Microsoft Fabric

#FABCONSQLCON2026

FABCON
Microsoft Fabric
COMMUNITY CONFERENCE

SQLCON
Microsoft SQL
COMMUNITY CONFERENCE

ATLANTA MARCH 16 - 20, 2026

Performance Tuning for Fabric Data Factory

Mohan Sankaran
VP Engineering
Microsoft Data Integration

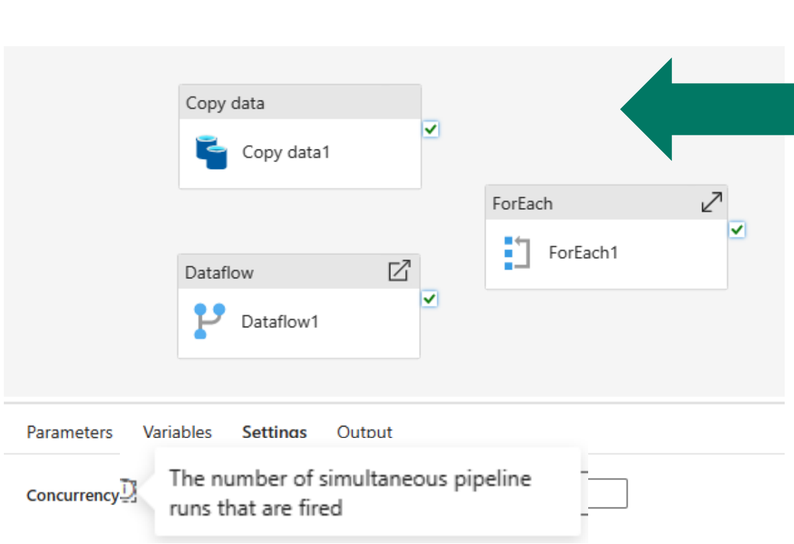
Chris Webb
Principal Program Manager
Fabric Customer Advisory Team

Pipelines

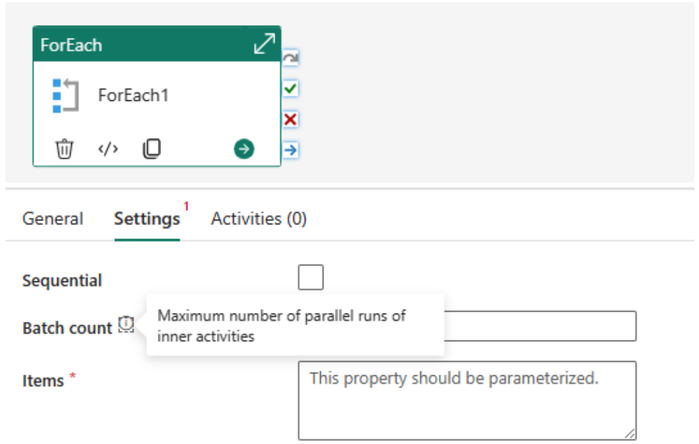
Pipelines Tip

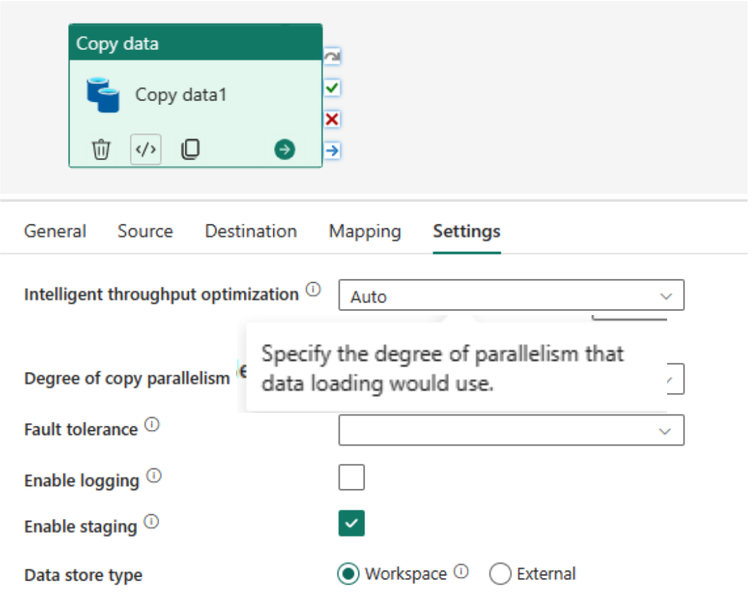
Parallelism vs. Sequential Execution

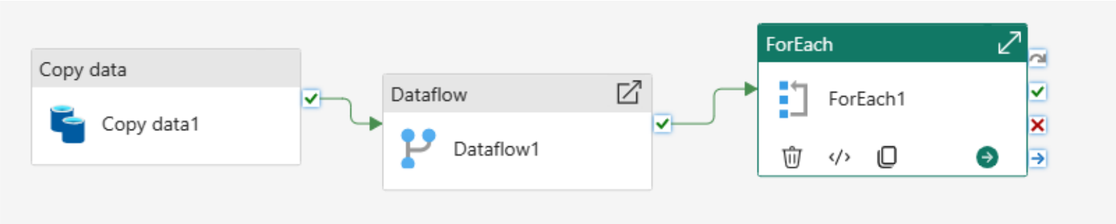
Multiple levels of parallelism available to you as a pipeline developer in Data Factory



These activities will all run in parallel



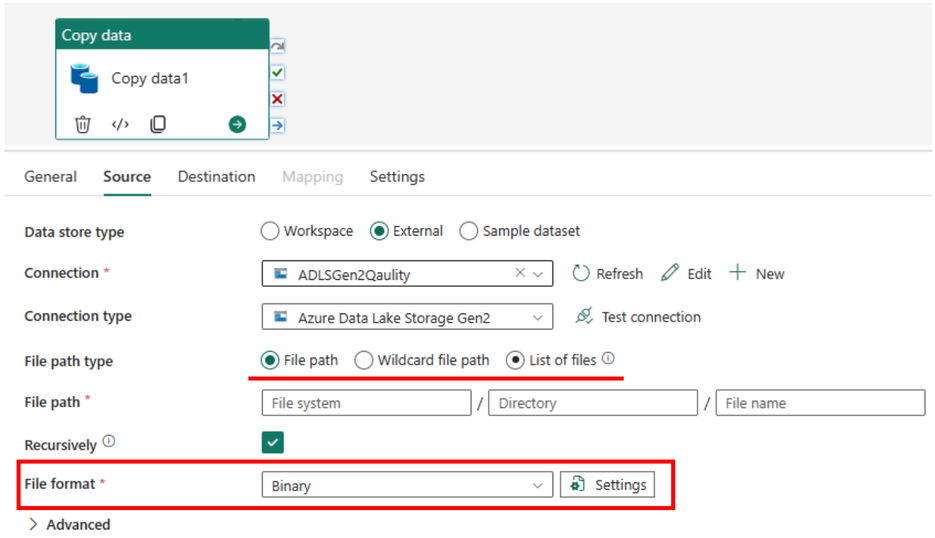


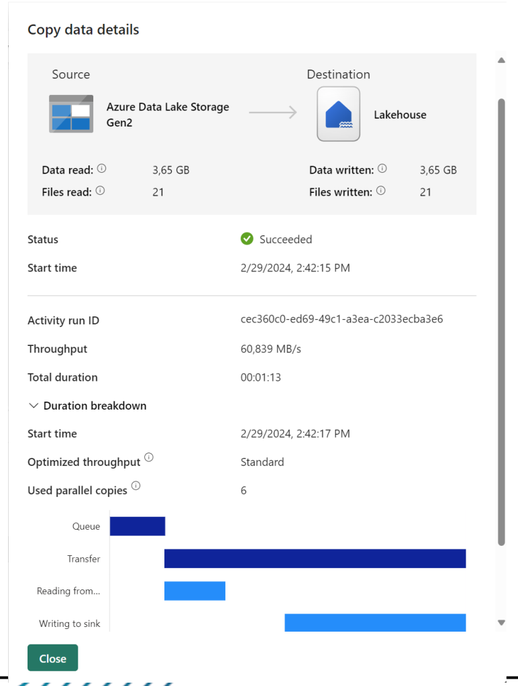


These activities will all run in sequence

Pipelines: Copy Activity

- Utilizing binary copy can improve load times to LH Files or storage accounts by 60-70%
- New Fabric Copy optimization for binary to improve data movement speeds 2x
- Tip1: Wildcards, list of files, and folders inside Copy source rather than iterate over many files using For-Each at pipeline level
- Tip2: Use For-Each with multiple Copy activities at pipeline level when
  - Single Copy activity uses up all its resources (maximum ITO/DIU with CPU/memory-intensive ops)
  - The data source is not supported to be partitioned and read in parallel by Copy activity



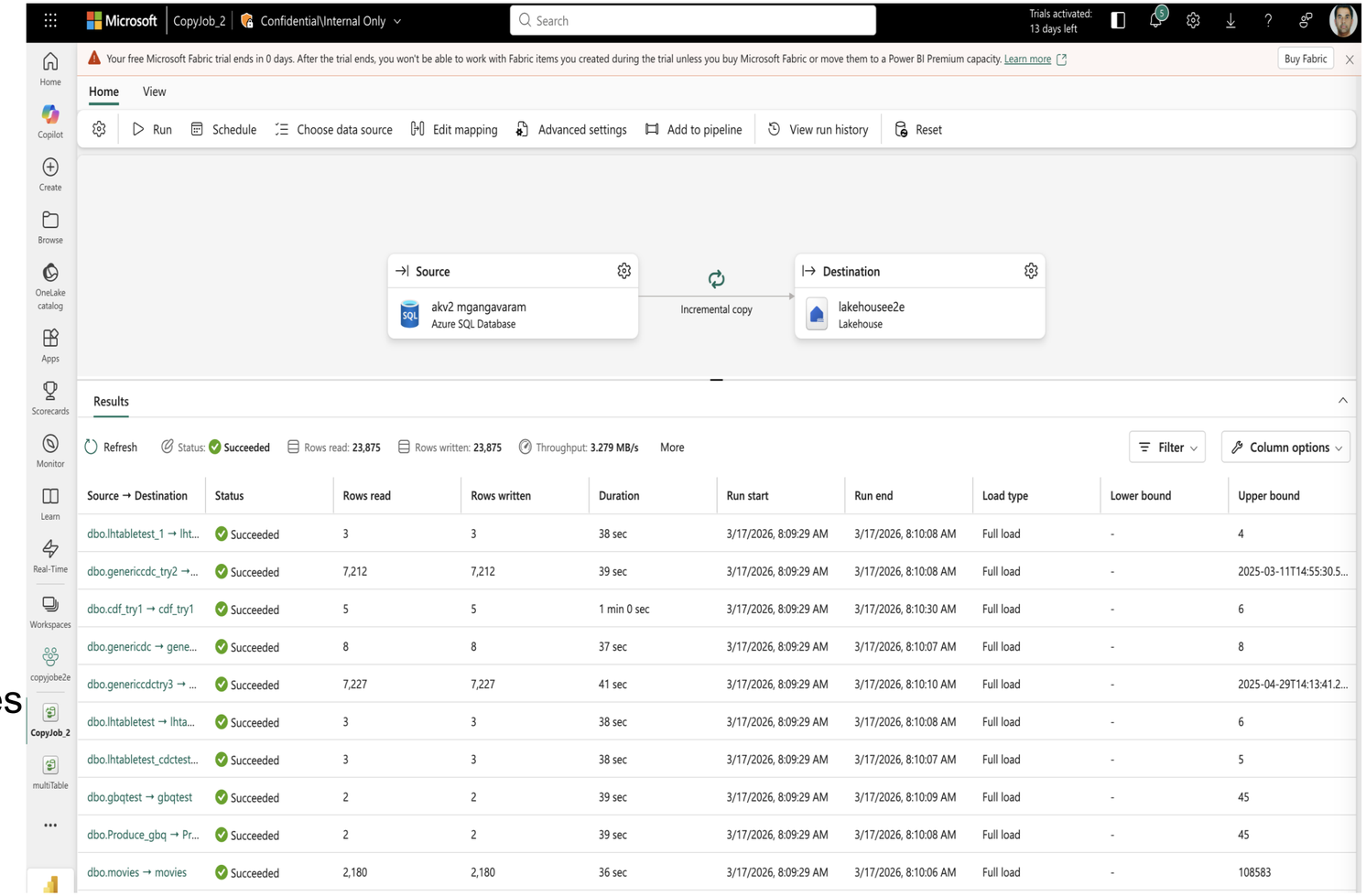


LFMM for a Single Copy Activity

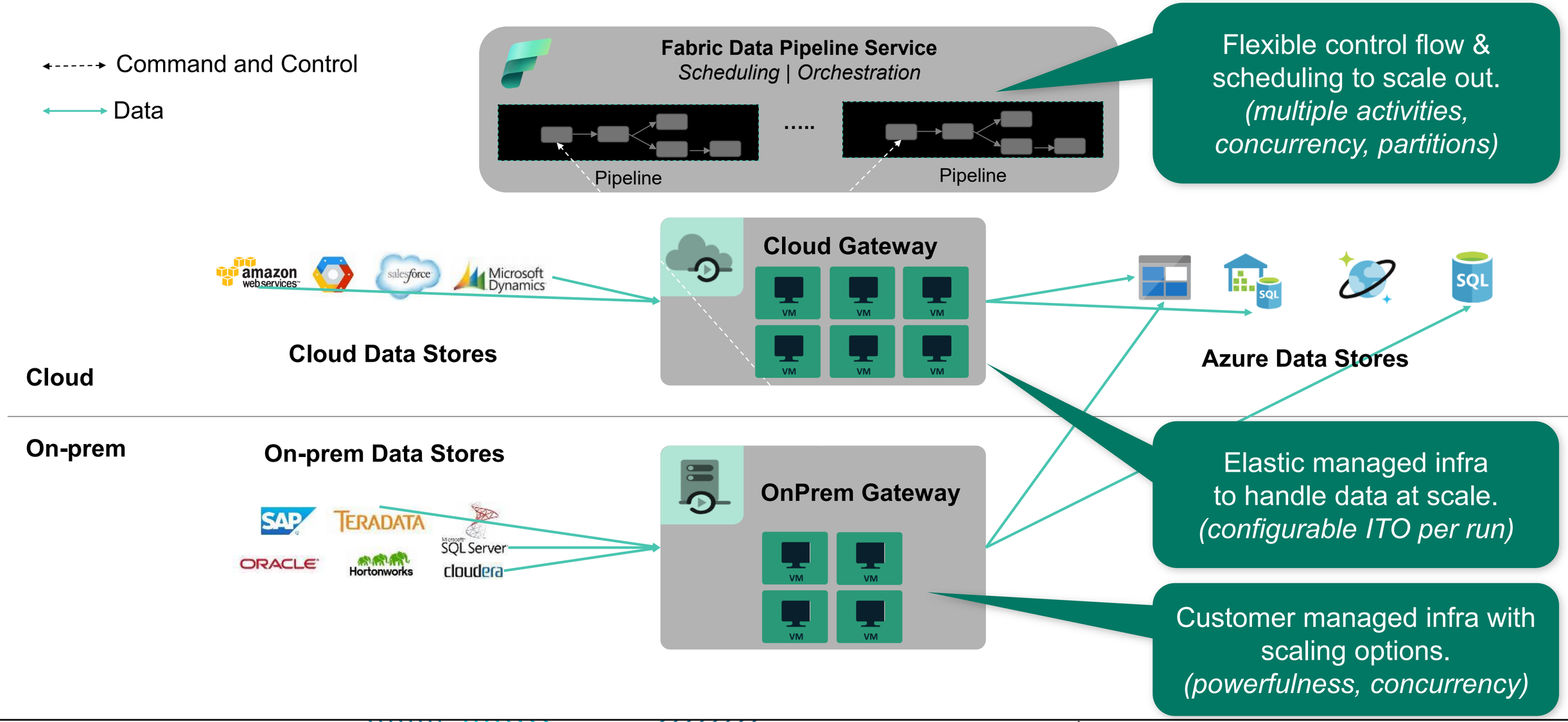
- **L**argest: 0.18 PB
- **F**astest: 6.02 GB/s
- **M**ost Files: 21.38 Million
- **M**ost Rows: 710.29 Billion

CopyJob

- One-stop solution for all data movement needs
- Bulk Table Selection and Mapping to Sink
- Column Mapping Rules for flexible schema transformations
- Batch, Incremental (Watermark) and CDC load capabilities
- Intelligent Auto-Scaling for optimal resource utilization
- High-performance engine for large-scale data movement
- Seamless integration with Data Factory Pipelines

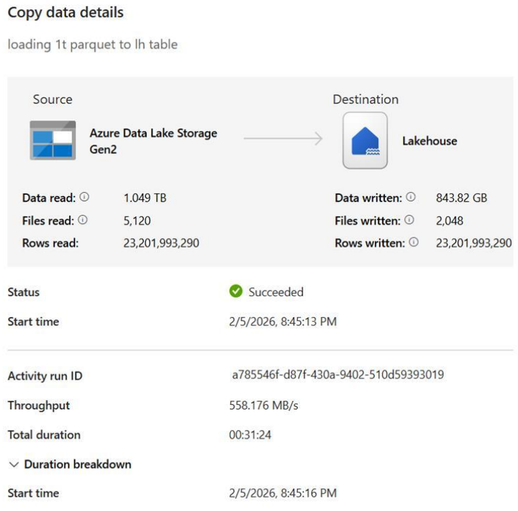


Understand How Pipeline Scales



Copy Performance Metrics (Parquet to Lakehouse)

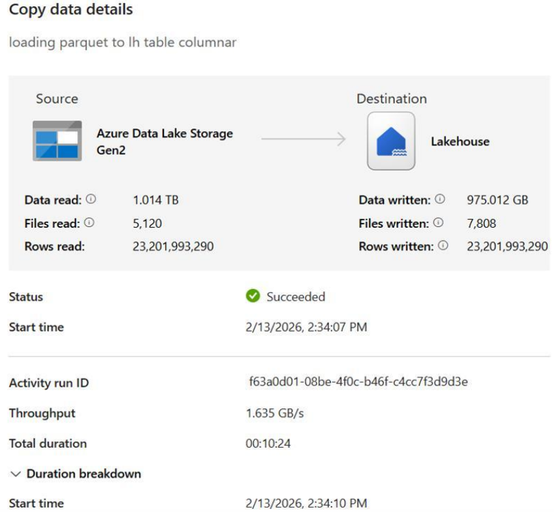
Parquet -> LH (Normal)



558 MB/s

31 min 24 sec

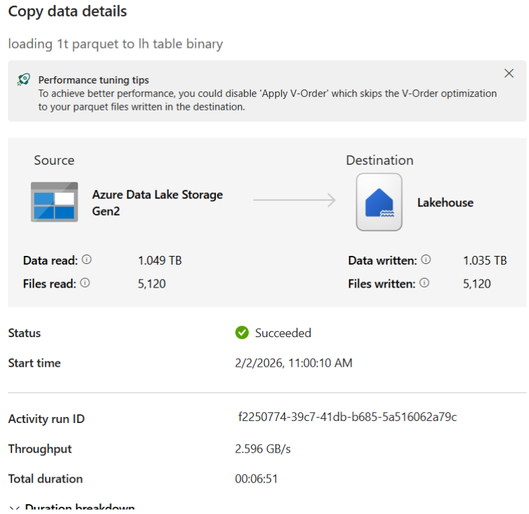
Parquet -> LH (Columnar) (*when schema mismatch*)



1.635 GB/s (3x)

10 min 24 sec

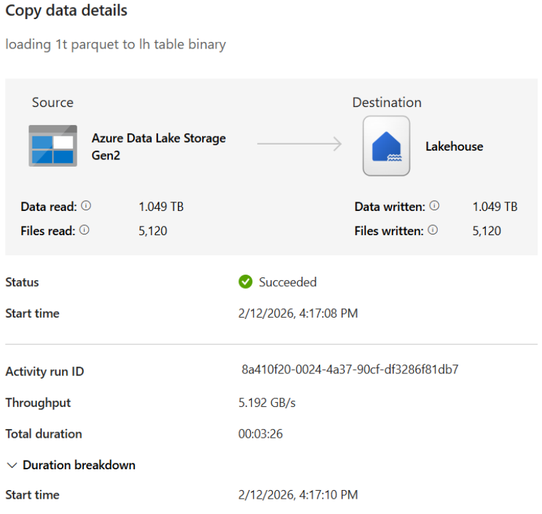
Parquet -> LH (V-Order) *(when schema match)*



2.596 GB/s (5x)

6 min 51 sec

Parquet -> LH (Binary) *(when schema match)*



5.192 GB/s (10x)

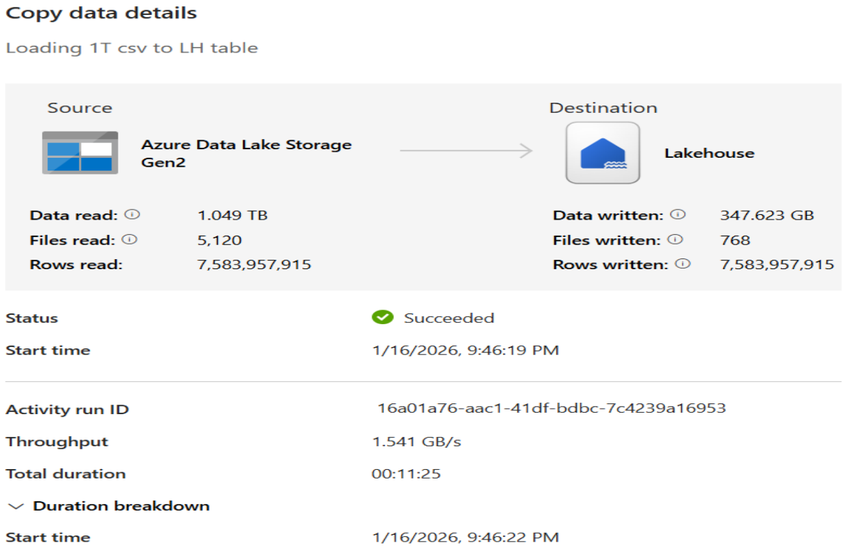
3 min 26 sec

Before

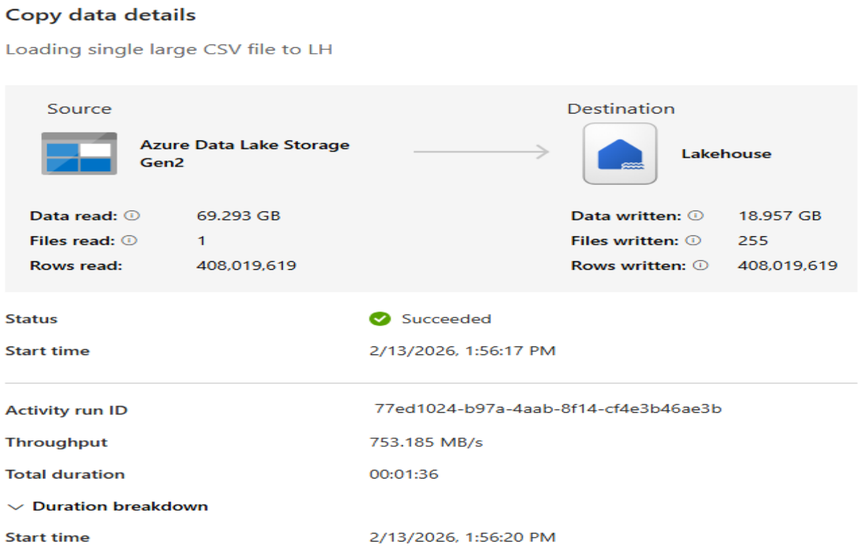
After

Copy Performance Metrics (CSV -> Lakehouse)

CSV Folder -> LH



CSV File -> LH



1.541 GB/s

11 min 25 sec

753.18 MB/s (50x)

1 min 36 sec

Throughput & Total Duration

Copy Performance Metrics with Warehouse

Parquet -> Warehouse (Direct)

Copy data details

loading 1t parquet to dw directly

Status	Succeeded
Start time	1/17/2026, 11:45:17 AM
Activity run ID	0eccbd5f-77ef-439a-a30b-41d96e75797f
Throughput	1.702 GB/s
Total duration	00:10:22
Duration breakdown	
Start time	1/17/2026, 11:45:19 AM

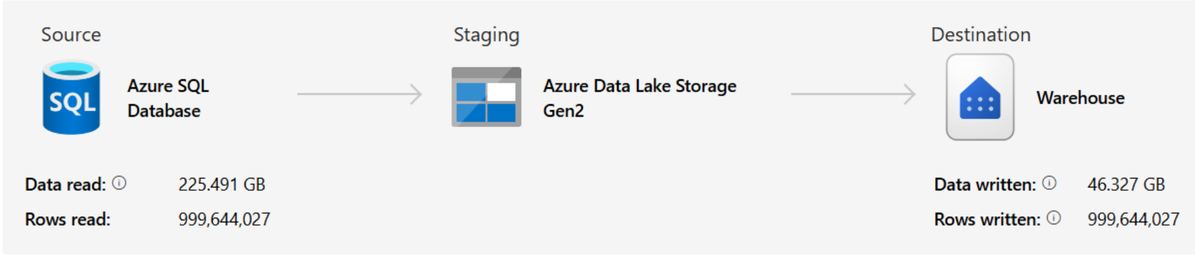
1.702 GB/s

10 min 22 sec

SQL -> Warehouse (Staging)

Copy data details

Loading 1 billion rows from SQL to DW



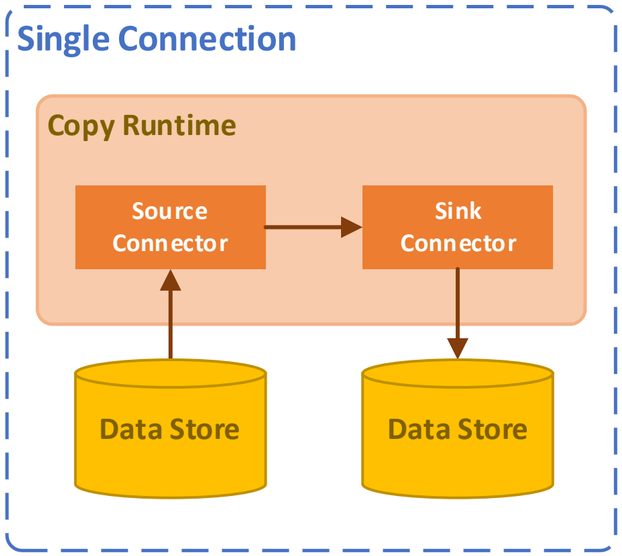
Status	Succeeded
Start time	2/13/2026, 2:30:18 PM
Activity run ID	a414ca18-30ac-402d-bdf9-365101d8ea15
Throughput	1.427 GB/s
Total duration	00:02:46
Source to Staging duration	00:01:46
Staging to destination duration	00:00:59

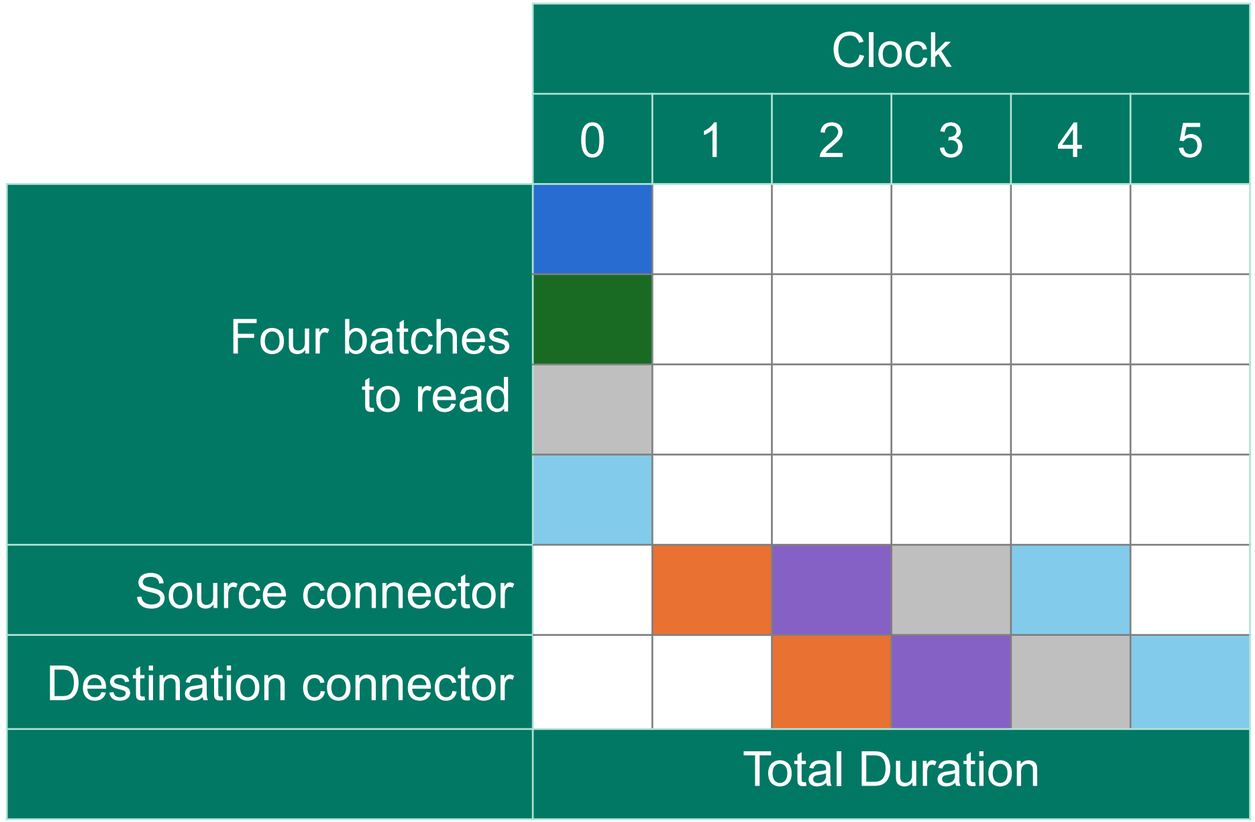
1.427 GB/s

2 min 46 sec

Throughput & Total Duration

How Copy Scales – Connection Level

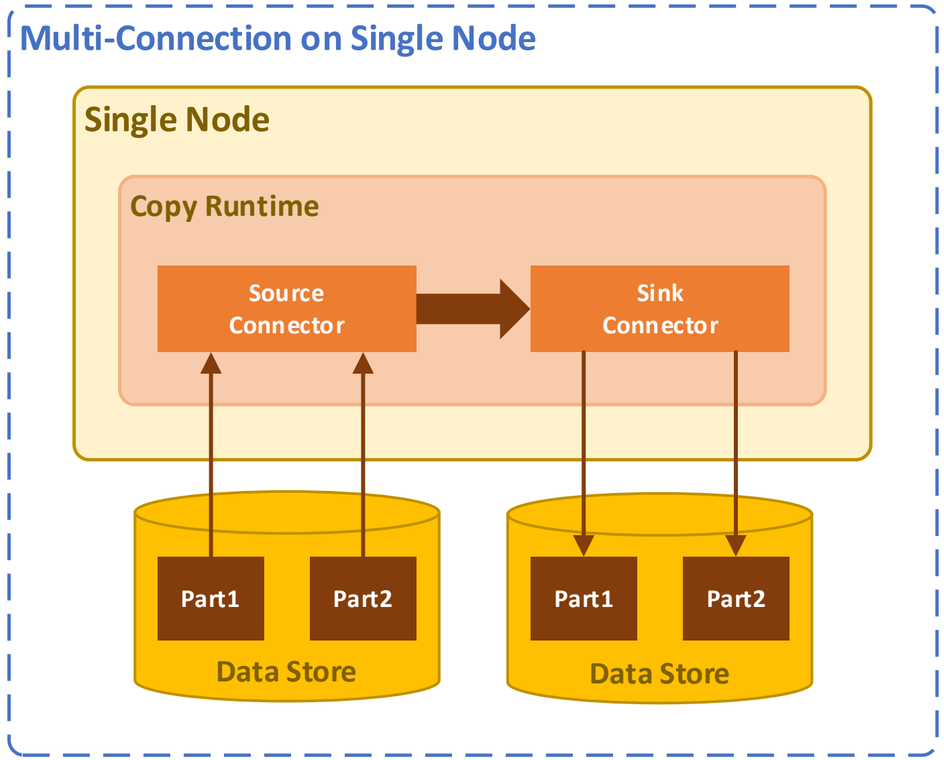




Pipeline processing

- **Less memory**: No need to load everything in memory and then write
- **Less total duration**: Read and Write are in parallel

How Copy Scales – Node Level



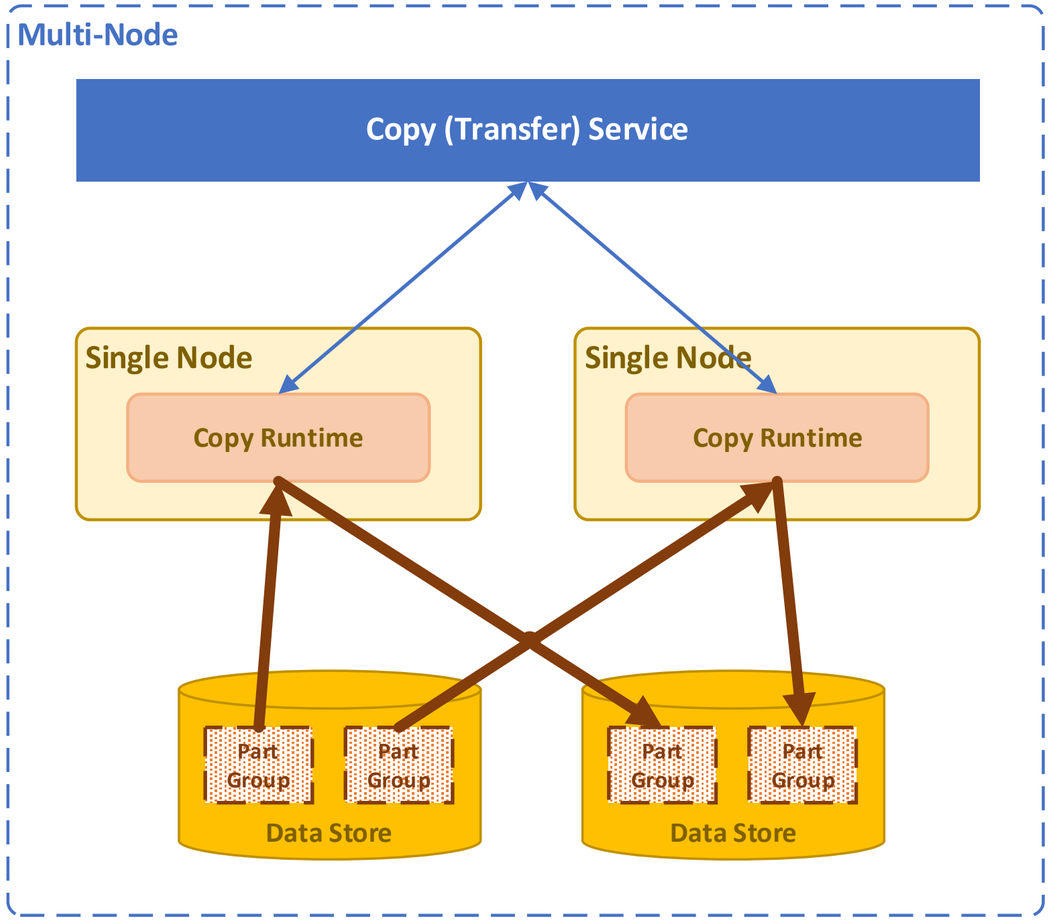
Producer & Consumer design

- Data is partitioned for multiple concurrent connections (even for single large file)
- Full node utilization: Data can be partitioned differently between source and destination to avoid any starving / idle connection

Partitions come from

- Physical partitions setup on DB side
- Dynamic partitions from different queries
- Multiple files
- Multiple parts from a single file

How Copy Scales – Multi-Node Level



Copy(Transfer) Service: Manages Copy activities

- Copy State Management / Monitoring
- Cross Machine Parallelism
- Billing
- Manages Compute Clusters and tasks running on them

Easily scale out to multiple stateless nodes when available partitions are more than what one node can afford.

Theoretically no upper limit on the performance.

Copy Design Considerations

Concepts

- **Intelligent Throughput Optimization (ITO)**: A measure that represents the power (a combination of CPU, memory, and network resource allocation) used for a single Copy activity.
- **Parallel Copy**: The maximum number of threads within the Copy activity that read from source and write to destination in parallel.
- **Max Concurrent Connections**: The upper limit of concurrent connections established to the data store during the activity run. (Usually used to avoid throttling)

Connector Partitioning Behavior

- Files based connectors
  - **CSV files**: Data is split into partitions of 256 MB each
  - **Other file formats**: One file = one partition
- Tabular connectors with partition read support
  - MSSQL family, Oracle, Teradata, Azure PostgreSQL, Netezza, SAP HANA, SAP Table, SAP Open Hub
- Tabular connectors with auto-partition support
  - Lakehouse Table, Salesforce
- **Default perf settings**
  - Intelligent throughput optimization: **Auto** (Use maximum available VM resources)

CopyJob Optimizations

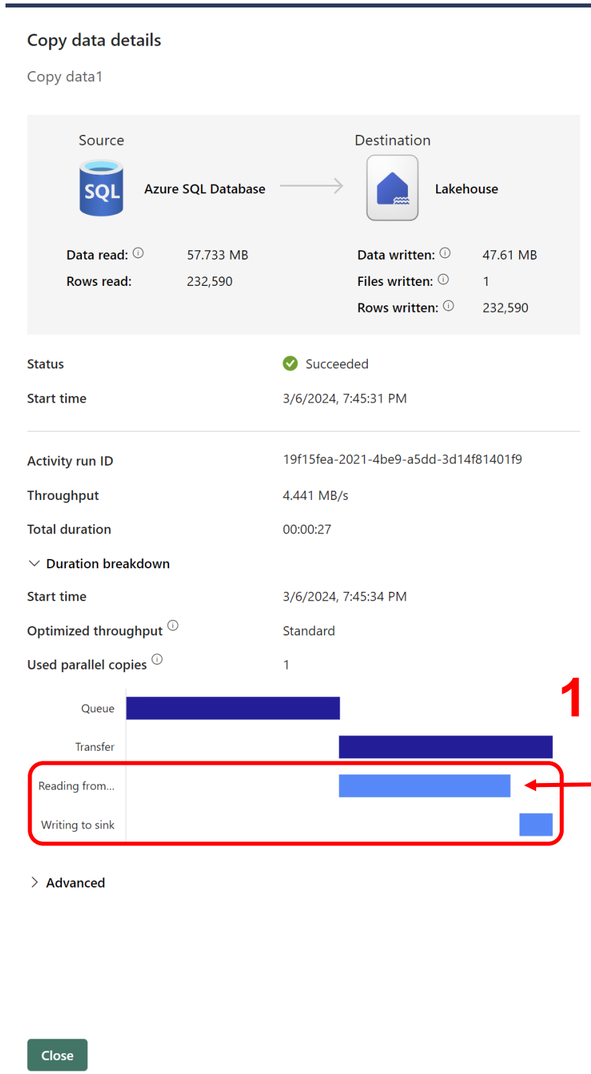
- CopyJob supports trickle feed patterns for near real-time data ingestion
- Schedule CopyJob at frequent intervals (e.g., every 5–15 minutes) to continuously move delta changes
- Incremental (Watermark) and CDC modes ensure only new/changed data is picked up each run
- CopyJob now introduces automatic partitioning for tabular and file sources
- No manual partition configuration required — CopyJob intelligently determines optimal partitioning strategy
- Adaptive batch sizing — dynamically adjusts batch sizes based on source throughput and sink write latency
- Enhanced parallelism controls — fine-grained tuning of concurrent table copies for maximum throughput across multiple tables
- Cost-aware scaling — balances performance with capacity unit consumption

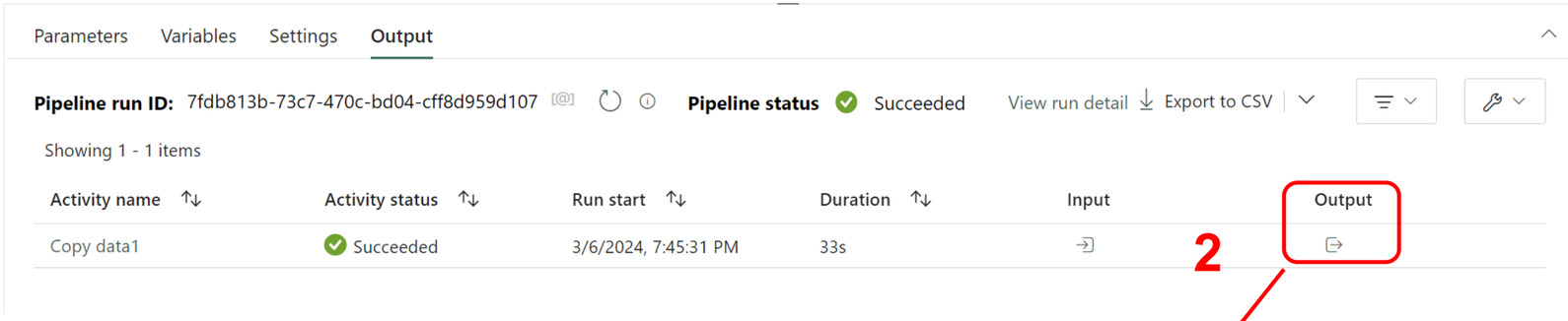
Troubleshoot Performance Issues

Common bottlenecks

- Network (cross region/OnPrem)
- Source/Destination data stores
  - Busy neighbors on the same data store
  - Low service/compute tier
  - Throttling mechanism on data store
  - Complex/unoptimized query (huge timeToFirstByte)
- Compute/Memory-intensive Operations
  - Serialization / Deserialization / Compression / Decompression (e.g. gzip)
  - Data Type conversion (e.g. string to number)
  - Row-Column Transpose (e.g. CSV to Parquet)

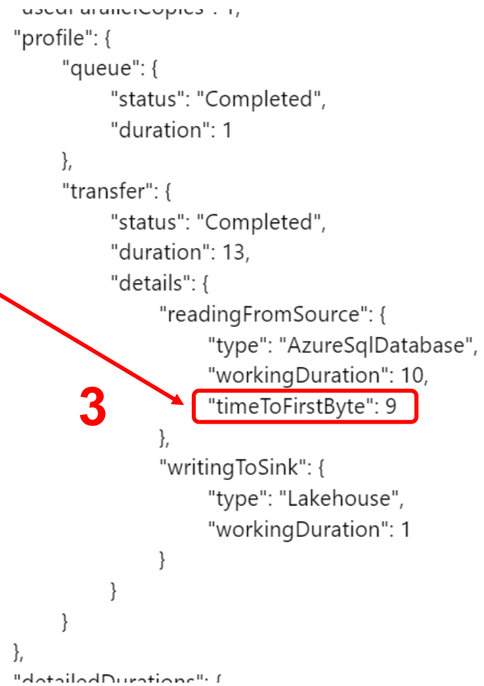
Troubleshoot Performance Issues – Example





Indicating that most time was spent on preparing/running the query on DB side. Optimize the query or upgrade DB tier to eliminate this bottleneck.

Bottleneck at Source



Troubleshoot – Workspace Monitoring Integration

Centralized Monitoring Hub

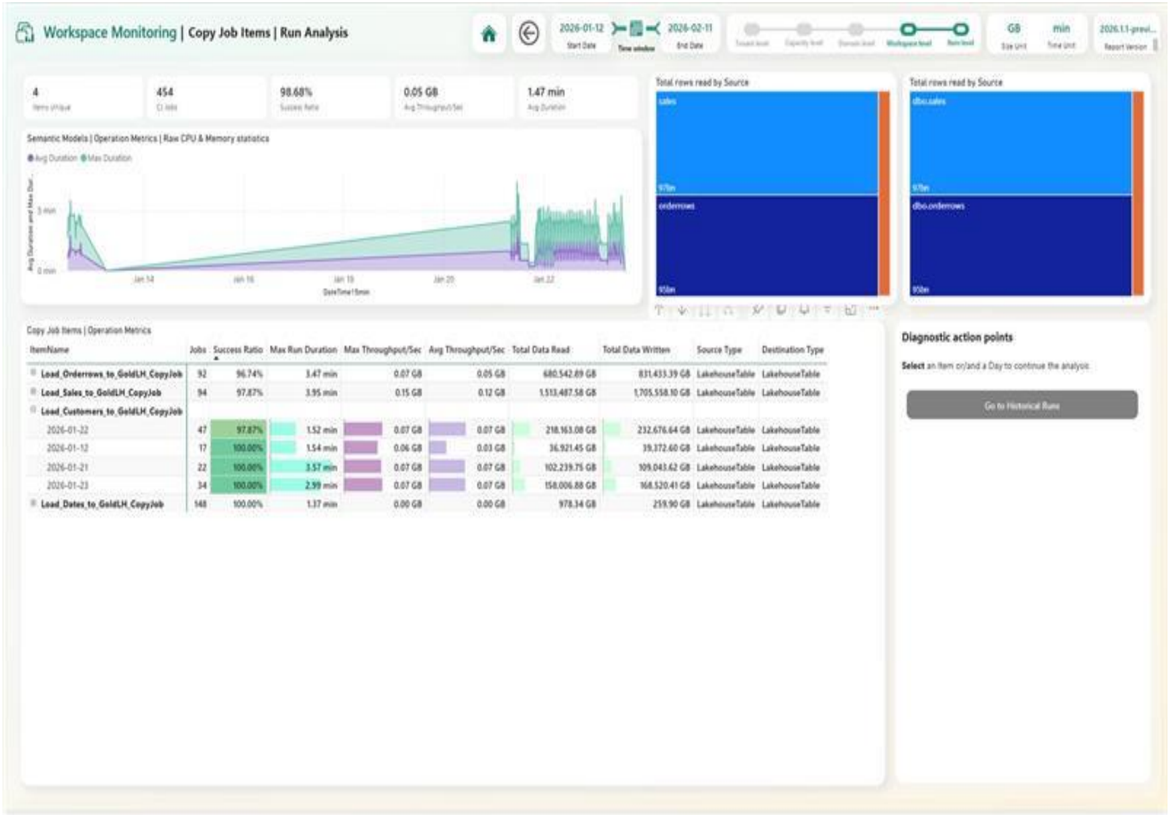
- View all Copy runs across workspaces from a single pane — no need to open the authoring surface
- Filter by status (Running, Succeeded, Failed) to quickly identify issues
- Up to 30 days of historical run data available for trend analysis and root-cause investigation

Rich Data Movement Diagnostics

- **Row-level metrics**: Rows read, rows written, files read, files written
- **Volume & throughput**: Data read/written with average throughput (KB/s) for performance benchmarking
- **Incremental copy tracking**: Load type (Full vs Incremental), lower/upper watermark bounds per table

Troubleshooting Failed Runs

- Drill into any failed run to see exact error messages and failure diagnostics
- Per-table status breakdown — pinpoint which specific table copy failed in a multi-table job
- Compare run start/stop timestamps and durations across runs to detect performance degradation



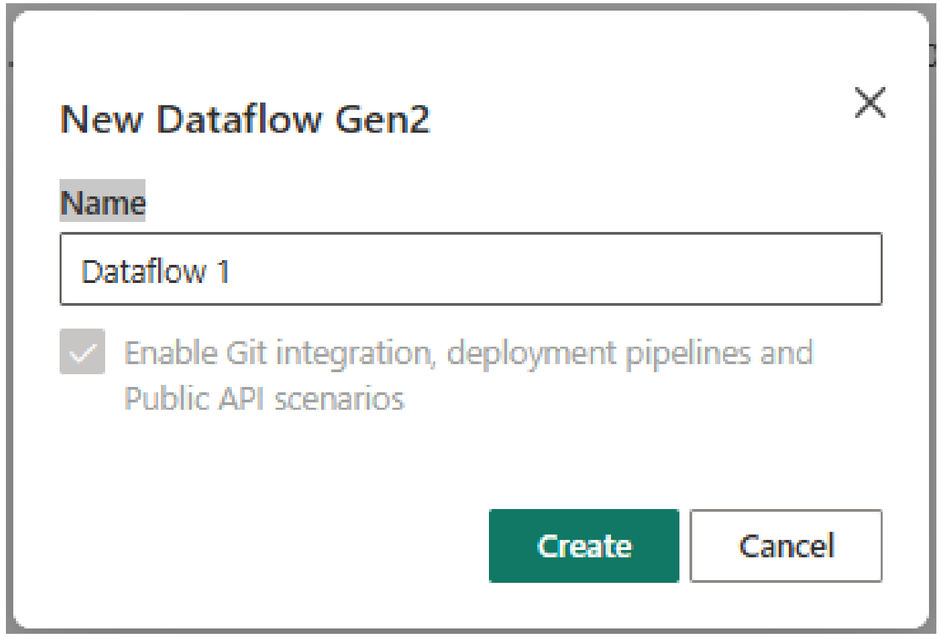
Dataflow Gen2

Dataflow Versions

- There are three versions of Dataflows:
  - Dataflows Gen1
  - Dataflows Gen2
  - Dataflows Gen2 CICD (replaces Gen2)

- Dataflows Gen2 CICD are where all the innovation is happening – especially around performance

- Upgrade to Dataflows Gen2 CICD by:
  - Right-clicking on an existing Dataflow
  - Using Pat Mahoney's accelerator in the Fabric Toolbox https://github.com/microsoft/fabric-toolbox/tree/main/accelerators/DFG2-migration-accelerator

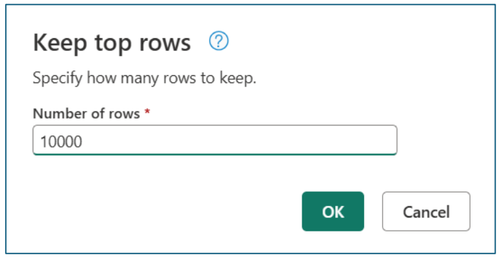


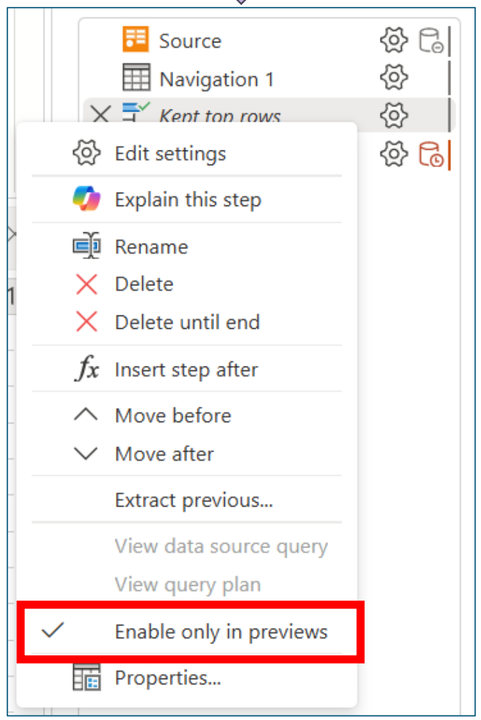
Generally Available

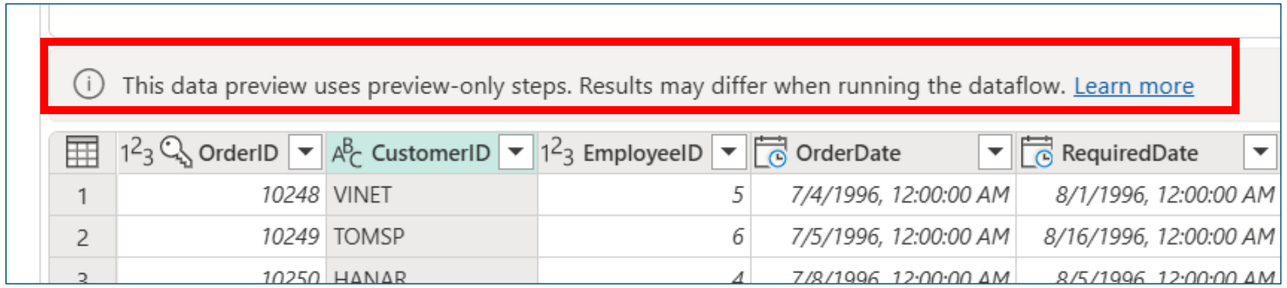
Preview-Only Steps

- New tool for improving design-time performance:
  - Step 1: Add a filter step (“classic” filter, top rows, etc.)
  - Step 2: Select “Enable only in preview” in the Step menu

- Result: preview queries run over a subset of the data, making them up to orders of magnitude faster



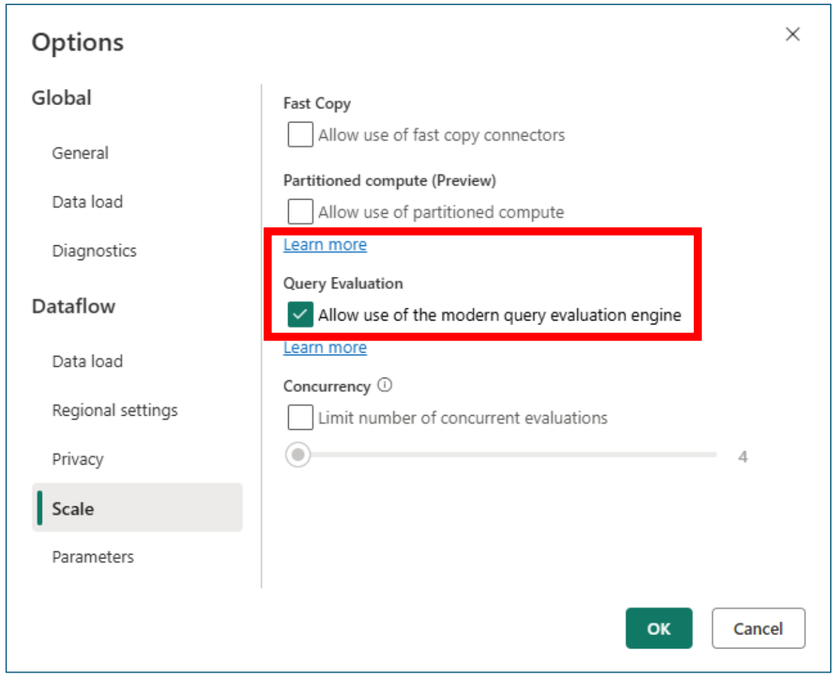




Modern Query Evaluation Engine

Generally Available

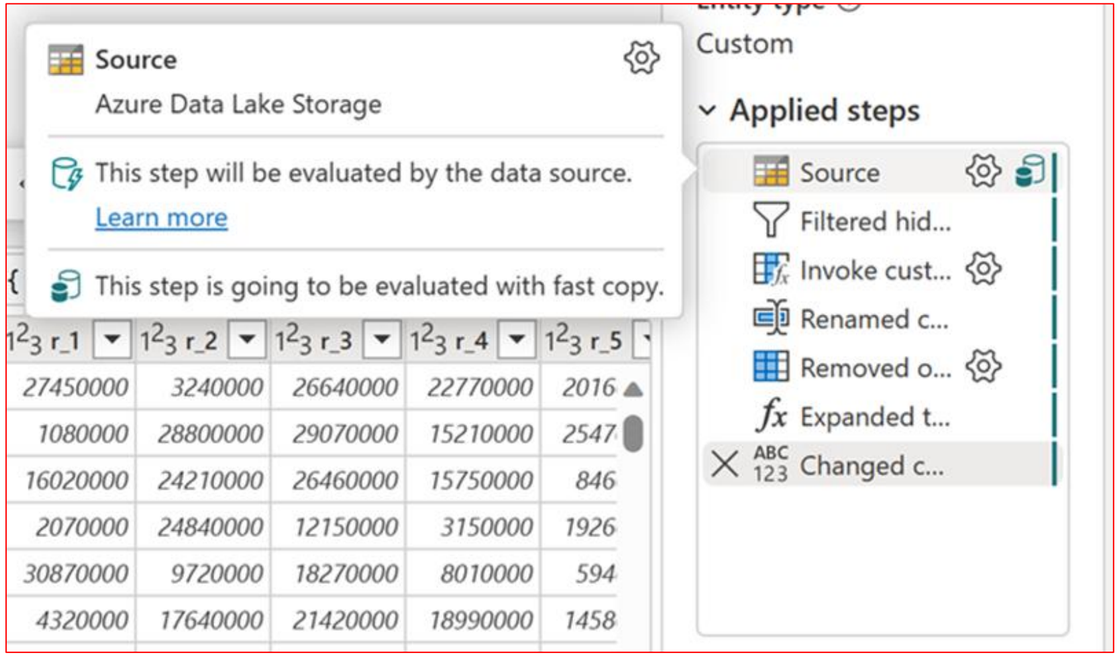
- New “M” processing infrastructure that leverages platform advancements. Significantly improves performance of type conversions and in-memory transformations
- Available only for Dataflow Gen 2 with CI/CD
- Substantially faster than previous M query evaluation engines (almost 2x faster in some scenarios). Note: improvements vary based on data source and transformations.



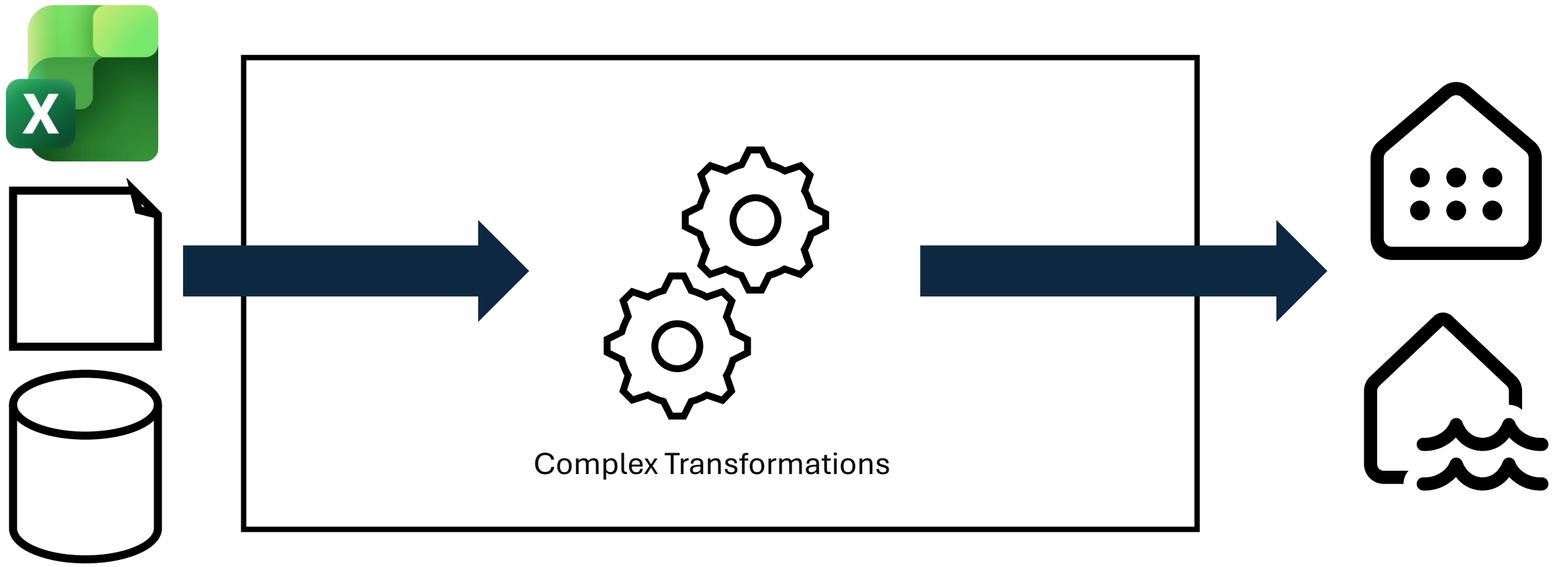
Dataset	Size	vs. Dataflow Gen1	vs. Dataflow Gen 2 (old engine)
2017 Yellow Taxi Trip Data (1 CSV file)	9.8GB	**~60% faster**	**~40% faster**
2017 – 2021 Yellow Taxi Trip Data (5 CSV files)	32.2GB	**~65% faster**	**~45% faster**

Fast Copy

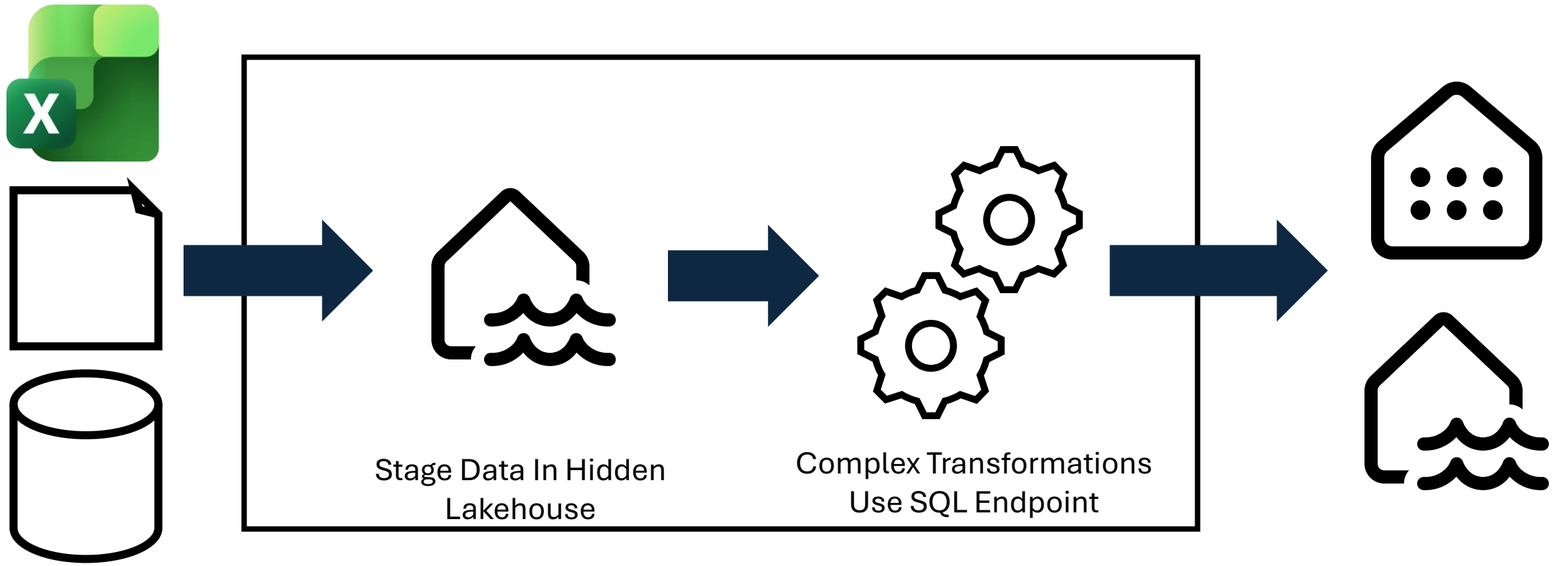
- Dataflows can sometimes use the Copy Activity engine → “Fast Copy”
- Can result in big performance gains
- Only for:
  - Data sources supported by Copy Activity
  - Limited transformations



Staging


Complex Transformations
Slow Data Sources
Dataflow
Fabric Destination

Staging


Stage Data In Hidden Lakehouse
Complex Transformations Use SQL Endpoint
Slow Data Sources
Dataflow
Fabric Destination

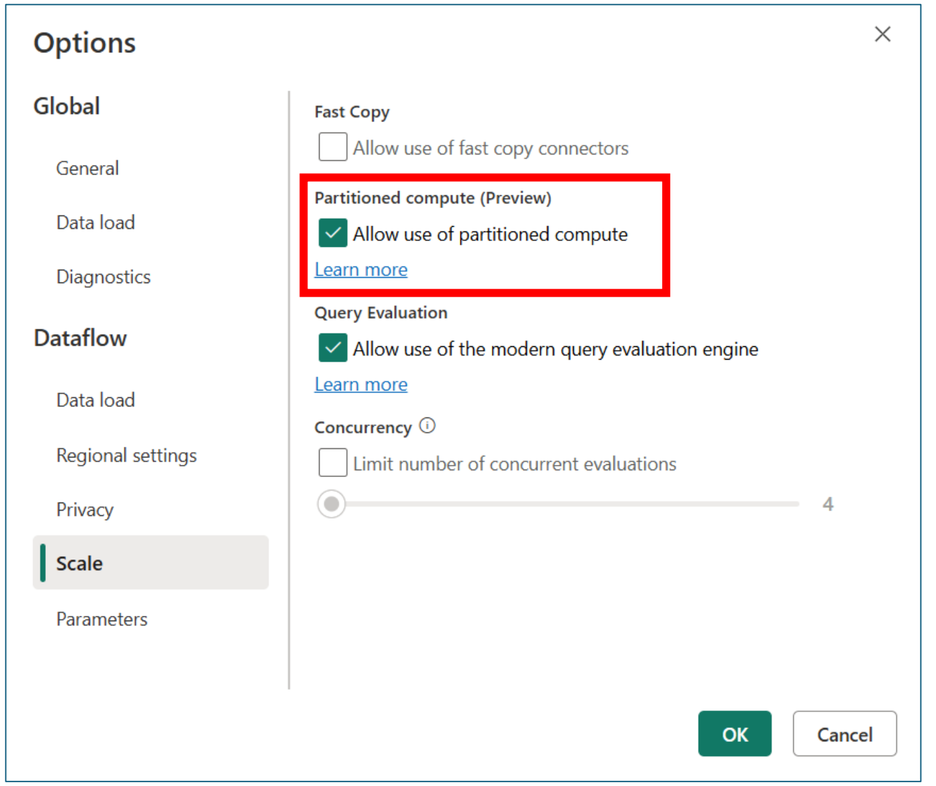
Partitioned Compute

Public Preview

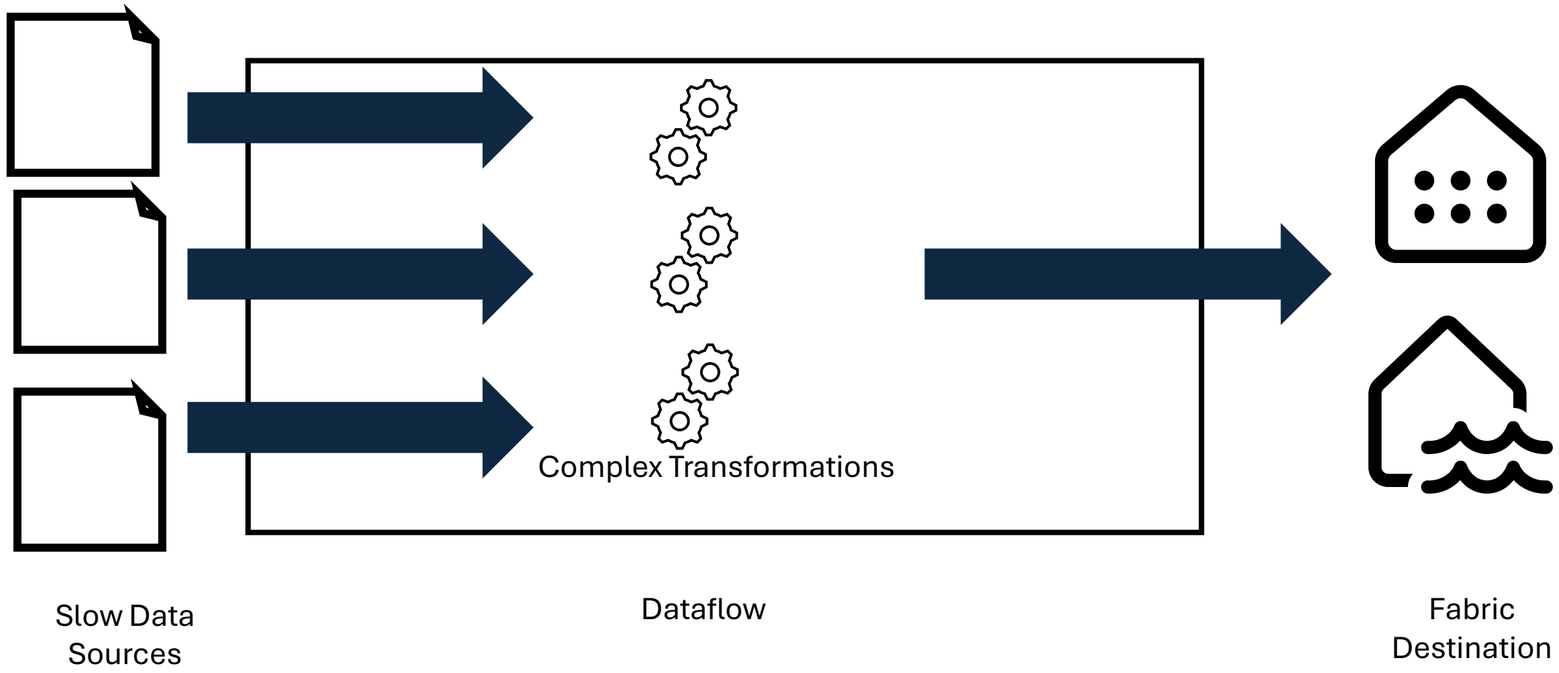
- New Partitioned Compute capability enables parallelized evaluation of the rows in a table
  - Step 1: Enable Partitioned Compute Option
  - Step 2: Use the standard Combine Files experience
- Result – generated M will now include this:

```
Table.ReplacePartitionKey(
withRelativePath, {"Relative Path"})
```

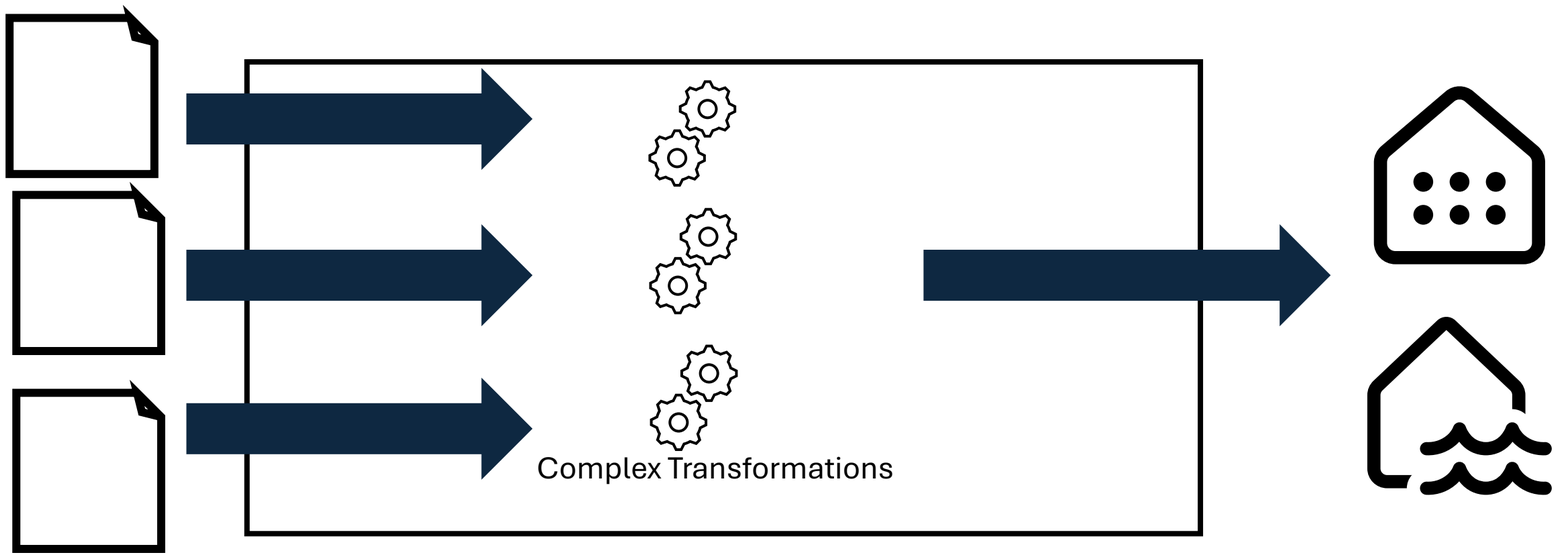
- Custom specification of partition key can be used for advanced scenarios beyond files



Partitioned Compute

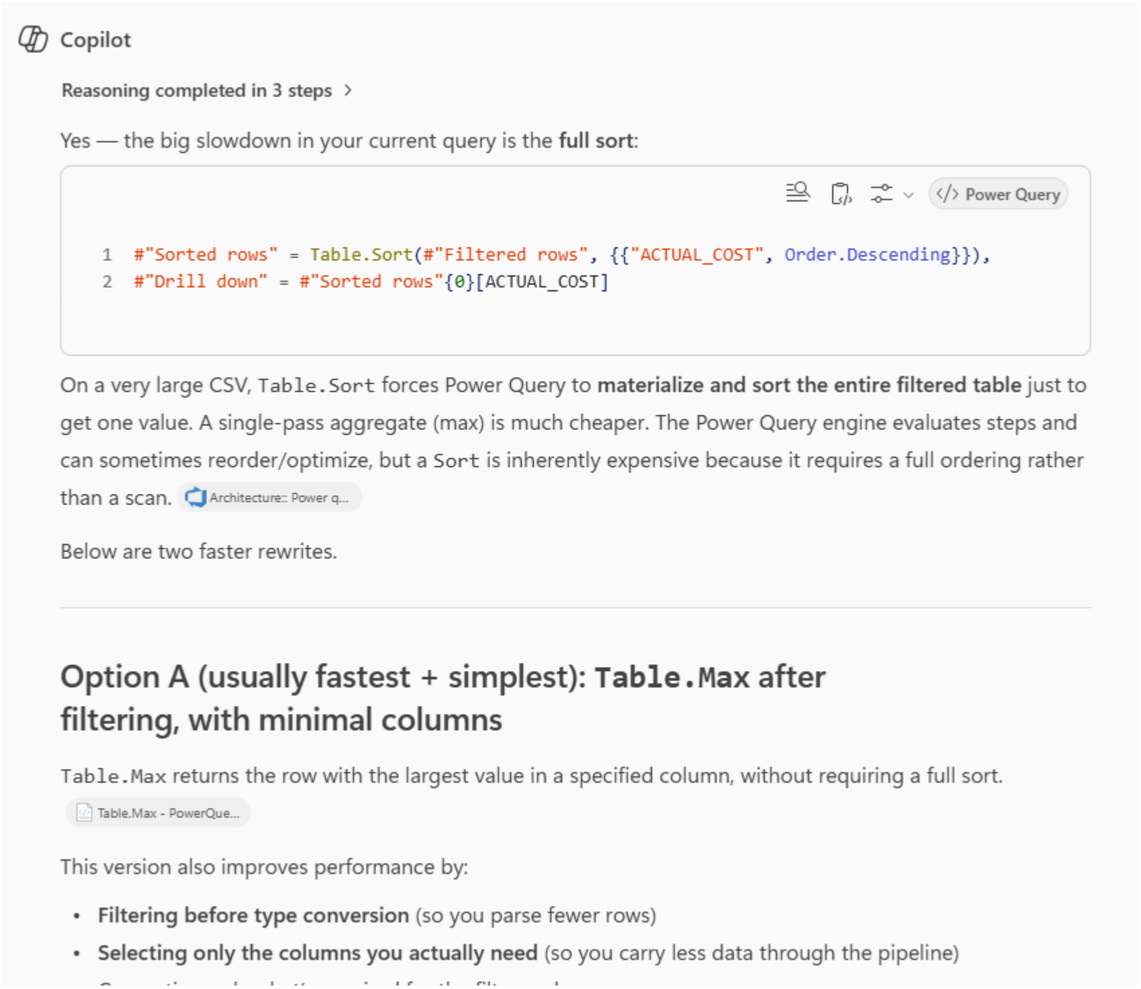


Partitioned Compute



Copilot

- Different ways of solving the same problem in Dataflows can result in different refresh performance
- Copilot inside the Dataflows editor cannot answer questions about optimizing performance
- However M365 Copilot (or your favorite AI tool) can look at the M code for your queries and suggest optimizations



Sound off.
The mic is all yours.
Influence the product roadmap.

Join the Fabric User Panel

Share your feedback directly with our Fabric product group and researchers.

https://aka.ms/JoinFabricUserPanel

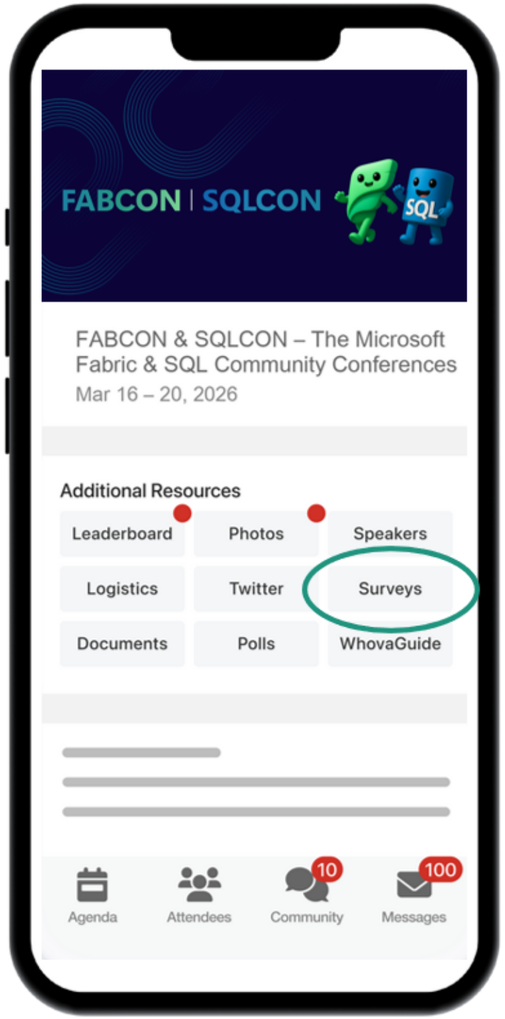
Join the SQL User Panel

Influence our SQL roadmap and ensure it meets your real-life needs

https://aka.ms/JoinSQLUserPanel

How was the session?

Complete Session Surveys in Whova for your chance to WIN PRIZES!



Get Two Fabric Certifications for FREE

Attendees of FABCON can take the Fabric Analytics Engineer or Fabric Data Engineer exam for free. Be part of the 2 fastest growing role-based certifications in Microsoft history.

Request your voucher by March 23, 2026.

https://aka.ms/fabcon/cert100