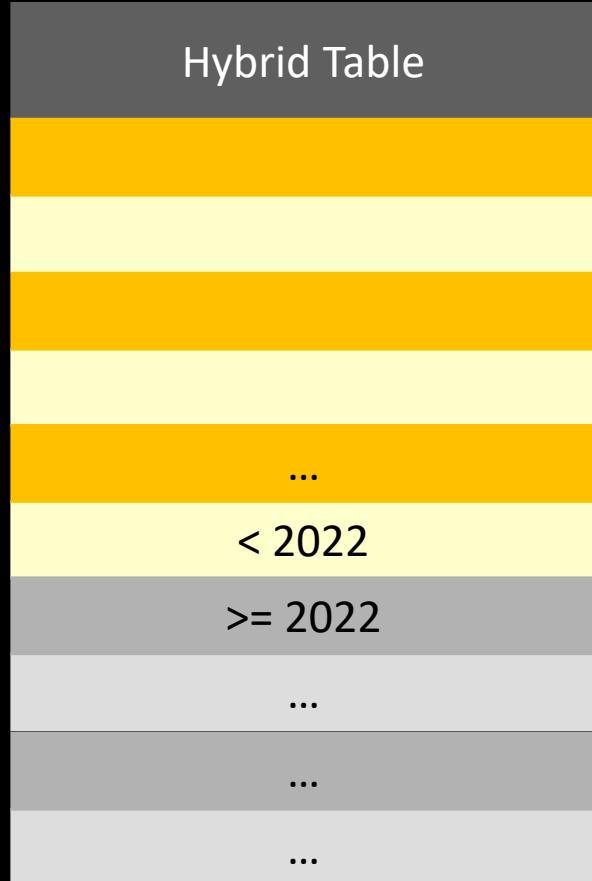
The screenshot shows the Tabular Editor interface with the following details:

- Left pane (Model View):** Shows the project structure with nodes like Model, Data Sources, Perspectives, Relationships, Roles, Shared Expressions, Tables, and Partitions.
- Middle pane (Expression Editor):** Displays DAX code for a partition:

```
let
    Source = Sql.Databases((SynapseEndpoint)),
    #"WideWorldImportersDW-Standard" = Source[[Name=(DatabaseName)]][Data],
    Fact_Sale = #"WideWorldImportersDW-Standard"[[Schema=(SynapseSchema), Item="Sales"]][Data],
    #"Filtered Rows" = Table.SelectRows(#"Fact_Sale", each [InvoiceDateKey] < #date(2022, 01, 01) ),
    SplitByCharacter = Table.TransformColumnNames(#"Filtered Rows", each
        Text.Combine(
            Splitter.SplitTextByCharacterTransition({{"a".."z"}, {"A".."Z"}})(_)
            , " ")
        )
    in
        SplitByCharacter
```

A red box highlights the part of the code where rows are filtered based on the date key.
- Bottom pane (Properties):** Shows the properties for the selected object (Sales Hybrid-DQ). A red box highlights the "Mode" field, which is set to "DirectQuery".

Create your own hybrid table setup, using 3rd party tooling like Tabular Editor



} DirectQuery

} Import

DirectQuery Partition definition

```
let
    Source = Sql.Databases((SynapseEndpoint)),
    #"WideWorldImportersDW-Standard" = Source{[Name=(DatabaseName)]}[Data],
    Fact_Sale = #"WideWorldImportersDW-
Standard"[{[Schema=(SynapseSchema),Item="Sales"]}] [Data],
    #"Filtered Rows" = Table.SelectRows(#"Fact_Sale", each
        [InvoiceDateKey] < #date(2022, 01, 01) and
        [InvoiceDateKey] >= #date(2013, 01, 01)),
        SplitByCharacter = Table.TransformColumnNames(#"Filtered Rows", each
            Text.Combine(
                Splitter.SplitTextByCharacterTransition({"a".."z"}, {"A".."Z"})(_)
                , " ")
            )
        in
            SplitByCharacter
```

Import Partition definition

```
let
    Source = Sql.Databases((SynapseEndpoint)),
    #"WideWorldImportersDW-Standard" = Source{[Name=(DatabaseName)]}[Data],
    Fact_Sale = #"WideWorldImportersDW-
Standard"[{[Schema=(SynapseSchema),Item="Sales"]}] [Data],
    #"Filtered Rows" = Table.SelectRows(#"Fact_Sale", each
        [InvoiceDateKey] >= #date(2022, 01, 01) and
        [InvoiceDateKey] < #date(2022, 03, 31)),
        SplitByCharacter = Table.TransformColumnNames(#"Filtered Rows", each
            Text.Combine(
                Splitter.SplitTextByCharacterTransition({"a".."z"}, {"A".."Z"})(_)
                , " ")
            )
        in
            SplitByCharacter
```



Demo Hybrid Tables
Historic data via DirectQuery

Refresh challenges





Refresh options

- Scheduled in the service
- Manual trigger
- Power Automate
- PowerShell
- API



Effective refreshing

Considerations

- SQL Refreshing the entire model takes too long with high load on sources
- SQL Can we only refresh certain tables?
- SQL Can we only refresh certain partitions?
- SQL Can we use DQ tables/partitions (Hybrid Tables)

What do we need?

- SQL Async refresh API
- SQL XMLA Endpoints



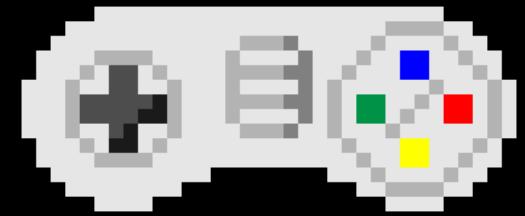
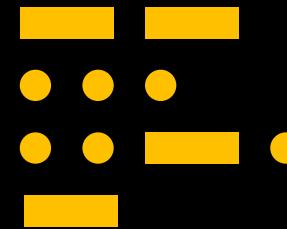
Async refresh API (preview)

- Specify the objects to refresh

POST

<https://api.powerbi.com/v1.0/myorg/groups/f089354e-8366-4e18-aea3-4cb4a3a50b48/datasets/cfafbeb1-8037-4d0c-896e-a46fb27ff229/refreshes>

```
{  
  "type": "Full",  
  "commitMode": "transactional",  
  "maxParallelism": 2,  
  "retryCount": 2,  
  "objects": [  
    {  
      "table": "DimCustomer",  
      "partition": "DimCustomer"  
    },  
    {  
      "table": "DimDate"  
    }  
  ]  
}
```

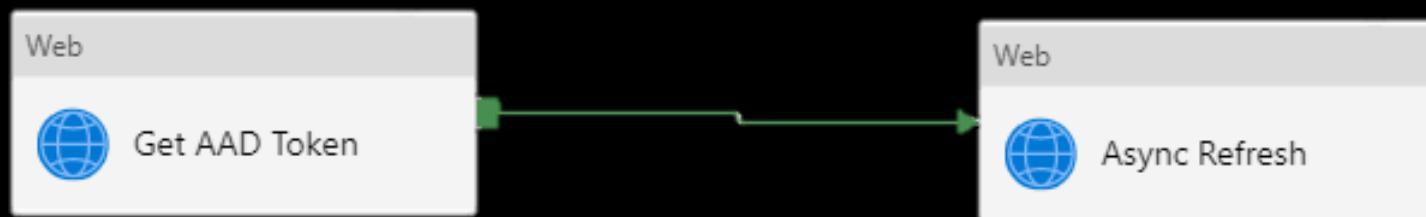


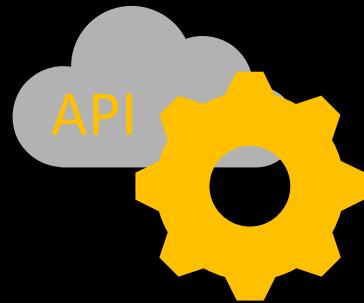
— Demo Asynchronous refresh API



Central E2E orchestration

- Combine pipelines from Data Platform with Power BI
- Lowest latency between source and report
- Consider including backup operations for Power BI
- Incremental loading where possible





Demo end-2-end orchestration

Scaling





Scaling data platform

Spark Cluster:

- Use multiple cluster configs
- Autoscale -> *It can take 1 to 5 minutes for a scaling operation to complete*
- Dynamic allocation of executors
- Automatic pause

Size	vCore	Memory
Small	4	32 GB
Medium	8	64 GB
Large	16	128 GB
XLarge	32	256 GB
XXLarge	64	512 GB
XXX Large (Isolated Compute)	80	504 GB



Scaling data platform

If you are unsure what size cluster to utilize, start with a Medium Spark pool with three nodes and the ability to scale to ten nodes. After executing your spark jobs, view the Apache Spark history server to check performance.

- There are a few areas to monitor:
 - **Disk Spillage** – If there are jobs where data spilled to disk, consider increasing the size of your Spark instance.
 - **Number of Tasks vs. Number of Executors** – Utilizing larger clusters can cause high Garbage Collection (GC). To prevent this, it is best to have less than 5 tasks executing per core when running Spark jobs. If you have exceeded this, consider scaling up your Spark instance.
 - **Executor Usage** – In the Diagnostics tab, the Executor Usage Graph will show you the executor usage over time. If executor usage is very low, consider scaling down your Spark instance.



Auto-scale (gen2 only)

Auto-scale adds:

- Additional vCores
- Applies for at least 24h

Configured through

- Max. number of scalable vCores
- Azure subscription – Pay as you go

Power BI Premium > AutoScale test

Premium Generation 2 (preview)
Improve performance and easily track your usage with Premium Generation 2. Enable the preview today. Note: Once enabled, the capacity stops emitting metrics to the metrics app. [Learn more](#)

Enabled

Size | P1
8
Base v-cores

Auto scale | On
0
Additional v-cores in use
Max = 2

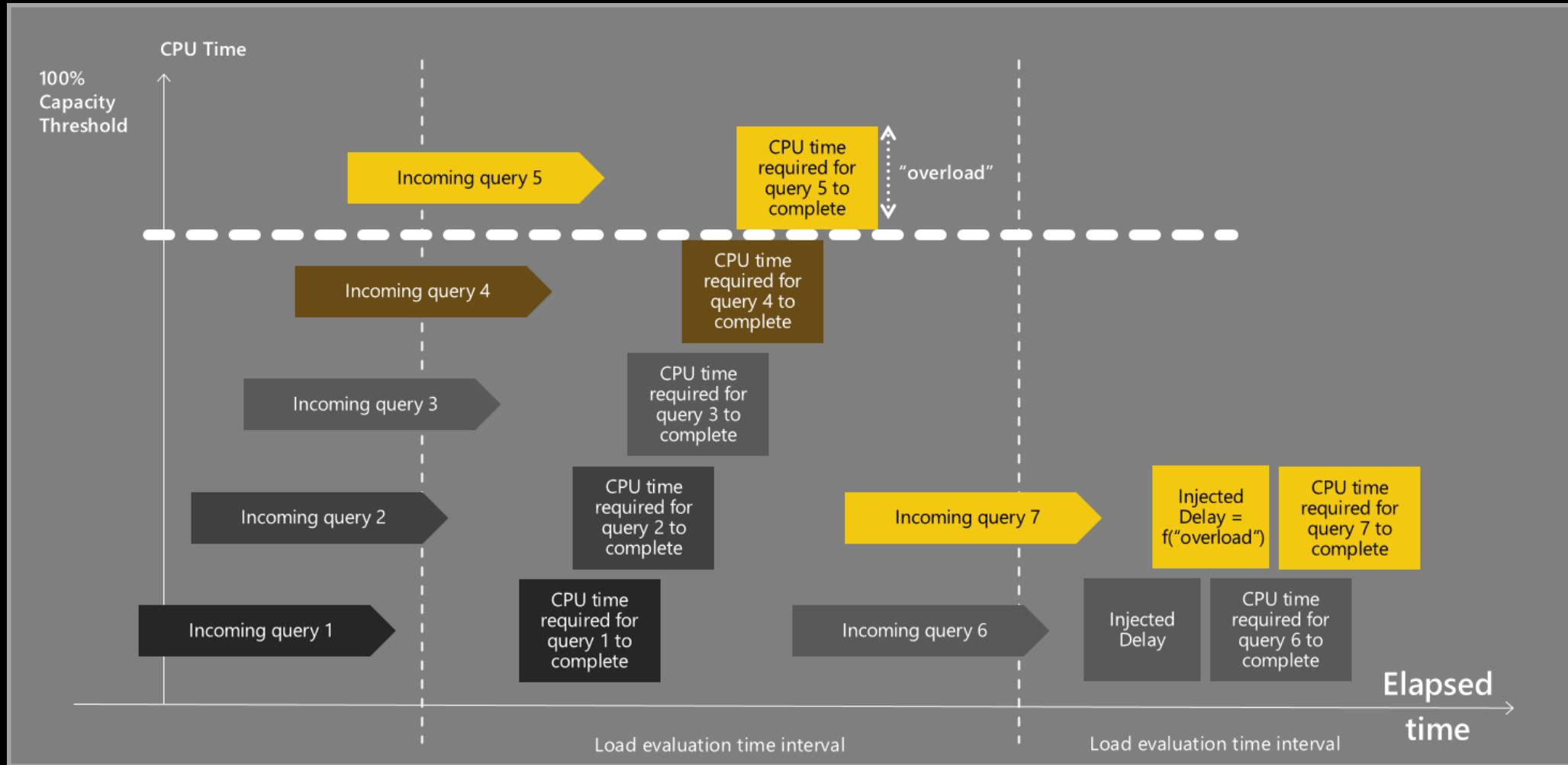
The Premium SKU size you purchased is a P1, which gives you access to 8 v-cores.

[Change size](#) [Manage auto-scale](#)

Auto-scale gives the flexibility to use as much capacity as you need, when you need it. [Learn more](#)

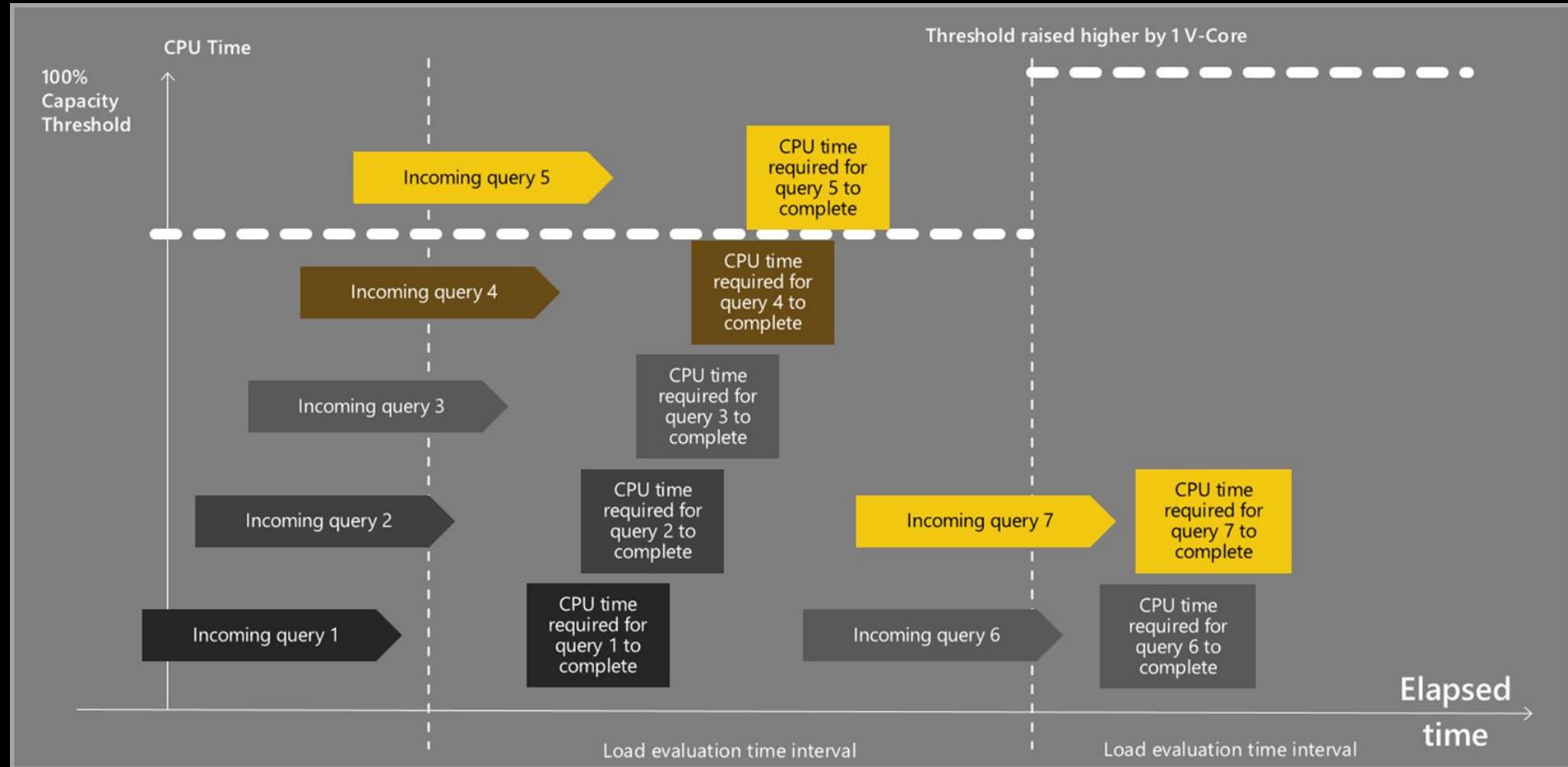


Power BI Premium – Auto scale





Power BI Premium – Auto scale



Monitoring - revisited



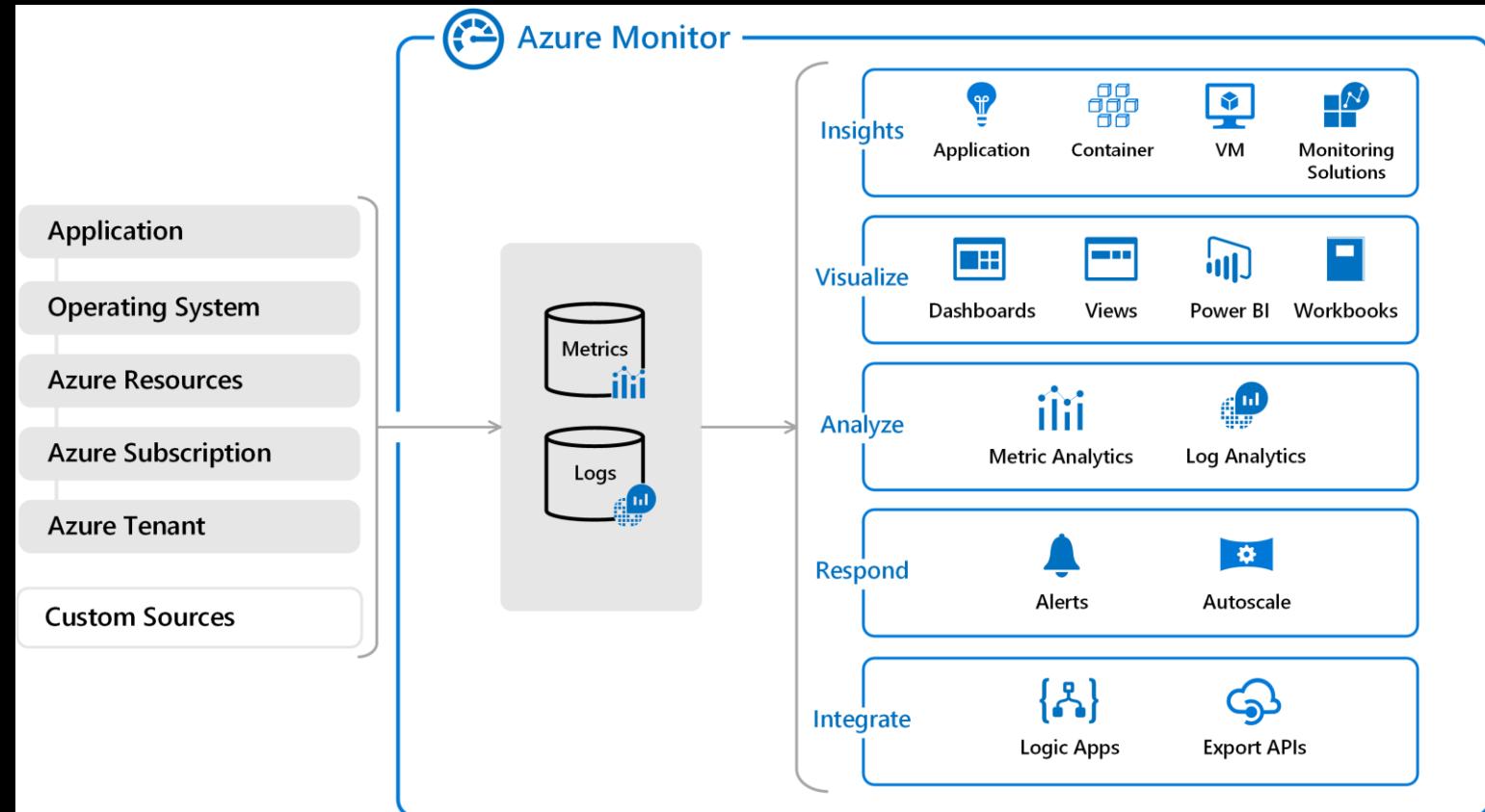


Data platform monitoring



Log Analytics Workspace / Azure Monitor

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.





Scenarios

- Pipelines:
 - Identify (unusual) long durations
 - Identify unusual amount of runs
 - Trend analysis for durations
- Spark cluster:
 - Analyse utilization of driver and worker nodes
- SQL Serverless:
 - Data scanned by DirectQuery & import to datasets
- Storage:
 - Analyse trend in amount of data stored
 - Analyse trend in amount/duration of reads and writes
- Cost:
 - Analyse Resource Costs

Solution cost analysis

Our Azure cost in a period of January 1st till March 6th.

Obviously, this is **not representative** till everyone's case, depending on scaling etc.

Service Name	Service Resource	Cost
Azure Synapse Analytics	vCore	\$191.15
Power BI Embedded	A1	\$110.13
Azure Synapse Analytics	Data Stored	\$28.97
Azure Synapse Analytics	Data Processed	\$20.86
Azure Synapse Analytics	Azure Hosted IR Data Movement	\$20.57
Azure Synapse Analytics	100 DWUs	\$7.55
Load Balancer	Included LB Rules and Outbound Rules	\$2.18
Load Balancer	Overage LB Rules and Outbound Rules	\$1.7
Storage	Hot Iterative Read Operations	\$1.36
Storage	Hot Read Operations	\$1.08
	Total	\$385.55

Power BI monitoring





Take care of your capacity

Permissions



Two levels of assignment permissions

Management



Work with a consistent management experience

Workspaces



Easily assign workspaces

Capacity



Move content from shared to dedicated capacity



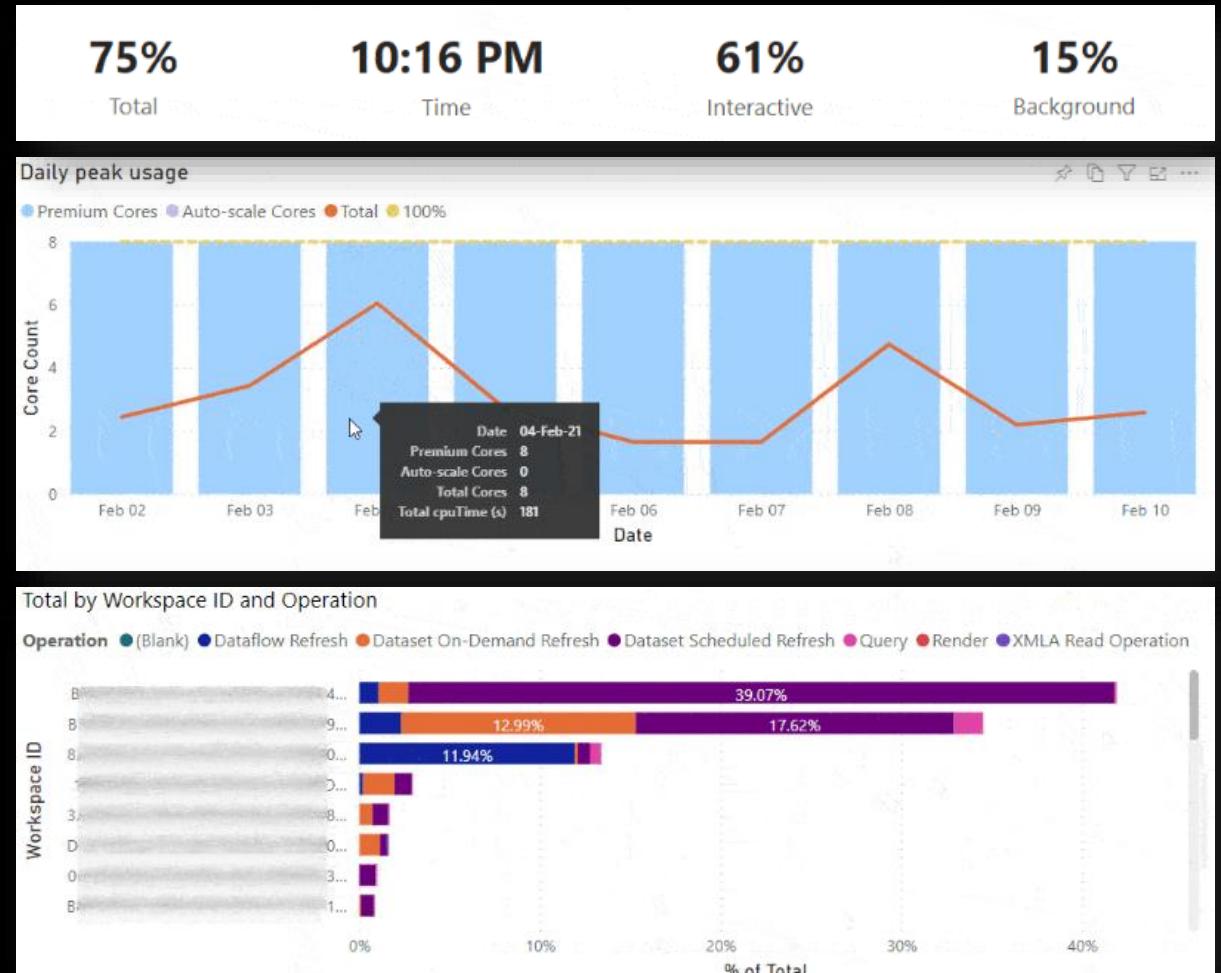
Capacity metrics app (only gen2)

Available vCores

- SQL Available vCores at a given time

Utilization

- SQL Per workspace
- SQL Per refresh type for datasets & dataflows
(on demand / scheduled)
- SQL Per operation type
(XMLA / query / rendering)





Spikes in capacity usage

Influencing factors

- Amount of datasets
- Dataset composition
- Model schema
- Peak performance
- Row-level security
- Number of visuals on screen
- Concurrent viewers
- Usage of dataflows



Spikes in capacity usage

Take aways

- Focus on capacity CPU & overload minutes
- One-time peaks – **Don't worry too much!**
- Continuous overload – **Investigate what's causing it!**



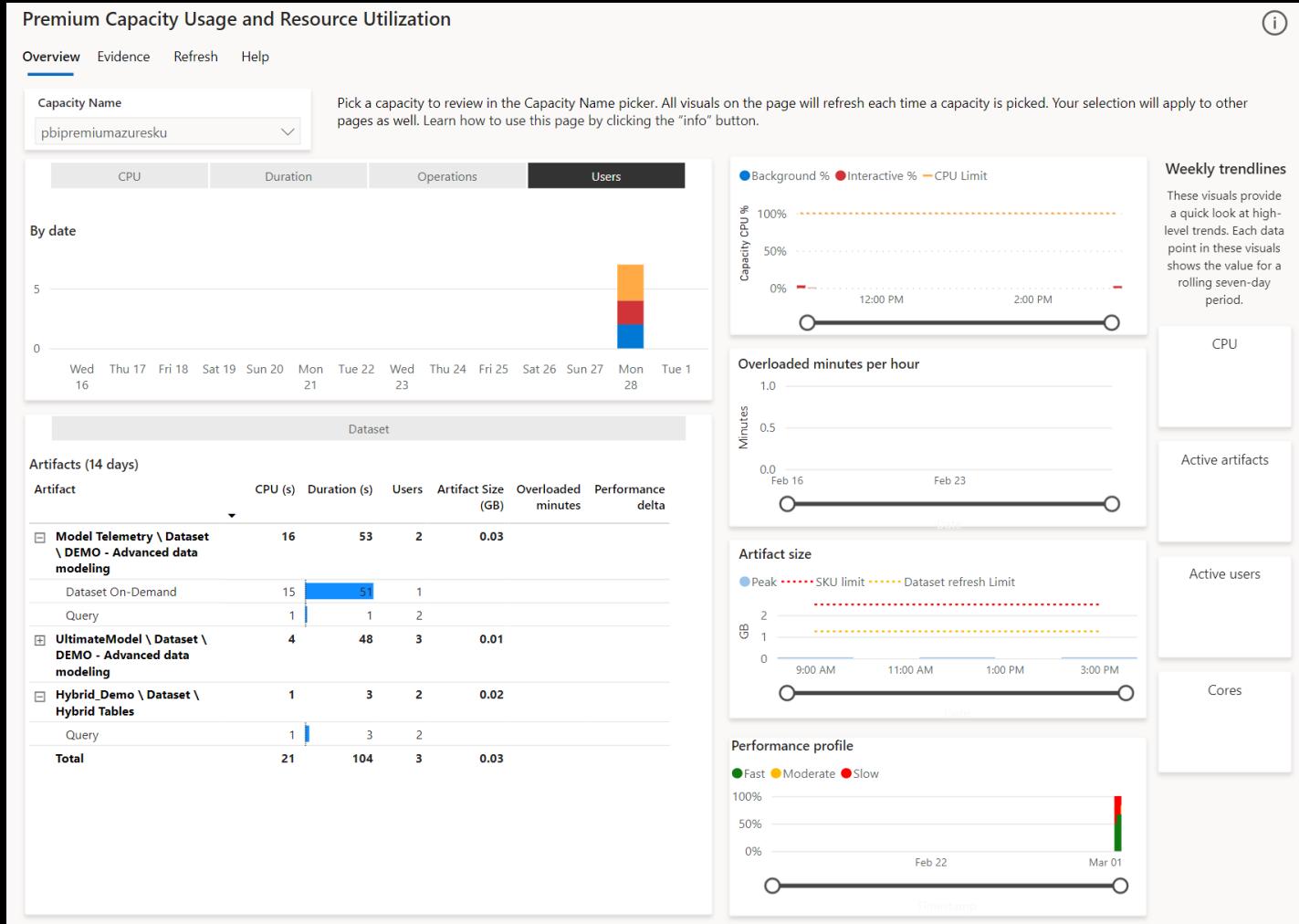
Spikes in capacity usage

Considerations

- **Optimization:** Limit columns on a dataset, avoid heavy operations
- **Scale-out:** isolate large contributors to a separate capacity, instead of upgrading.
- **Scale-up:**
Consider Auto-scale if peaks happen on specific days (like month end) or upgrade capacity (last resort)



Template report to get insights



- Overview
- Evidence
- Refresh

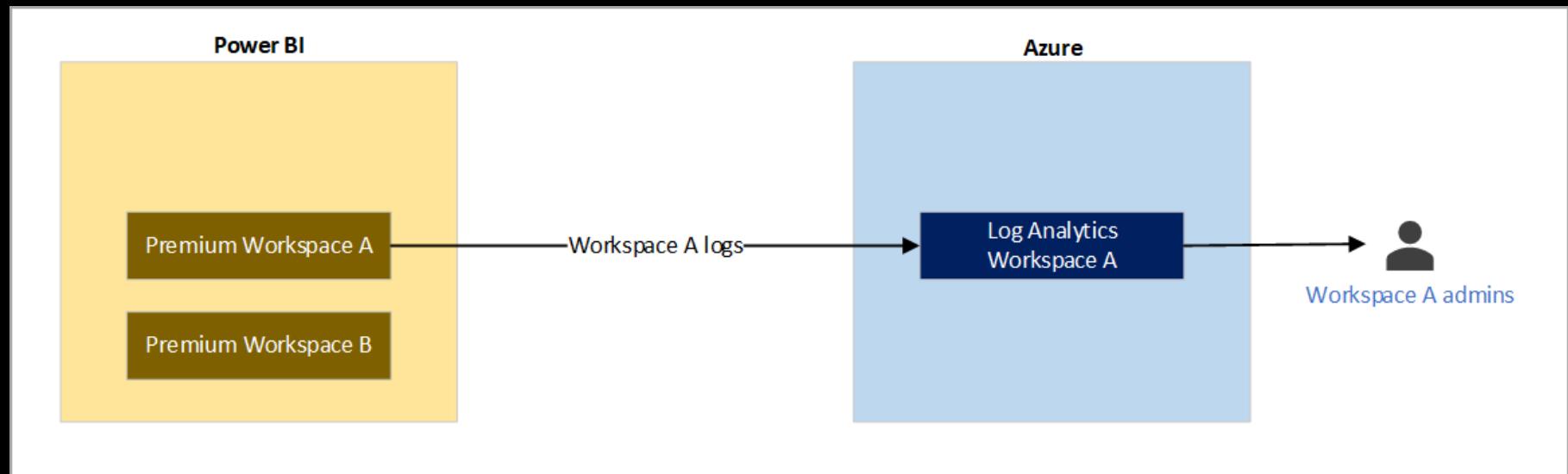


— Capacity metrics app



Power BI Azure Log Analytics (ALA) integration

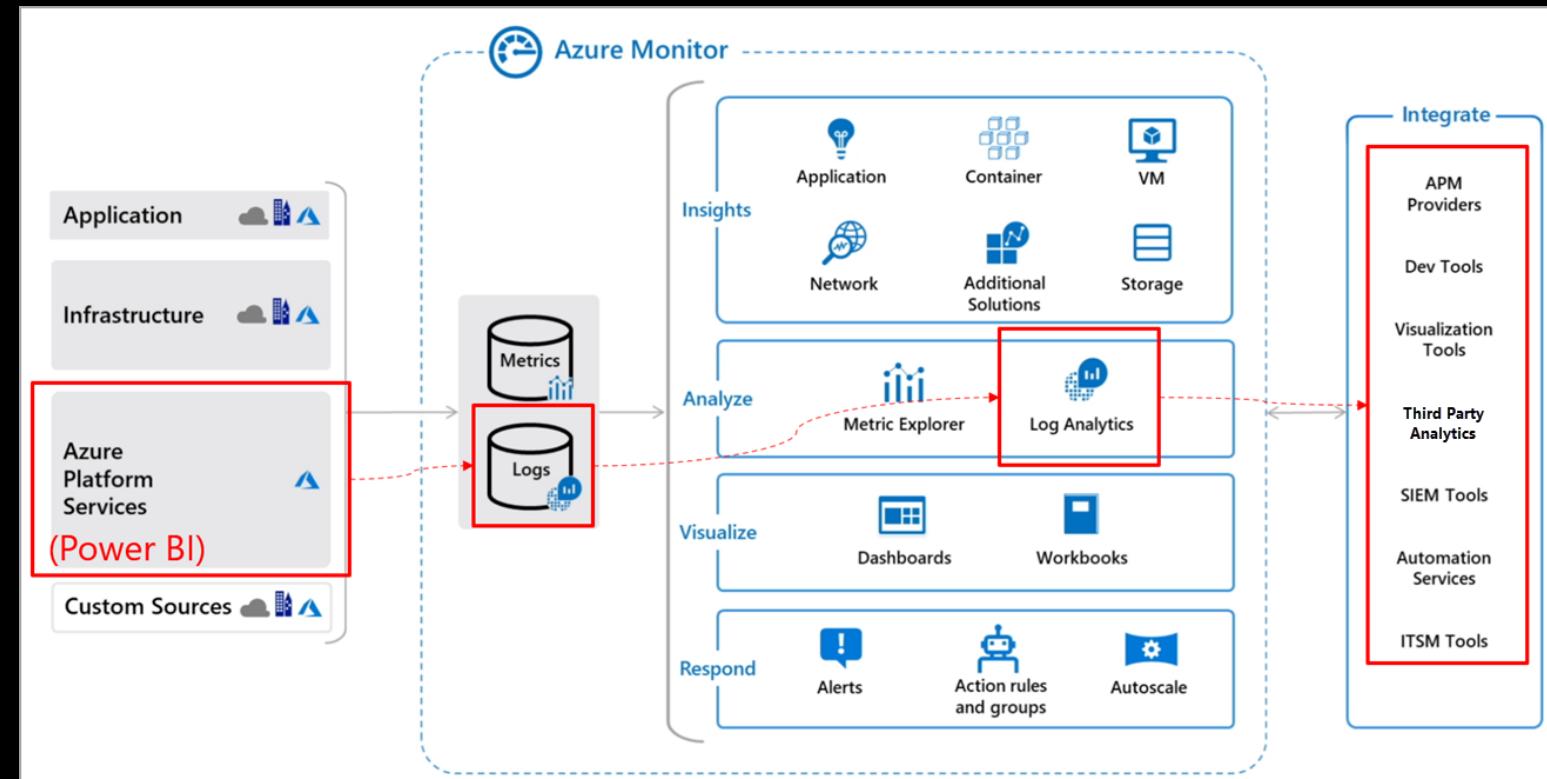
Configure Azure Monitor integration on Power BI workspace level to allow workspace administrators to collect, analyze and act on telemetry data. PBI integration with ALA exposes events from the Analysis Services engine.





Log Analytics Workspace / Azure Monitor

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.



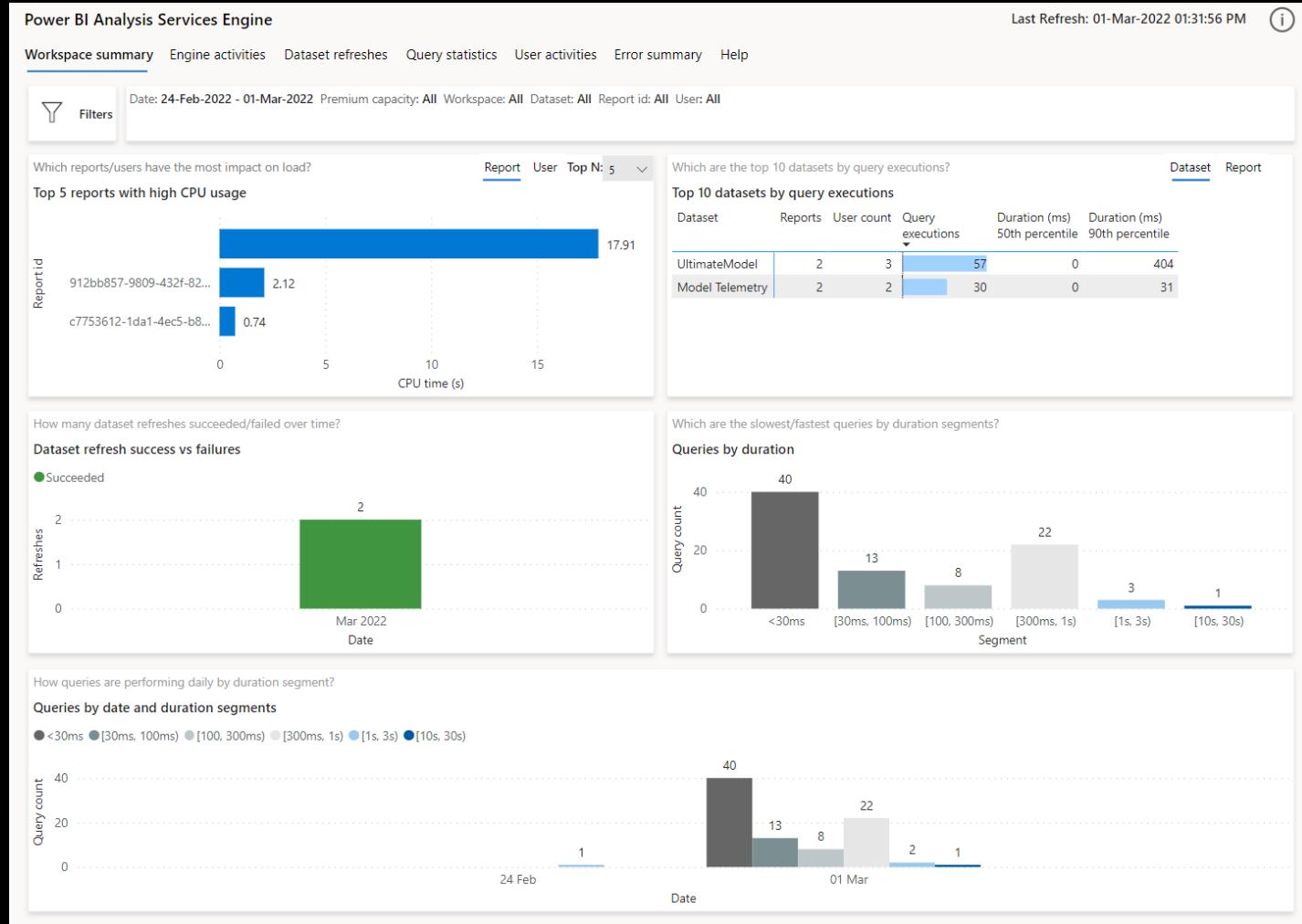


Scenarios

- Identify high or unusual engine activity by capacity / ws / user
- Analyze query performance
- Analyze dataset refresh durations and engine operations
- Investigate impact of custom operations via Premium XMLA



Template report to get insights



- SQL Workspace summary
- SQL Engine activities
- SQL Dataset refreshes
- SQL Query statistics
- SQL Error summary



Power BI Analysis Services Engine

Power BI Analysis Services Engine Last Refresh: 31-Aug-2021 03:23:45 PM ⓘ ⏪

Workspace summary Engine activities Dataset refreshes Query statistics **User activities** Error summary Help

Filters Date: 02-Aug-2021 - 31-Aug-2021 Premium capacity: All Workspace: All Dataset: All Report id: All User: All

Top 5 users by operation

User	Operations
f74879ff81d94807a...	2,548
bd0e6a9d01aeb180d...	2,107
7e5e12d28c27808d1...	2,006
6f80c7eb8ca4b36d06...	1,556
f219bc8d882eddc59f...	1,370

Queries vs CPU time by users

Daily user and operation count

Hourly user and operation count

User details

User	Last active date	Dataset	Report	Operations	Refreshes	Queries	Avg CPU time	Avg duration (ms)
00cb9564a7b034cae5d03714cb23b540	31-Aug-2021	5	1	423	105	12	954,885	120
0128eb4ff809385d9ef4a5f6e72180b0	31-Aug-2021	2	2	62	76	33	63	129
022933a63c6eb677e0433e5924cd6f70	31-Aug-2021	2	2	199	67	53	424	228
0271eb53a3f942a5204a1427a56dfd9f	31-Aug-2021	3	3	438	201	51	171	111
02c4ce099350c133cc2461d8f79aff8	31-Aug-2021	4	4	96	34	57	152	100
02f2bc7a7c21a8069b4447a89ff68ef	31-Aug-2021	2	2	38	7	37	139	63
0394d214768430b3e6866e63a995c5a1	31-Aug-2021	2	2	209	78	100	120	120



Power BI Analysis Services Engine

Power BI Analysis Services Engine | Last Refresh: 01-Oct-2021 01:30:46 PM [i](#) [e](#)

Workspace summary Engine activities Dataset refreshes Query statistics User activities Error summary Help

Filters User: All Scenario: All

CPU time (s) and count of operations by scenarios over period of time

Discover completed Query completed — Count of operations

200

150

100

50

0

0.71K

0.15K

0.01K

800

600

CPU time (s)

Count of operations

Sep 28 Date

Operations (711)

Start date/time	User	Application	Operation	Scenario	CPU time (ms)	Event text	Duration (ms)	Xmla request id
28-Sep-2021 07:21:49 AM	ea1f1a2d4e0f7b00e7908f4e40304		DiscoverEnd	Discover completed	0	<pii><RestrictionList x...	0	f0676ec7-1cb9-43bc-858f-1ac286cfbd81
28-Sep-2021 07:21:49 AM	ea1f1a2d4e0f7b00e7908f4e40304		SessionInitialize	User session started		<pii>MSXInsightsDefau...	0	f0676ec7-1cb9-43bc-858f-1ac286cfbd81
28-Sep-2021 07:21:49 AM	ea1f1a2d4e0f7b00e7908f4e40304		DiscoverEnd	Discover completed	0	<pii><RestrictionList x...	0	d6f37444-c77e-41f4-99c1-8c45298c9a2c
28-Sep-2021 07:21:49 AM	ea1f1a2d4e0f7b00e7908f4e40304		DiscoverEnd	Discover completed	0	<pii><RestrictionList x...	0	6f27dc97-daaf-45e8-bfa4-d425148a93cc
28-Sep-2021 07:21:50 AM	ea1f1a2d4e0f7b00e7908f4e40304		DiscoverEnd	Discover completed	641	<pii><RestrictionList x...	643	6d7e58d3-097e-4a83-83eb-8438aaa5afc2
28-Sep-2021 07:21:56 AM	ea1f1a2d4e0f7b00e7908f4e40304		DiscoverEnd	Discover completed	16	<pii><RestrictionList x...	0	17dc51ed-d032-4886-b1a9-890415fe328d
28-Sep-2021 07:21:56 AM	ea1f1a2d4e0f7b00e7908f4e40304		DiscoverEnd	Discover completed	0	<pii><RestrictionList x...	0	20bc987b-b4cb-419e-a515-00b2fe38f588
28-Sep-2021 07:21:56 AM	ea1f1a2d4e0f7b00e7908f4e40304		SessionInitialize	User session started		<pii>MSXInsightsDefau...	0	17dc51ed-d032-4886-b1a9-890415fe328d
28-Sep-2021 07:21:56 AM	ea1f1a2d4e0f7b00e7908f4e40304		SessionInitialize	User session started		<pii>MSXInsightsDefau...	0	20bc987b-b4cb-419e-a515-00b2fe38f588
28-Sep-2021 07:21:56 AM	ea1f1a2d4e0f7b00e7908f4e40304		DiscoverEnd	Discover completed	16	<pii><RestrictionList x...	0	9fa65e69-635f-424d-b5ac-e4c5b8627146
28-Sep-2021 07:21:56 AM	ea1f1a2d4e0f7b00e7908f4e40304		DiscoverEnd	Discover completed	16	<pii><RestrictionList x...	0	a437c347-8886-464b-8b5d-f339a21dc36
28-Sep-2021 07:21:56 AM	ea1f1a2d4e0f7b00e7908f4e40304		DiscoverEnd	Discover completed	0	<pii><RestrictionList x...	0	a37449d2-882c-47e7-ae82-ecb7182bf43
Total					163,528		152,116	



Power BI Analysis Services Engine

Power BI Analysis Services Engine

Last Refresh: 01-Oct-2021 01:30:46 PM

Workspace summary Engine activities Dataset refreshes Query statistics **User activities** Error summary Help

Filters Date: 26-Sep-2021 - 01-Oct-2021 Premium capacity: All Wokspace: All Dataset: All Report id: All User: All

Operations Queries Top N: 5

Top 5 users by operation

User	Operations
3f7ecc014cc48b5cf2...	222
ea1a7a2ca8ae278b6...	198
335989582a9c58801...	110
27f4484675f0aaaf256...	107
d661cf332ee8e0cbd...	107

Queries vs CPU time by users

Daily user and operation count

Date	Users	Operations
Sep 26	0.00K	0.10K
Sep 27	0.46K	0.01K
Sep 28	0.71K	0.02K
Sep 29	0.01K	0.47K
Sep 30	0.68K	0.01K
Oct 01	0.00K	0.02K

Hourly user and operation count

Hour	Users	Operations
0	0.00K	0.02K
1	0.00K	0.00K
2	0.00K	0.00K
3	0.19K	0.00K
4	0.00K	0.00K
5	0.09K	0.00K
6	0.23K	0.00K
7	0.35K	0.00K
8	0.00K	0.00K
9	0.37K	0.00K
10	0.00K	0.00K
11	0.00K	0.00K
12	0.00K	0.00K
13	0.00K	0.00K
14	0.00K	0.00K
15	0.00K	0.00K
16	0.00K	0.00K
17	0.00K	0.00K
18	0.00K	0.00K
19	0.00K	0.00K
20	0.00K	0.00K
21	0.00K	0.00K
22	0.00K	0.00K
23	0.00K	0.00K
24	0.00K	0.00K

User details

User	Last active date	Dataset	Report	Operations	Refreshes	Queries	Avg CPU time	Avg duration (ms)
3f7ecc014cc48b5cf2...	01-Oct-2021	2	2	222		115	219	228
ea1a7a2ca8ae278b6...	01-Oct-2021	2	2	198		43	133	132
335989582a9c58801...	01-Oct-2021	2	2	110		58	72	104
27f4484675f0aaaf256...	01-Oct-2021	2	3	107		32	316	340
d661cf332ee8e0cbd...	01-Oct-2021	2	2	107		49	75	70
f22a8aef8a6901f8a0d0...	01-Oct-2021	2	2	106		77	156	166
f6c047f0e9879709c07f...	01-Oct-2021	2	2	90		64	186	181
Total	01-Oct-2021	4	6	2,454		959	196	178



Power BI Analysis Services Engine

Power BI Analysis Services Engine

Last Refresh: 31-Aug-2021 03:23:45 PM

Workspace summary **Engine activities** Dataset refreshes Query statistics User activities Error summary Help

Date: 02-Aug-2021 - 31-Aug-2021 Premium capacity: All Dataset: All Report id: All Scenario: All

Filters

How much is the daily CPU usage and operations count by the scenario?

CPU Time (s) and Count of operations by Date and Scenario

Discover completed Query completed Refresh completed Count of operations

CPU Time (s)

Aug 08 Aug 15 Aug 22 Aug 29

23.64K 0.10K 4.80K 0.07K 0.53K 4.15K 3.89K 0.13K 0.14K 0.51K

Capacity | Workspace | Dataset | Report

Operations CPU time (ms) Duration (ms) 50th percentile Duration (ms) 90th percentile

Category	Operations	CPU time (ms)	Duration (ms)	50th percentile	Duration (ms)	90th percentile
A9CE1428-DCB1-4...	28,678	1,857,326	219	856		
b1...	29,345	767,780	0	47		
857ad480-fbee-46de-940f-bd5c9e3e...	9	190	47	71		
MSXIC...	17,260	638,532	0	16		
5203c2d4-7d40-4e1e-9127-210c67a...	6,979	609,304	266	1,189		
a98bbd56-6846-46a4-9ed0-3447c5a...	3,509	414,467	404	2,542		
be5ba108-d7c7-4b4e-8987-25973ee...	1,469	141,679	406	1,141		
60d2bb92-b8be-4934-8f99-a4984aa...	1,641	124,791	281	1,564		
4cd5ce4b-b0a2-4c46-aecc-536c1235...	13	4,486	594	1,812		
MSXI...	4,273	287,424	94	453		
2b2b0b90-79d0-48b3-994f-cea249a...	6,383	271,390	0	16		
861b1d09-7a7c-48c1-9864-43240de...	135	6,862	47	345		
...	4,266	15,107	0	16		
SMSP...	5	1,843	0	0		

What is CPU time, duration and operations count by capacity, workspace, dataset and report?

Which are the peak hours with CPU usage?

CPU time (ms) and count of operations by hour and scenario

Discover completed Query completed Refresh completed Count of operations

CPU Time (s)

Time of day

Count of operations

0.0M 0.1M 0.2M

0.0M 0.1M 0.2M

2.59K 4.41K 5.85K 3.31K 7.79K 4.86K 3.71K 6.25K 1.80K

Data-Marc.com



Power BI Analysis Services Engine

Power BI Analysis Services Engine Last Refresh: 01-Oct-2021 01:30:46 PM ⓘ ⏪

Workspace summary Engine activities Dataset refreshes **Query statistics** User activities Error summary Help

Filters Date: 26-Sep-2021 - 01-Oct-2021 Premium capacity: All Dataset mode: All Dataset: All Report id: All User: All

Query success vs failures count

Date	Successes (K)	Failures (K)
Sep 26	0.03K	0.00K
Sep 27	0.16K	0.00K
Sep 28	0.33K	0.00K
Sep 29	0.15K	0.00K
Sep 30	0.28K	0.00K
Oct 01	0.00K	0.00K

Queries by aggregation usage

No
959 (100%)

Queries by date and segment

Date	<30ms	[30ms, 100ms]	[100ms, 300ms]	[300ms, 1s]	[1s, 3s]	[3s, 10s]
26-Sep-2021	1	4	7	10	11	0
27-Sep-2021	5	48	26	70	10	0
28-Sep-2021	12	87	86	128	19	2
29-Sep-2021	9	40	32	61	8	0
30-Sep-2021	12	79	61	105	19	1
01-Oct-2021	1	2	3	0	0	0

Duration CPU time Top N: 5 ▾

Top 5 queries by duration P50

Capacity Workspace Dataset Report Query	Count	Duration std dev %	Duration (ms) 50th percentile	Duration (ms) 90th percentile	Max duration (ms)
A90... (Planning)	3	15%	7,119	7,486	7,578
b1... (Planning)	0	0%	0	0	0
SMSP... (Planning)	0	0%	0	0	0
602... (Planning)	0	0%	0	0	0
MSXIC... (Planning)	0	0%	0	0	0
a97... (Planning)	0	0%	0	0	0

DEFINE VAR _DSOFilterTable =

Data-Marc.com



Power BI Analysis Services Engine

Power BI Analysis Services Engine Last Refresh: 31-Aug-2021 03:23:45 PM ⓘ

Workspace summary Engine activities Dataset refreshes Query statistics User activities Error summary Help

Filters Date: 02-Aug-2021 - 31-Aug-2021 Premium capacity: All Workspace: All Dataset: All Report id: All User: All

Report User Top N: 5

Top reports with high CPU usage

Report id	CPU Time (s)
602e4211-2dc9-4977-bb...	1857.33
5203c2d4-7d40-4e1e-91...	1694.65
a98bbd56-6846-46a4-9e...	609.30
2b2b0b90-79d0-48b3-9...	414.47
2b2b0b90-79d0-48b3-9...	287.42

Dataset refresh success vs failures

Date	Succeeded
Aug 03	12
Aug 04	26
Aug 05	25
Aug 06	23
Aug 07	9
Aug 08	4
Aug 09	6

Top 10 datasets by query executions

Dataset	Reports	User count	Query executions	Duration (ms) 50th Percentile	Duration (ms) 90th Percentile
SMSPI...	3	414	29,886	47	564
MSXIC...	6	324	13,910	0	833
MSXIC...	3	163	4,454	0	359

Queries by duration

Segment	Query count
<30ms	2,343
[30ms, 100ms]	14,468
[100, 300ms]	11,841
[300ms, 1s]	14,976
[1s, 3s]	4,143
[3s, 10s]	470
[10s, 30s]	7
>30s	2

Queries by date and duration segments

Date	<30ms	[30ms, 100ms]	[100, 300ms]	[300ms, 1s]	[1s, 3s]	[3s, 10s]	[10s, 30s]	>30s
02 Aug	643	3,580	3,517	864	75	2		
03 Aug	436	3,319	3,393	772	120	2	2	
04 Aug	141	678	785	214	22			
05 Aug	67	440	487	20	1			
06 Aug	62	373	422	10				
08 Aug	17	81	72	104	26			



Configuration

Admin portal

Usage metrics

Users

Premium Per User

Audit logs

Tenant settings

Capacity settings

Refresh summary

Embed Codes

Organizational visuals

Azure connections

Workspaces

Custom branding

Protection metrics

Featured content

Connect to Azure resources (preview)

- ▶ Tenant-level storage (preview)
- ▶ Workspace-level storage permissions (preview)

◀ Workspace-level Log Analytics permissions (preview)



Allow workspace admins to connect their own Log Analytics workspace

Save

Cancel



Configuration

The screenshot shows the Microsoft Power BI Admin Center. At the top, there is a navigation bar with a user profile icon, the name "admin" (with a blurred email address), a diamond icon, and a "Create app" button. Below the navigation bar are several tabs: "View", "Filters", "Settings" (which is highlighted with a red box), "Access", and "Search". Underneath these tabs, there are three main categories: "All", "Content", and "Datasets + dataflows". The "All" category is selected and underlined. The "Content" section displays a table with three items:

Name	Type	Owner	Refreshed	Next refresh
Customer360 TEST	Report	admin [blurred]	10/12/20, 11:05:34 AM	—
Customer360 TEST	Dataset	admin [blurred]	10/12/20, 11:05:34 AM	N/A
Customer360 TEST.pbix	Dashboard	admin [blurred]	—	—



Configuration

Settings

About Premium Azure connections (preview)

▶ Storage

◀ Log Analytics

Connect an Azure Log Analytics workspace to collect usage and performance logs for this workspace. [Learn more](#)

Connect to Azure

Subscription
Select an Option

Resource group
Select an Option

Log Analytics workspace
Select an Option

Save **Cancel**

Settings

admin [REDACTED]

About Premium Azure connections (preview)

▶ Storage

◀ Log Analytics

Connect an Azure Log Analytics workspace to collect usage and performance logs for this workspace. [Learn more](#)

Subscription a1 [REDACTED] 7e

Resource group rg-byola

Log Analytics workspace admir [REDACTED]

Configured by Admin Admin on 2020-10-12T12:00:00Z

Disconnect from Azure

Disconnecting will stop usage and performance data from flowing into your Azure Log Analytics workspace.

Disconnect from Azure



Configuration

Go to [AppSource > Power BI Log Analytics for Analysis Services Engine](#)

and select **Get it now.**

The screenshot shows the Microsoft AppSource page for the "Power BI Log Analytics for AS Engine (Preview)" app. The page includes the app's logo, a "Get it now" button, and a section titled "At a glance" showing five sample dashboards. Below this, there are sections for "Other apps from Microsoft" featuring the "Power BI Premium Capacity Utilization app..." and the "Power BI Premium Capacity Metrics App".

Power BI Log Analytics for AS Engine (Preview)

At a glance

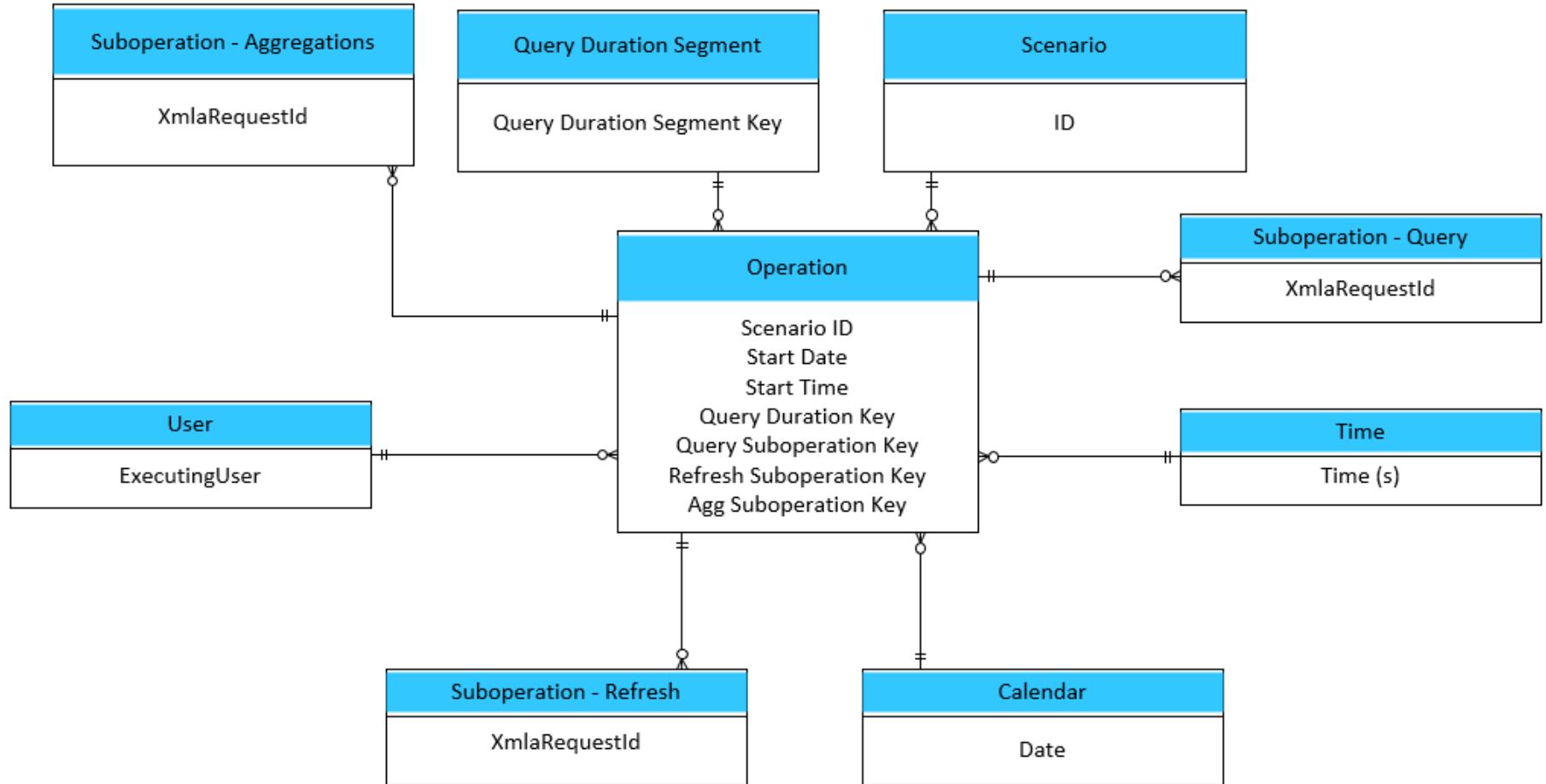
Other apps from Microsoft

- Power BI Premium Capacity Utilization app...**
- Power BI Premium Capacity Metrics App**



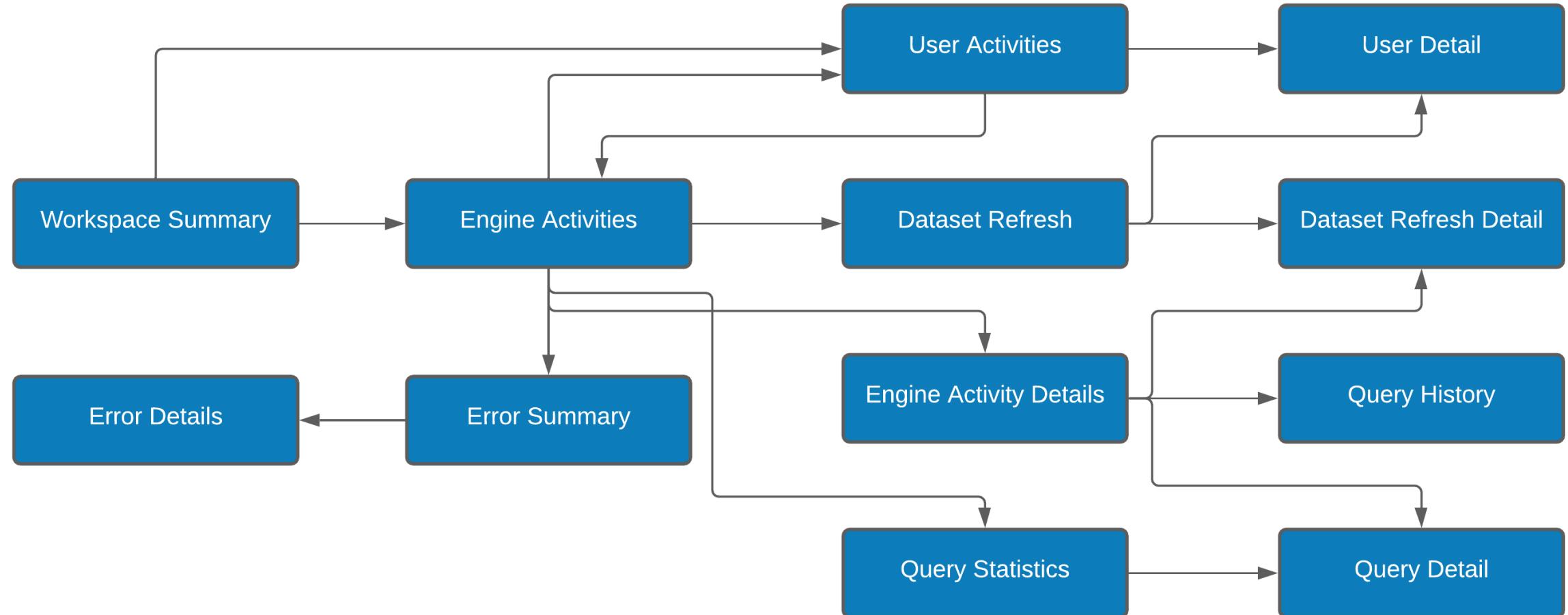
Report - ERD

ER Diagram





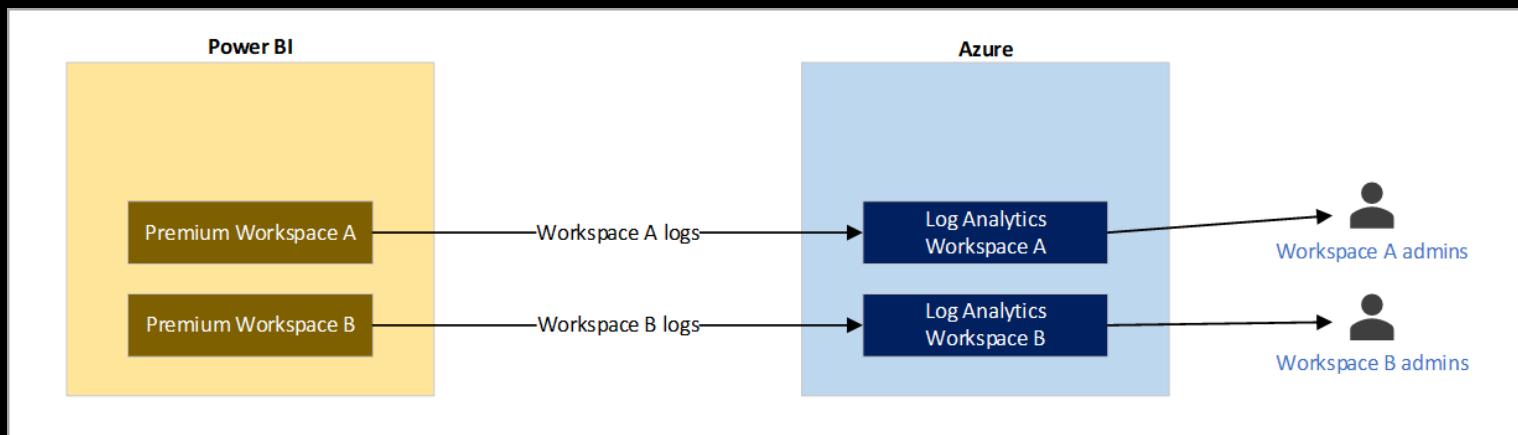
Report - workflow





Configuration pre-requisites – Generic

- Only Premium
- Only v2 workspaces
- Only for the datasets in the workspace (no shared datasets)
- 1:1 connection between Power BI workspace and ala workspace





Configuration pre-requisites – Azure

- Register '**microsoft.insights**' resource provider in subscription
- Create Log Analytics Workspace
- Owner role on the Log Analytics Workspace for:
 - The user who will set up Log Analytics integration in Power BI
 - The service principal '**Power BI Service**'
- Note: change the **retention period**



Cost info

REGION:

West Europe

Log Analytics

€105.53

- (i)* Daily log data ingested will depend on what you are monitoring with Log Analytics. [Learn more](#) about estimating data volumes.

Data Ingestion

1	×	30	×	€2.68	=	€67.05
DAILY LOGS INGESTED (GB/DAY)						

- (i)* This estimate is calculated using the most optimal pricing tier for the data ingestion. This calculation uses **Pay-As-You-Go tier**. [Learn more](#) about the pricing tiers

Data Retention

- (i)* The first month of retention is free

12						
TOTAL RETENTION (MONTHS)						
30	×	11	×	€0.12	=	€38.48
Total monthly ingestion (GB)		Additional retention (months)		Per GB/month		



Ad-blocker?

The screenshot shows a dark-themed web interface with a navigation bar at the top featuring icons for search, refresh, and user profile. The profile section displays the email address inbox@daveruijter.nl and the name MARC LELIUVELD next to a blue circular profile picture.

Notifications

[More events in the activity log →](#) [Dismiss all](#)

! Failed to retrieve schema. Please try to refresh the page.

We have experienced a connection issue while retrieving data. This is usually an indication that the network is down or a firewall or browser extension (such as an ad blocker) is mistakenly preventing access.

Connection Error
If issue persists, please open a support ticket. Request id: 256ea54b-65a6-4257-b248-a000e699c24e

a few seconds ago



Example KQLs

```
// log count per day for last 30d  
PowerBIDatasetsWorkspace  
| where TimeGenerated > ago(30d)  
| summarize count() by format_datetime(  
TimeGenerated, 'yyyy-MM-dd')
```

```
// average query duration by day for last 30d  
PowerBIDatasetsWorkspace  
| where TimeGenerated > ago(30d)  
| where OperationName == 'QueryEnd'  
| summarize avg(DurationMs) by  
format_datetime(TimeGenerated, 'yyyy-MM-dd')
```

```
//query duration percentiles for a single day in 1 hour bins  
PowerBIDatasetsWorkspace  
| where TimeGenerated >= todatetime('2021-04-28') and  
TimeGenerated <= todatetime('2021-04-29')  
| where OperationName == 'QueryEnd'  
| summarize percentiles(DurationMs, 0.5, 0.9) by  
bin(TimeGenerated, 1h)
```

```
// refresh durations by workspace and dataset for last 7d  
PowerBIDatasetsWorkspace  
| where TimeGenerated > ago(30d)  
| where OperationName == 'CommandEnd'  
| where ExecutingUser contains 'system'  
| where EventText contains 'refresh'  
| project WorkspaceName, DatasetName = ArtifactName,  
DurationMs
```



Using the template report

Power BI Log Analytics for Analysis Services Engine

This template will connect to the pre-configured Azure Log Analytics workspace where you are collecting Analysis Services engine log data. The template can be used to monitor engine load, engine usage trends, and identify slow queries and dataset refreshes. It provides summary and trend information at Workspace level with the ability to drill to detail about any individual query or dataset refresh. The template allows you to group or slice the data by useful properties such as Premium capacity, dataset, report, and user.

Days Ago To Start ⓘ:

Days Ago To Finish ⓘ:

Log Analytics Table ⓘ:

Log Analytics Workspaceld ⓘ:

UTC Offset Hours ⓘ:

Pagination Hours ⓘ:



GitHub location for .pbix

<https://github.com/microsoft/PowerBI-LogAnalytics-Template-Reports>



Customizations

- Visual ID
- Report Name
- DirectQuery



DirectQuery on the trace info

- Which datasets are refreshing right now?
- Which recent datasets refreshes succeeded, and which ones failed?
- If a dataset refresh failed, why did it fail?
- How long did these refreshes take?



Roadmap

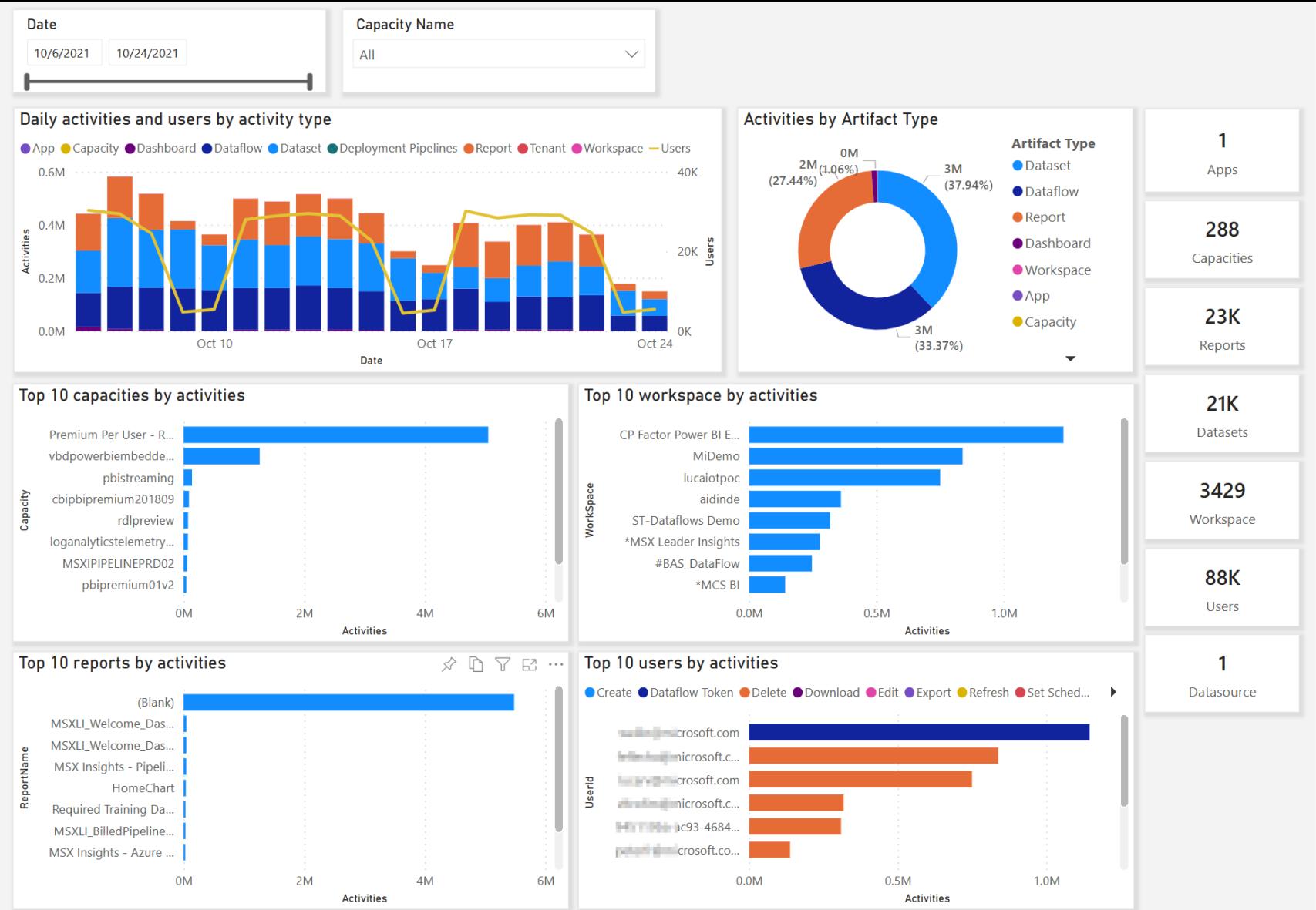
- **SQL** GA → March?
- **SQL** Removing the 1:1 workspace mapping restriction
- **SQL** No tenant level configuration to be expected
- **SQL** Capacity level configuration to be expected
- **SQL** API control to configure multiple workspaces at once



— Demo Log Analytics Integration



Power BI Usage Metrics





Power BI Usage Metrics

Power BI Usage Metrics

Tenant summary **Workspace summary** Report summary User summary Error summary Help

Date: 15-Jul-2021 - 27-Jul-2021 Workspace: All Report: All Dataset: All Dataset mode: All Embed type: All

Filters

Which workspaces/datasets have high report usage? Workspaces Dataset Top N: 5

Top 5 workspaces by report open

Workspace name	Report opens
{PII:H106(zTmtlWR4g...}	51,618
{PII:H106(R4uktt6iCjo...}	35,573
(Blank)	10,765
{PII:H106(NgsAW5FN...}	3,366
{PII:H106(SROG8SUW...}	2,190

Which reports have high usage? Top 5 reports by report open

Report name	Report opens
{PII:H106(aue6kKRTI...}	5,512
{PII:H106(pnS8IAHrj...}	1,935
{PII:H106(hcMMdkUy...}	1,793
{PII:H106(Ev5ss4H9D...}	1,110
{PII:H106(pRYuwx8X...}	956

4,274 Workspaces

3,549 Reports

3,354 Datasets

1 Users

130,931 Report opens

15 Report open failed

Which are the most active users? Top 5 report by unique users

Report name	User count
{PII:H106(/kViggPuA...}	1
{PII:H106(/IHZu+Fo...}	1
{PII:H106(/R1AzVY2...}	1
{PII:H106(/uY8vrMI...}	1
{PII:H106(/+nWsCl94...}	1

Which are the slowest reports? Top 5 reports by total duration

Report name	Total duration(sec)
{PII:H106(0ubSWx1/...}	25355.17
{PII:H106(09KXLZEgf...}	17923.24
{PII:H106(cVFeLt8krff...}	12342.37
{PII:H106(KTYrAMnhr...}	10240.25
{PII:H106(JBZUXn4e...}	9723.48

What is report usage by consumption/distribution method? Report views by consumption method

Consumption	Distribution
Power BI Web	
Embedding for your or...	
Power BI Mobile	
Simplified Embedding	
Embedding for your cus...	

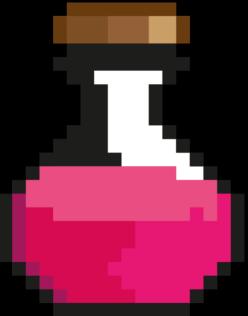
4,423 (3.38%)
42,847 (32.72%)
79,464 (60.69%)

What is daily report usage? Daily users and report views

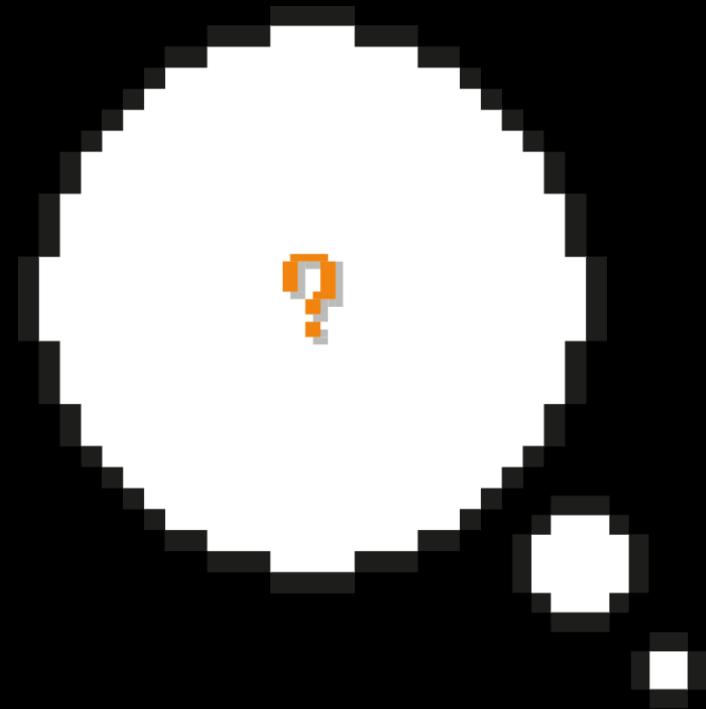
User count Report opens

Jul 15 Jul 22 Date

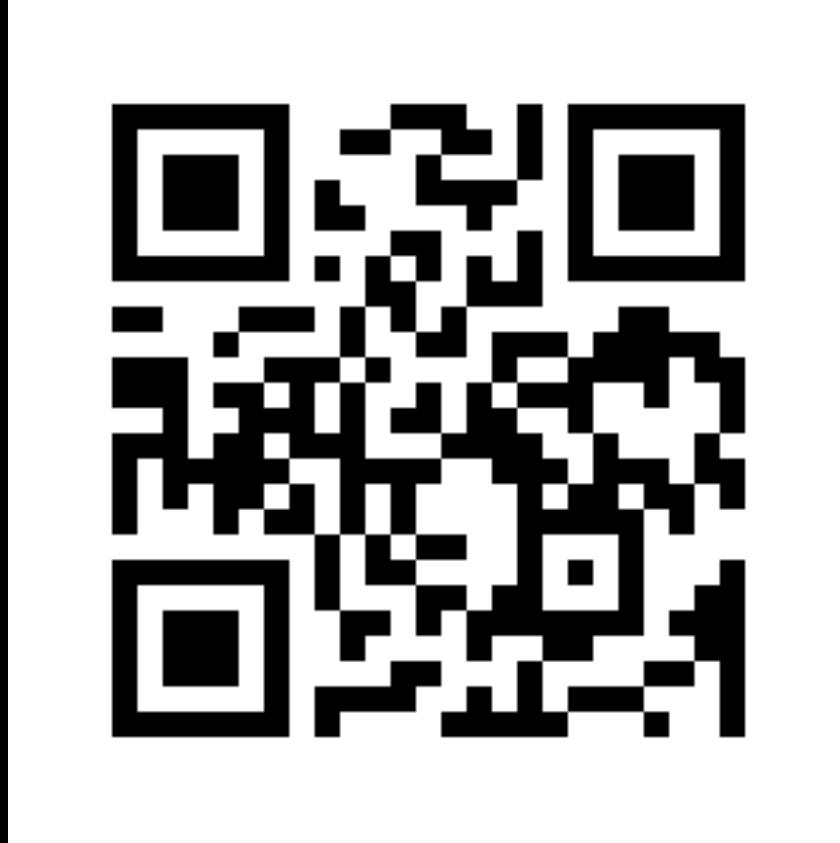
Wrap-up



Design and implement	Orchestrate	Performance	Cost
Better design and implement complex data models, including hybrid tables, aggregations, and combined storage modes (import, DirectQuery , dual).	Orchestrate the end-to-end data processing, with a pipeline chain from data ingest in the data lake house to the incremental Power BI dataset refresh.	Use different techniques to identify performance bottlenecks in your solutions and how to solve those ("does it fold"?).	Implement a cost-efficient solution, that still meets the scalability demands.



Any bonus questions?



Feedback
Feedback
Feedback



<https://sqlb.it/?6199>