At Ignite 2024, Microsoft announced the public preview of the SQL Server(?) database within the Microsoft Fabric ecosystem. This adds another tool to the ever-growing toolbox that is Fabric.

As usual, the release demo is based on small datasets and that's not the reality I'm dealing with. Customers have large amounts of data, we as consultants create strange procedures to clean and model the data and in the end the customer expects amazing performance for as little money as possible.

In this session, I'll take you along my tests to load and model the data into a star schema. The session will try and dig as deep as possible into performance, storage and costs of this database. Depending on the development of the Fabric portal, monitoring will be a part of this as well.

In the end you'll have a better understanding of what the Fabric SQL Database has to offer and if it could be a fit for you use case.

Fabric SQL Database

CAN I HAVE SOME MORE DATABASES PLEASE?

Thank you to our Fabric February Friends!

twoday

bouvet

sopra steria

DATAMASTERMINDS

ADVANCING ANALYTICS

Evidi

Profisee
Master Data Management

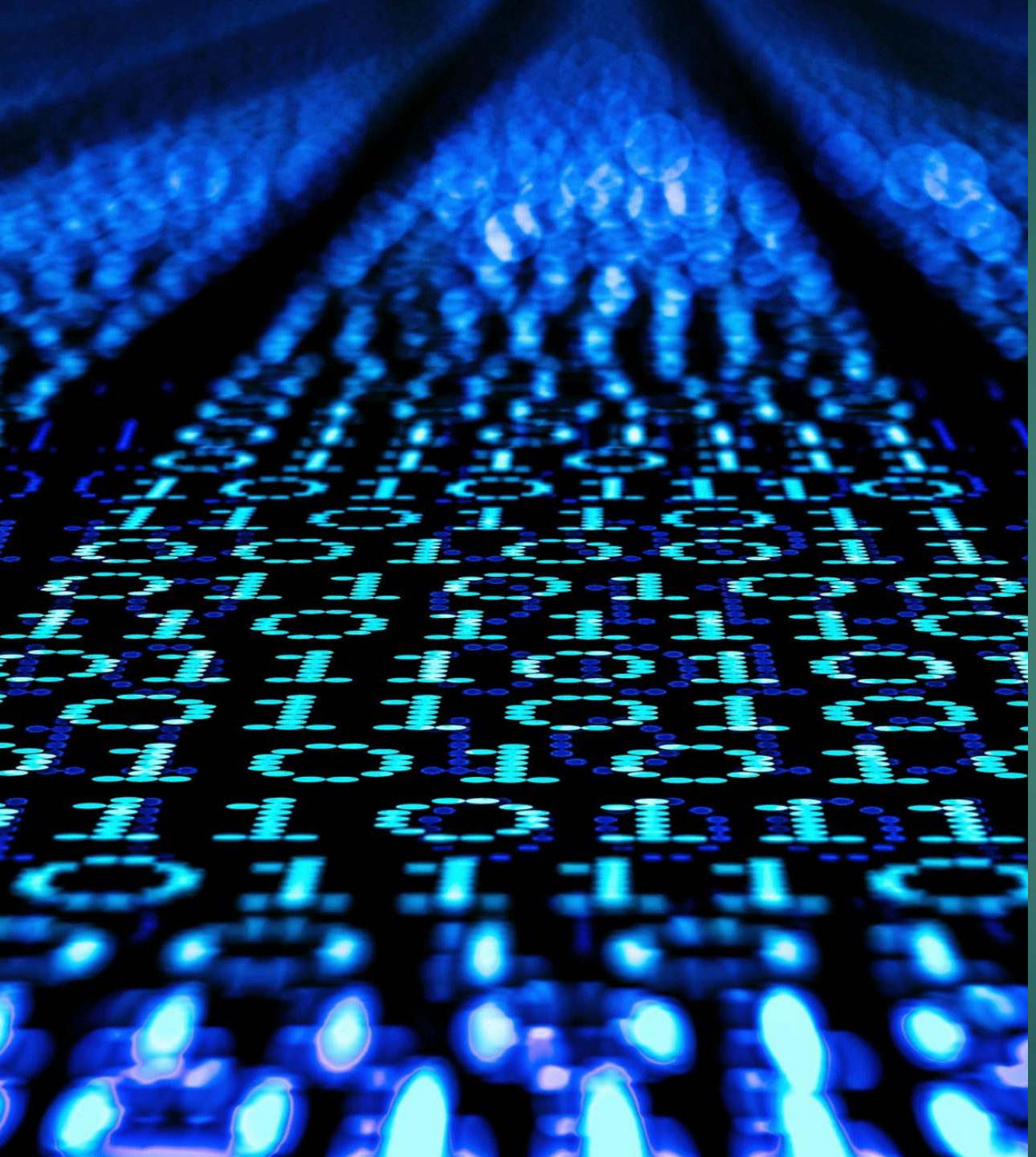
Tabular Editor

KURANT

Fraktal

CluedIn

Dufrain
THE DATA COMPANY



Concept

- It's an Azure SQL database, inside of Fabric
- It should be able to do everything you expect from an Azure SQL Database
- No monitoring outside of Fabric

Concept

- Serverless architecture
- Resource governor to control available CPU and Memory
- Even less configuration options than the Azure Sql Db

SQL DB versus Warehouse

Full syntax support vs. subset supported

Stores data in SQL format vs delta parquet storage

Mirroring in the background to Onelake

Metrics - Microsoft Azure | Fabric

https://app.fabric.microsoft.com/home?experience=fabric-developer

Apply | MVP Comm... | Application Status |... | MCT Lounge - Micr... | Microsoft Certified... | Microsoft MVP Awa... | Microsoft MVP Awa... | Exam DP-600: Imple... | AzureMCTHelper/R... | Other favorites

Fabric Home

Search

Trial: 43 days left

14

Home
Workspaces
OneLake
Monitor
Real-Time
Workloads
FabricSqlDb
Sql DB TPCH
Fabric

Welcome to Fabric

Create a workspace with a predesigned template called a task flow. Task flows keep your items organized. Learn more

New workspace | General | Basic data analytics | Data analytics using a SQL ... | Medallion | Event analytics | Lambda | Sensitive data insights | Basic machine learning models

Learn more about Fabric

What is Microsoft Fabric?
Watch this 1-minute introductory video

Ingest data into Fabric
Complete an end-to-end tutorial for Data Factory

Build a lakehouse
Complete an end-to-end tutorial for Lakehouse

Build a warehouse
Complete an end-to-end tutorial in Data Warehouse

Build a machine learnir
Complete an end-to-end tu

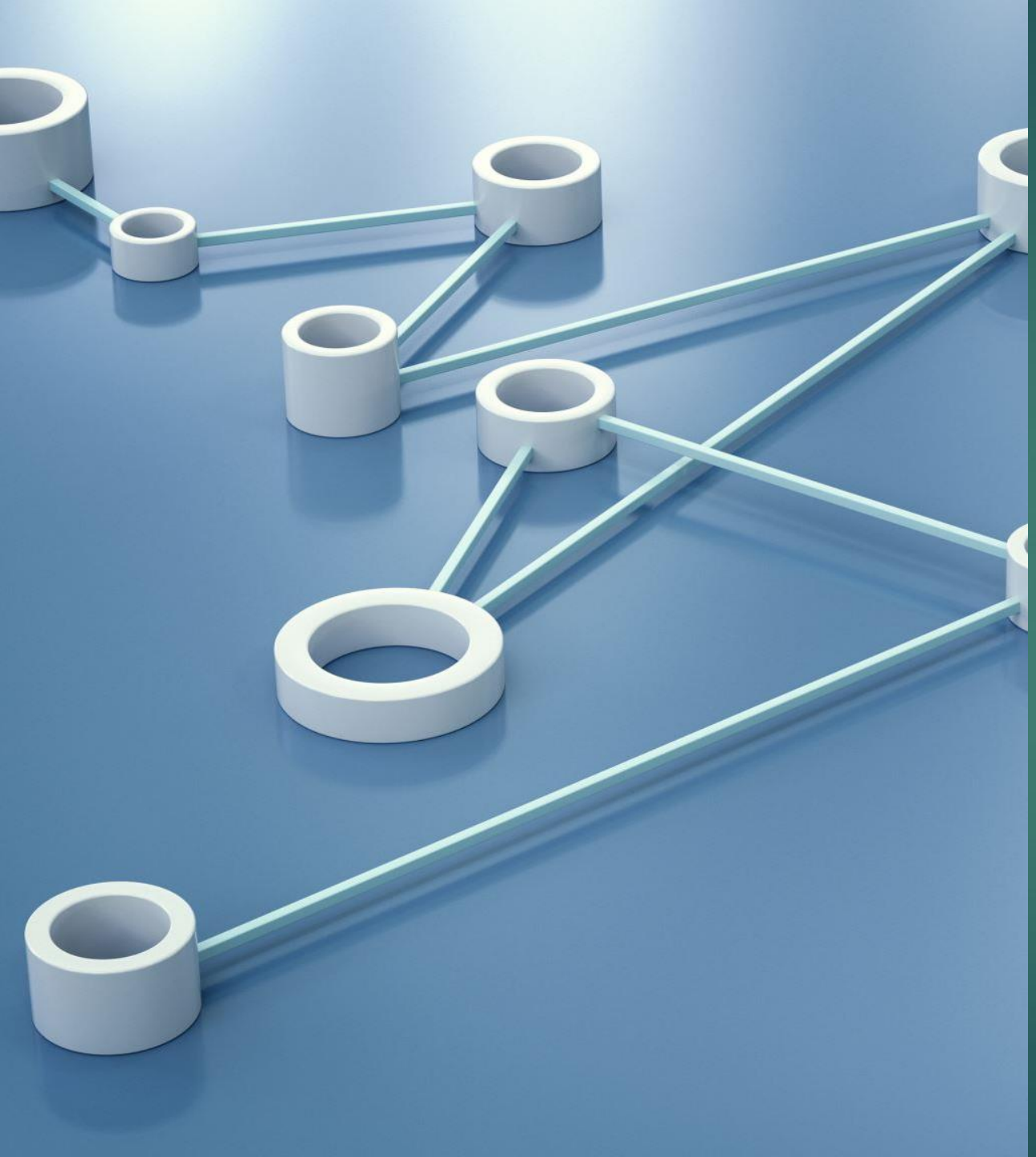
Show less

Quick access

Recent workspaces | Recent items | Favorites

Filter by keyword | Filter

Name	Opened
FabricSqlDb	now
TrialF64	20 hours ago



Deployment and structure

- Single database, provisioned within seconds
- Performance partially depends on your F SKU
- Performance controlled by Resource Governor
  - It can change with changes in the load

Use cases

- OLTP databases
- Meta data storage for ELT/ETL, master data management
- Data warehousing

Reitse Eskens

Engineer | Architect

Axians Business Analytics

Reitse.Eskens@axians.com

/in/reitseeskens

https://sqlreitse.com

@2meterDBA

Microsoft® Most Valuable Professional

Microsoft CERTIFIED TRAINER MCT

axians

Can I break it?

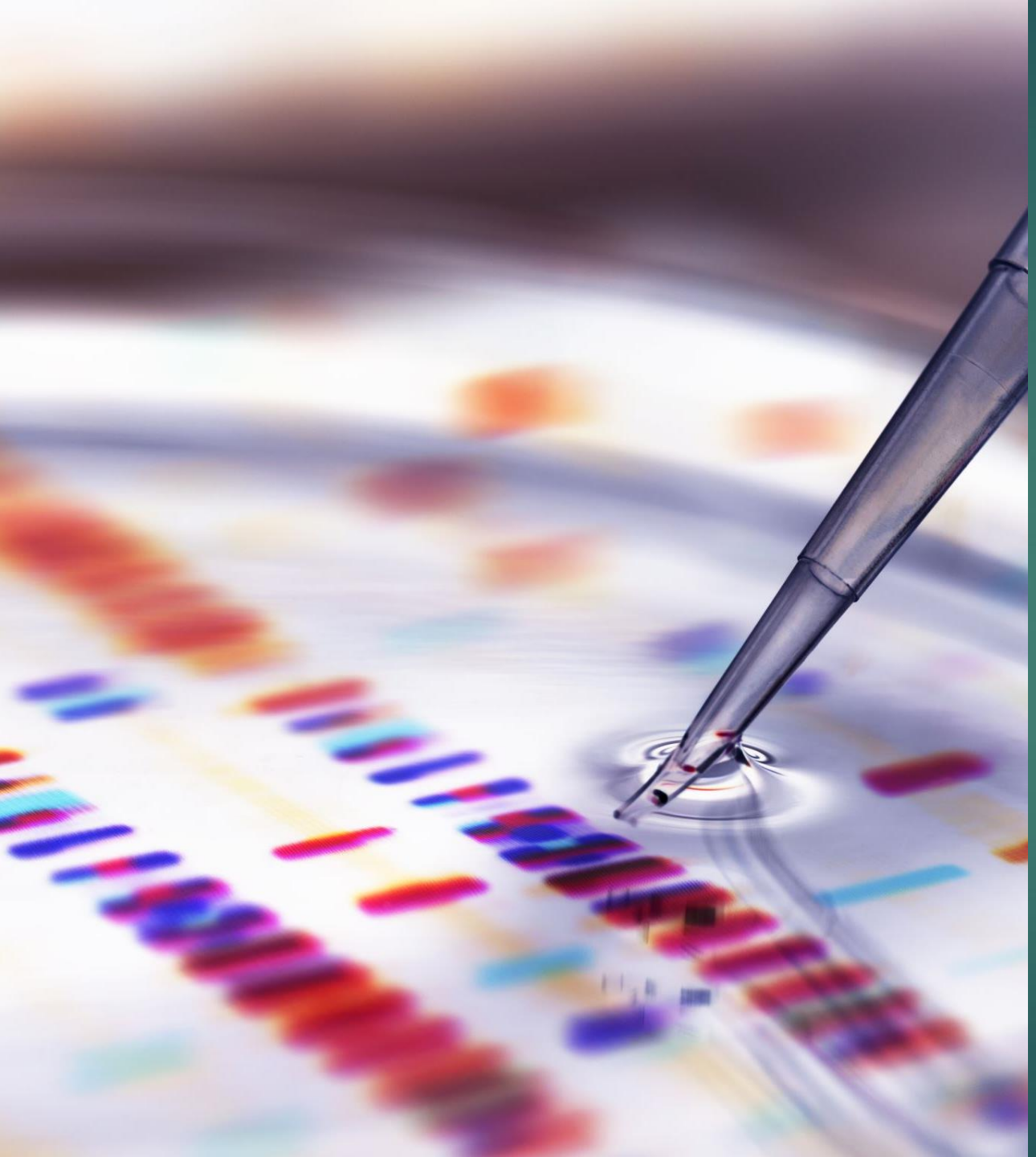
Maybe?

Disclaimer

NOT USED IN PRODUCTION YET

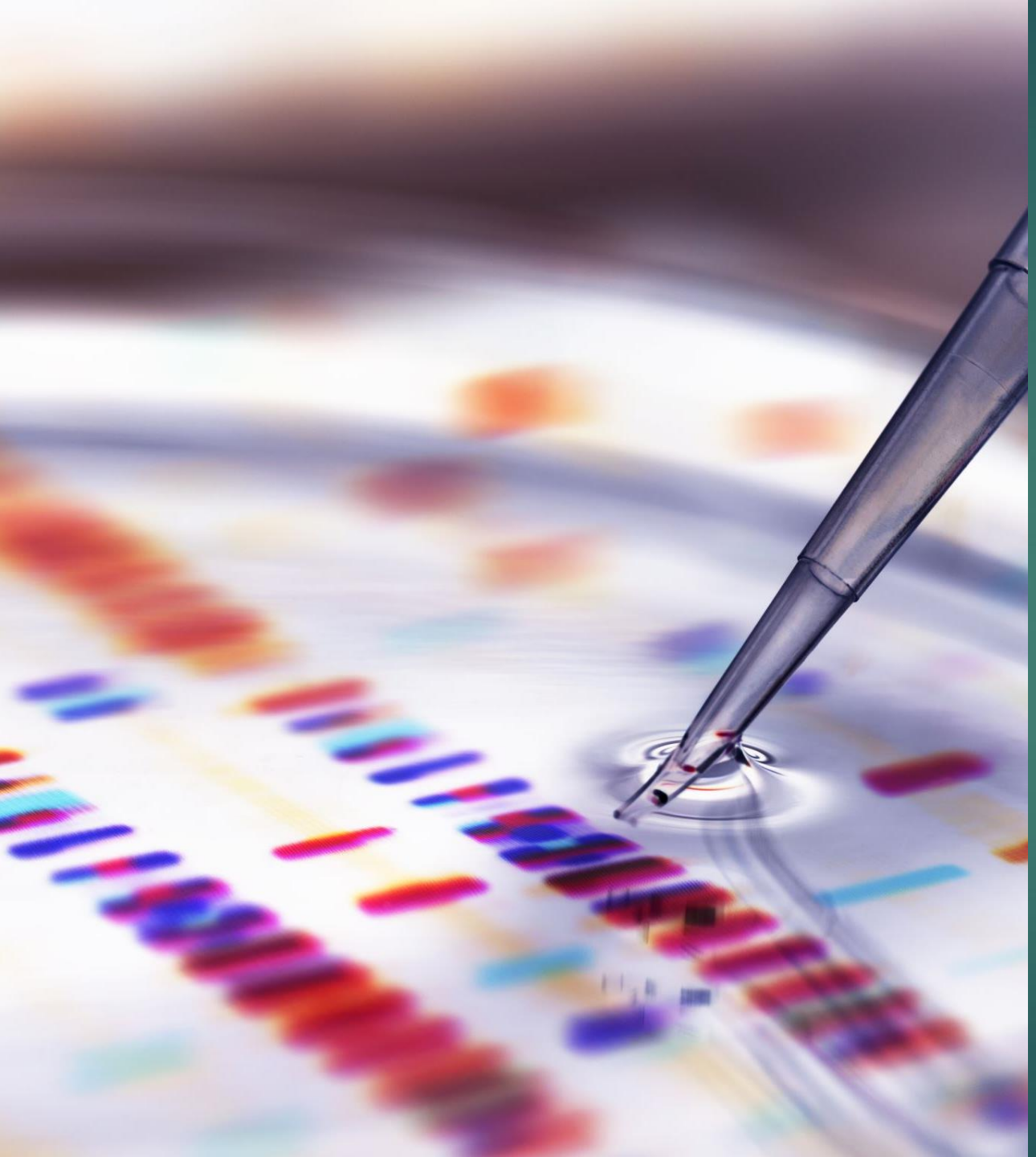
TESTING DONE ON TRIAL CAPACITY AND F16

IN PREVIEW



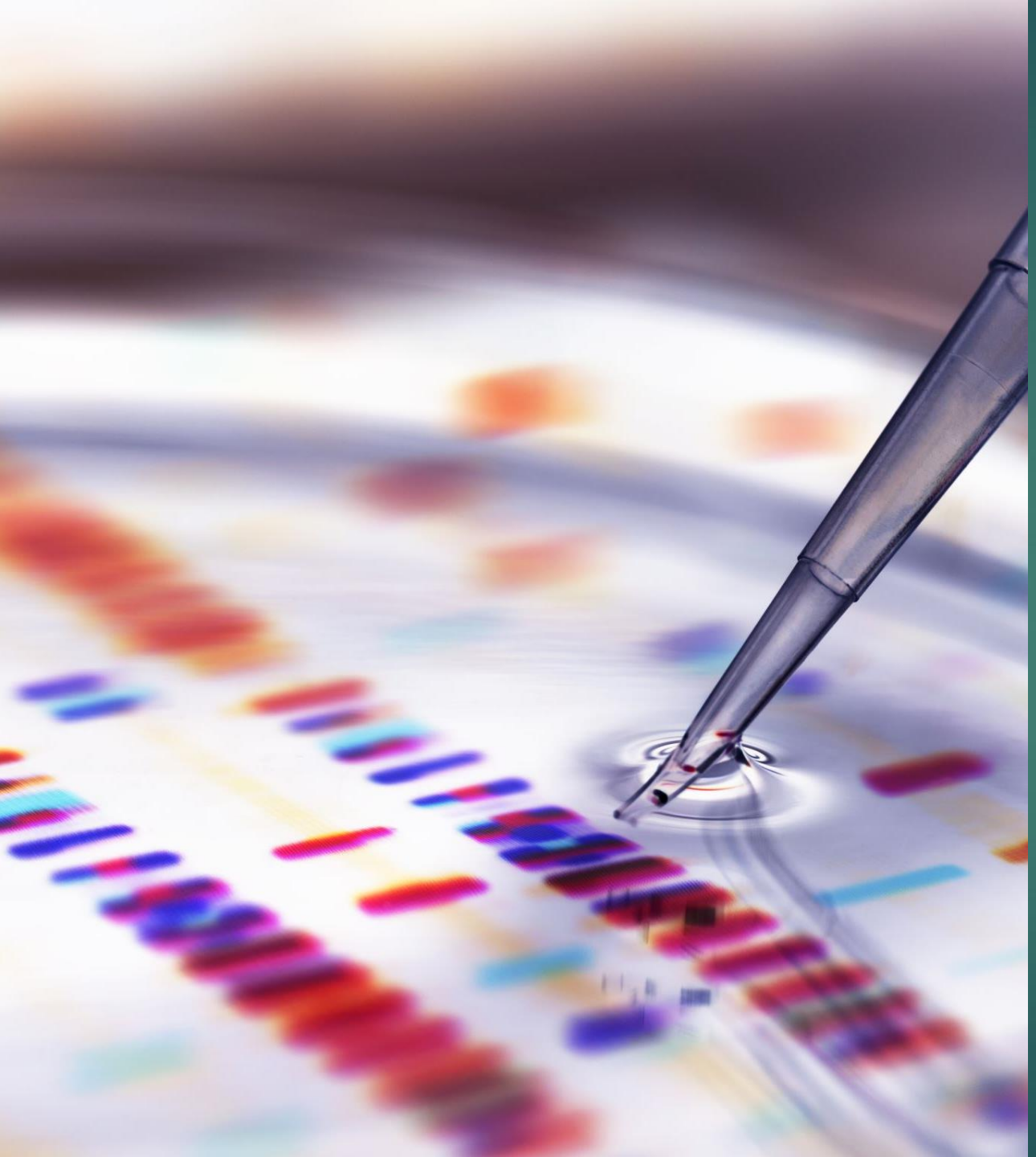
Performance

- Ingestion testing
- Stored procedure testing
- Query testing



Ingestion test

- Using a pipeline
- Using PowerShell ingesting CSV



Ingestion test

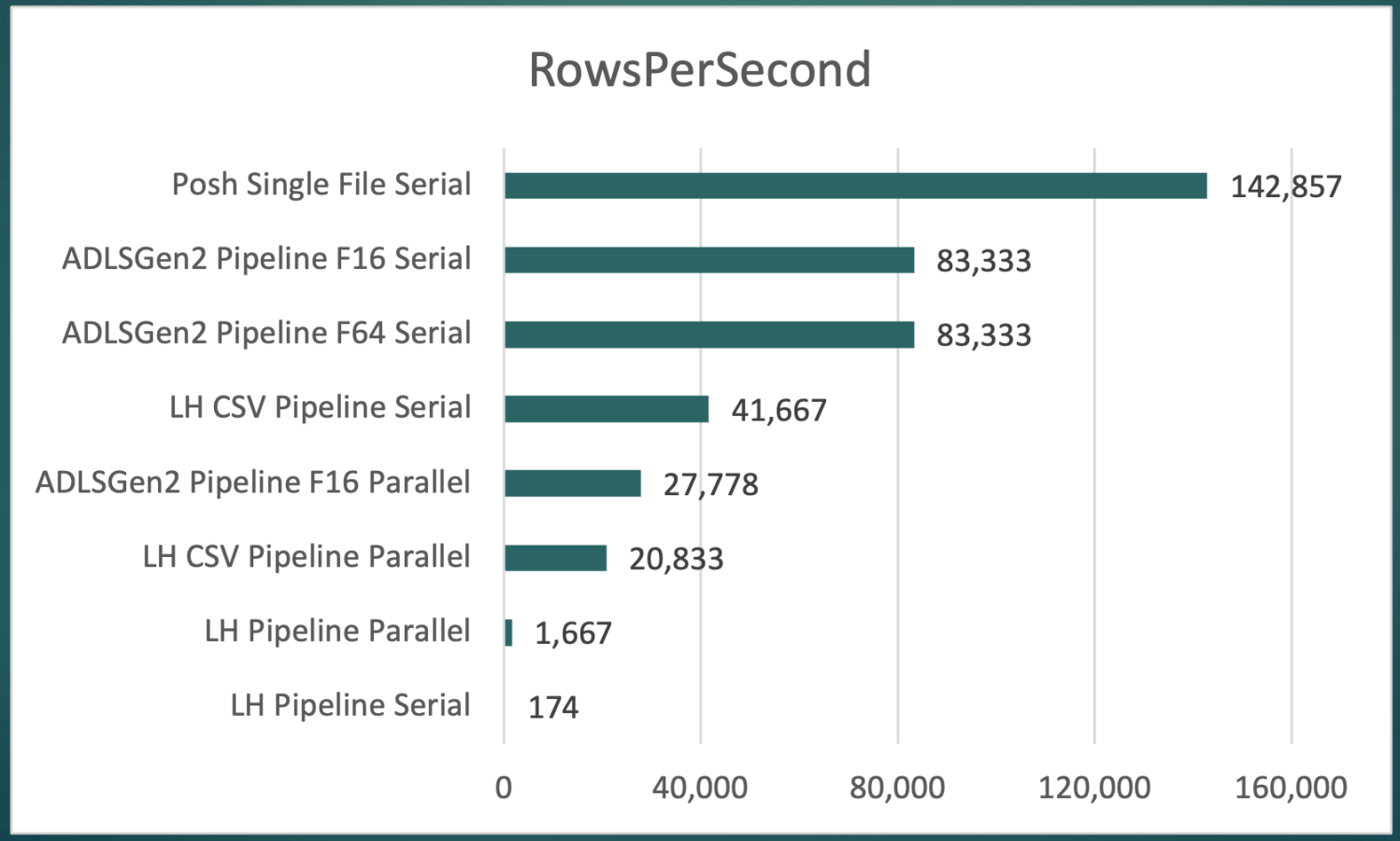
- Reading from Lakehouse Tables
- Reading from Azure Storage account
- Reading from Fabric Lakehouse Files
- Reading from local laptop (SSD)

Performance: The dataset

Table Name	Rows	Data read	Data Written	Files read
Regions	5	394 Bytes	738 Bytes	1
Nations	25	2 KB	4KB	1
Supplier	5,000,000	723 MB	1,2 GB	5
Customer	60,000,000	10 GB	17 GB	4
Part	100,000,000	12,4 GB	20,8 GB	5
PartSupp	400,000,000	62 GB	110 GB	5
Order	750,000,000	91 GB	132 GB	5
LineItem	3,600,066,144	488 GB	581 GB	6

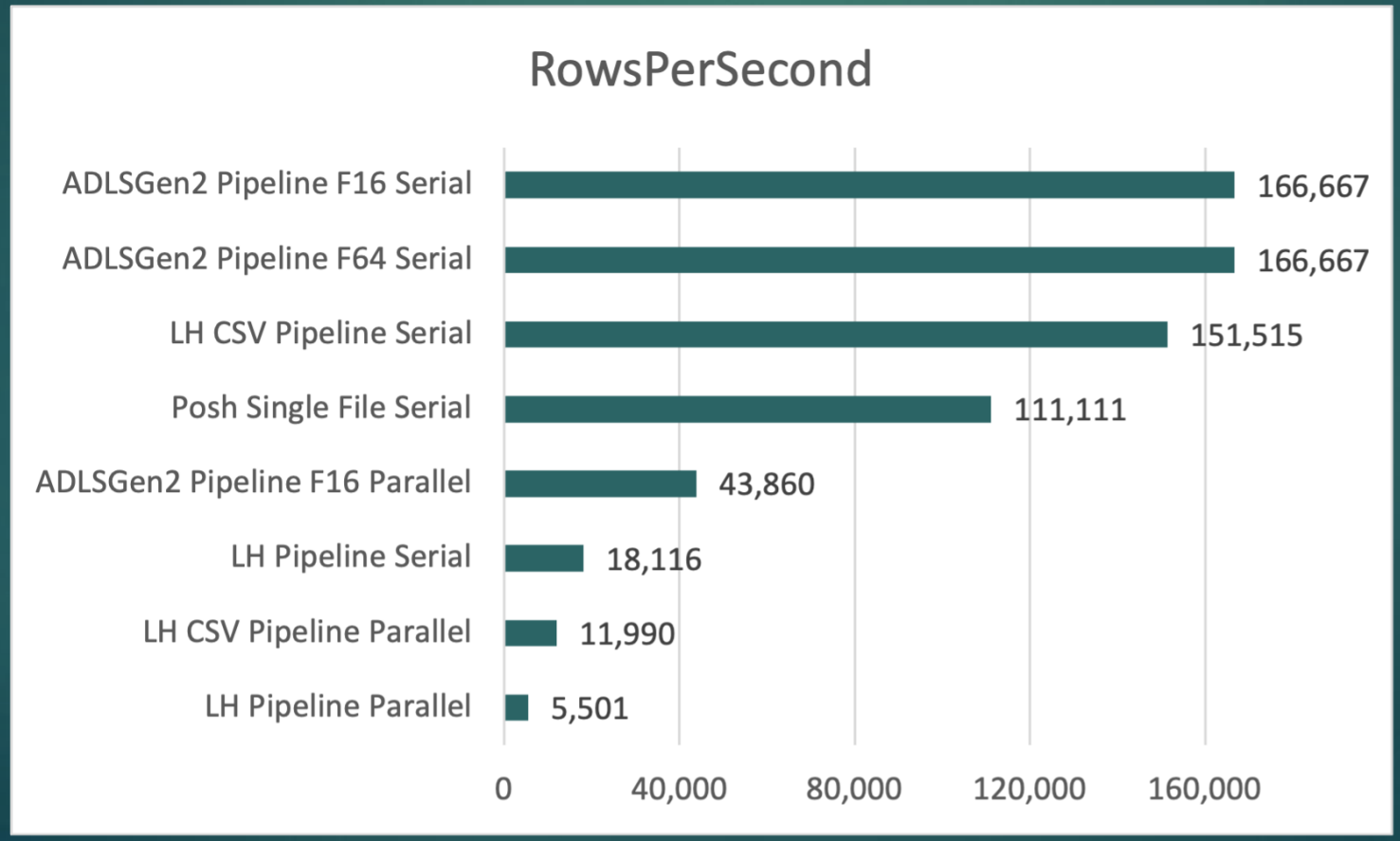
Performance Summary: Ingesting small files (Supplier)

- 5,000,000 rows, 723 MB



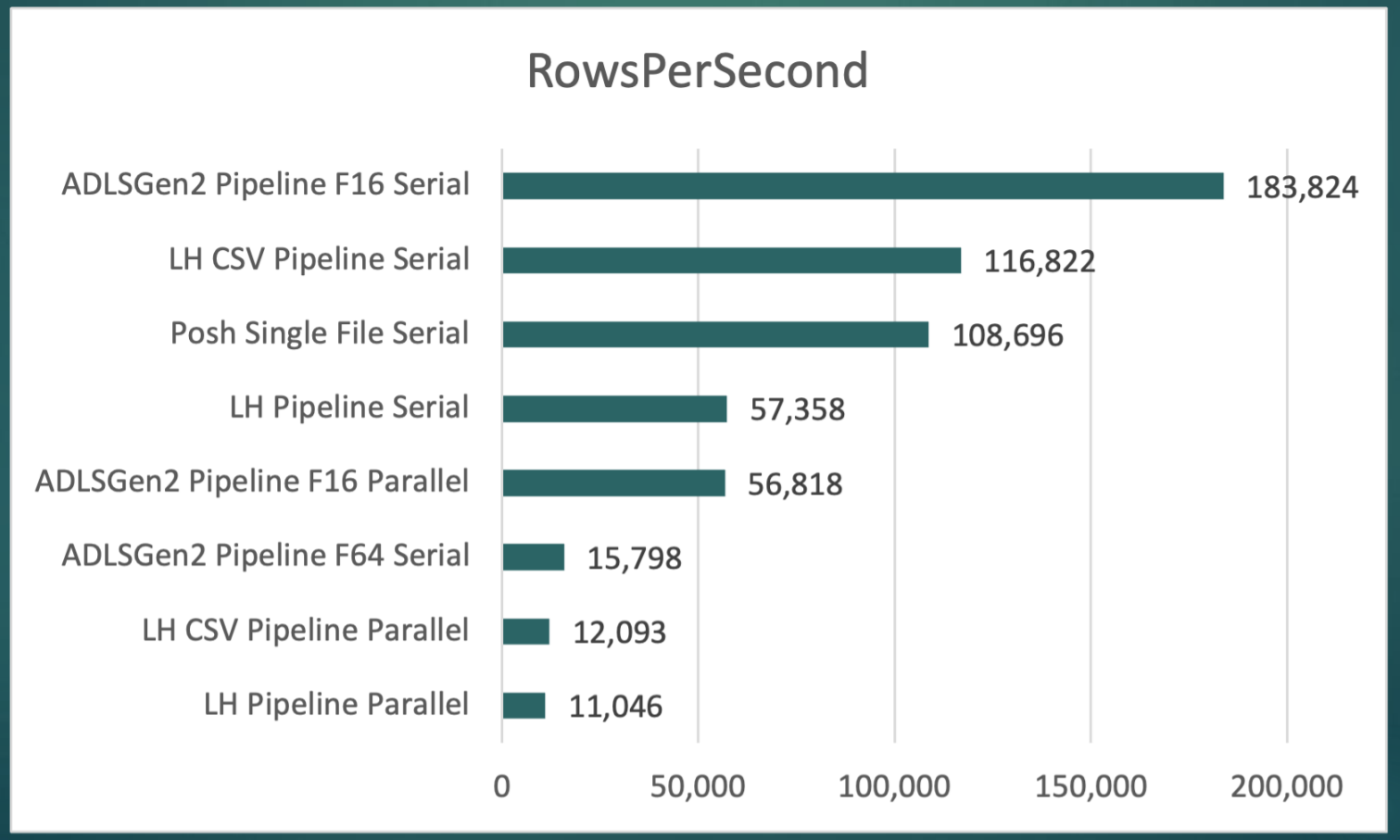
Performance Summary: Ingesting medium files (Parts)

- 100,000,000 rows, 12,4 GB



Performance Summary: Ingesting the big files (Orders)

- 750,000,000 rows, 91 GB



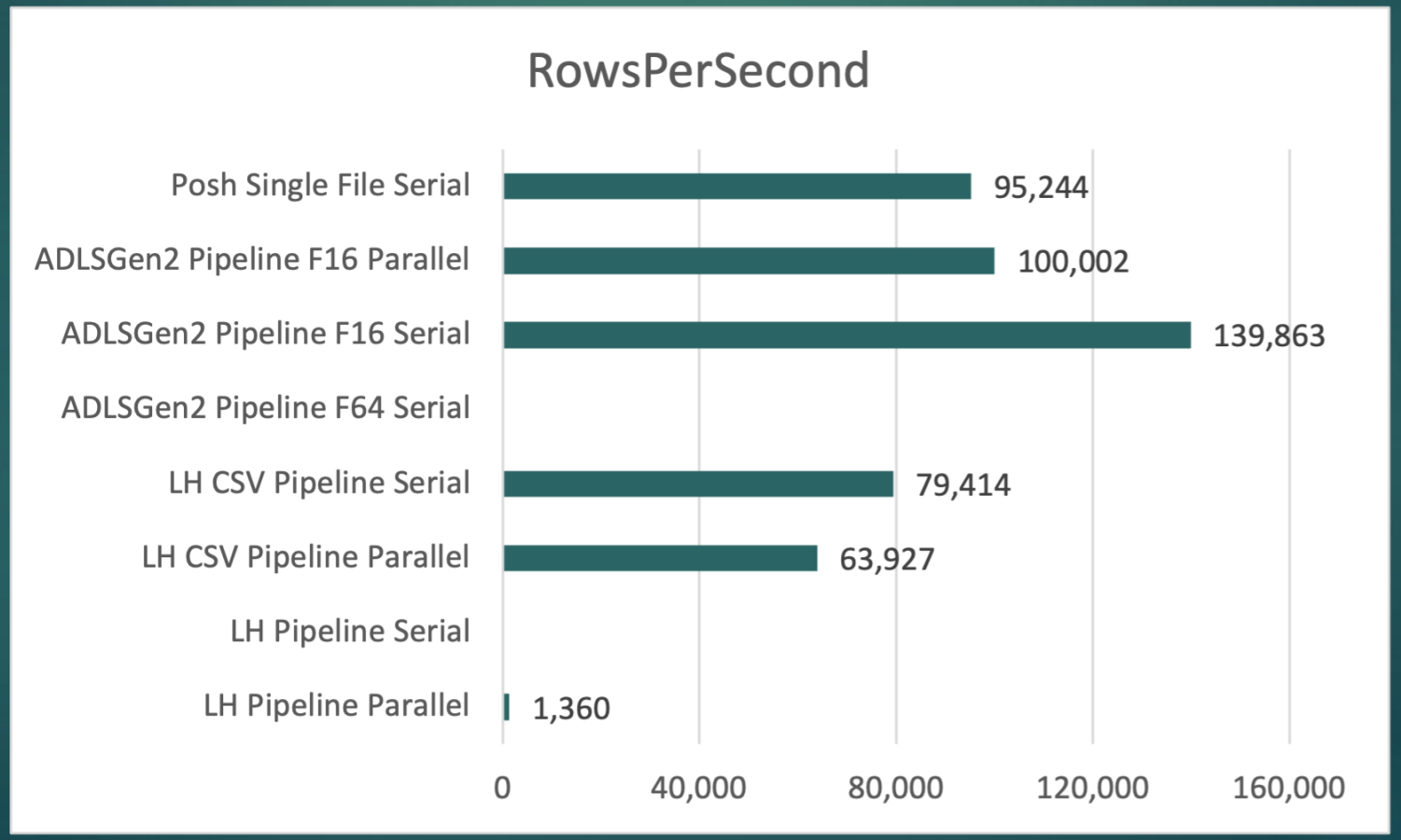
But, not all of these results are equal

NOT EVERY FLOW FINISHED (ON TIME)

And…

Performance Summary: Ingesting the big files (Line Items)

- 3,600,066,144 rows, 488 GB



What happened?

FLOWS DIDN'T START, RAN OUT OF TIME (12 HOURS) OR THE CAPACITY RAN OUT OF STEAM

Summary

READING FROM LAKEHOUSE INTO FABRIC SQL IS SLOWER THAN READING FROM CSV FILES IN A STORAGE ACCOUNT

CHECK YOUR TIME LIMIT ON YOUR PIPELINES

SERIAL PROCESSES CAN RUN FASTER THAN PARALLEL PROCESSES

Comparing with Azure SQL

Azure SQL Setup, west europe

AZURE SERVERLESS DATABASE

12 CORES, 2 MINUMUM

750 GB STORAGE

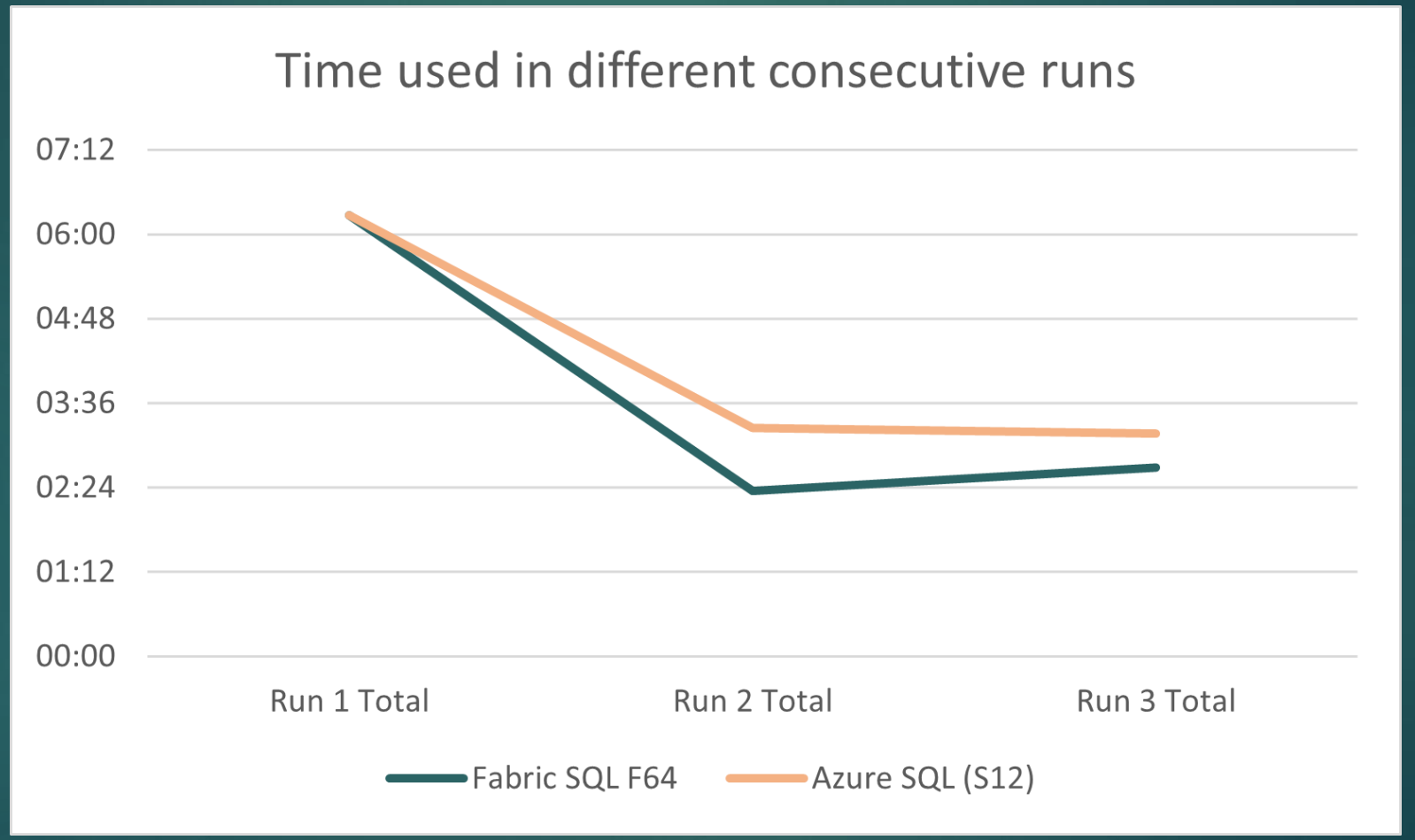
Script setup

USING POWERSHELL TO INGEST DATA

DBATOOLS.IO COMMANDLETS

THREE RUNS

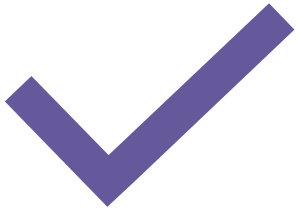
Comparing with Azure SQL



Comparing with Azure SQL

Summary

First run is slow, the storage files (MDF) need to be grown out



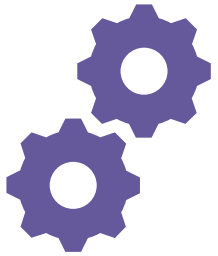
Second and third run are comparable

Azure SQL a little slower, but cheaper (YMMV)

Let’s create a model!

Code setup

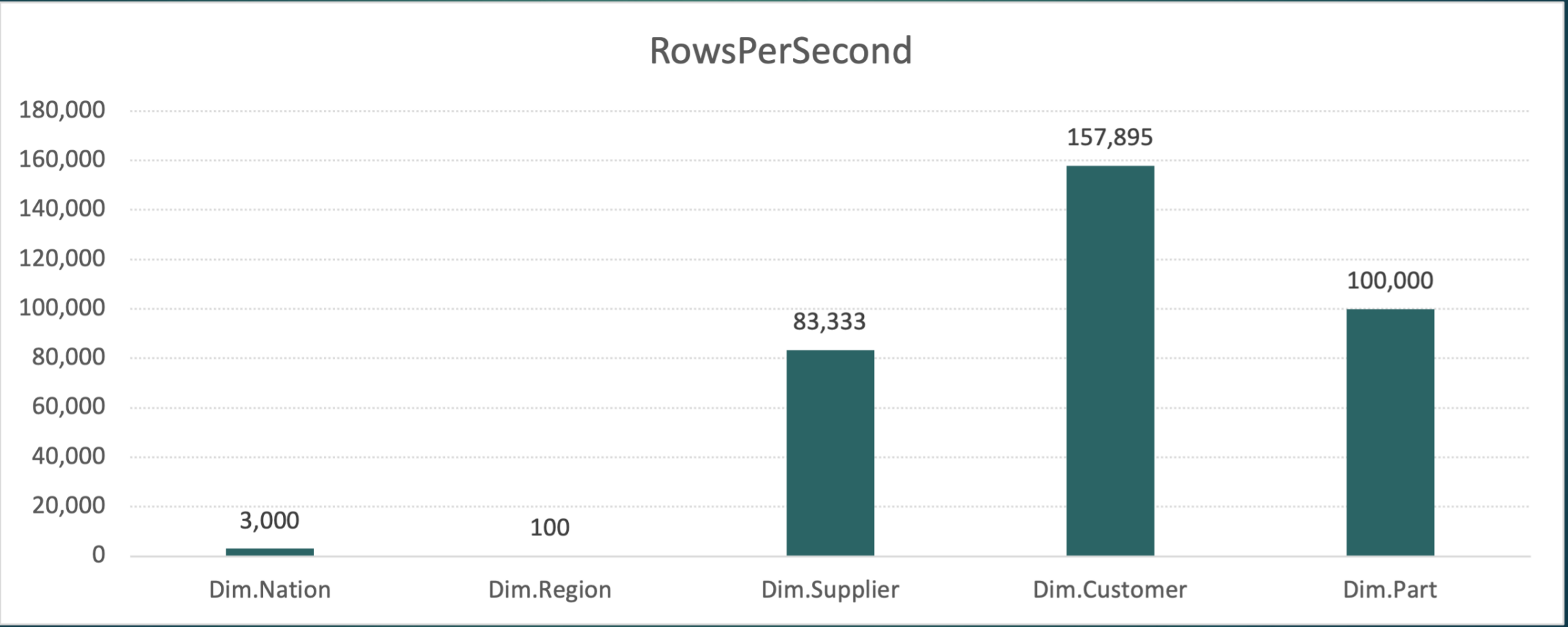
Stored procedures filling tables



Using MERGE to detect changes in data

Results from second run

Performance Summary: using procedures to populate the dimensional model

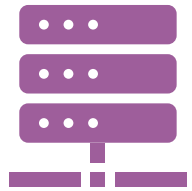


Summary

Time-out after 15 minutes

No setting found in sys.databases or sys.database_scoped_configurations

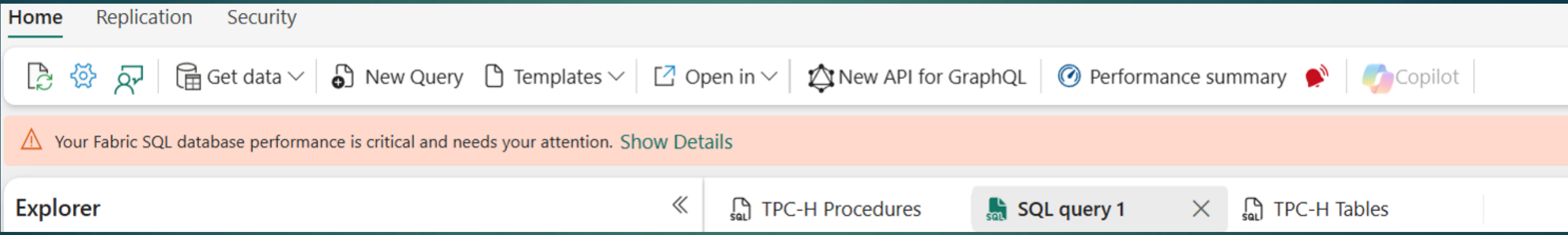
Speed really depends on your query techniques, investment in tuning and data size



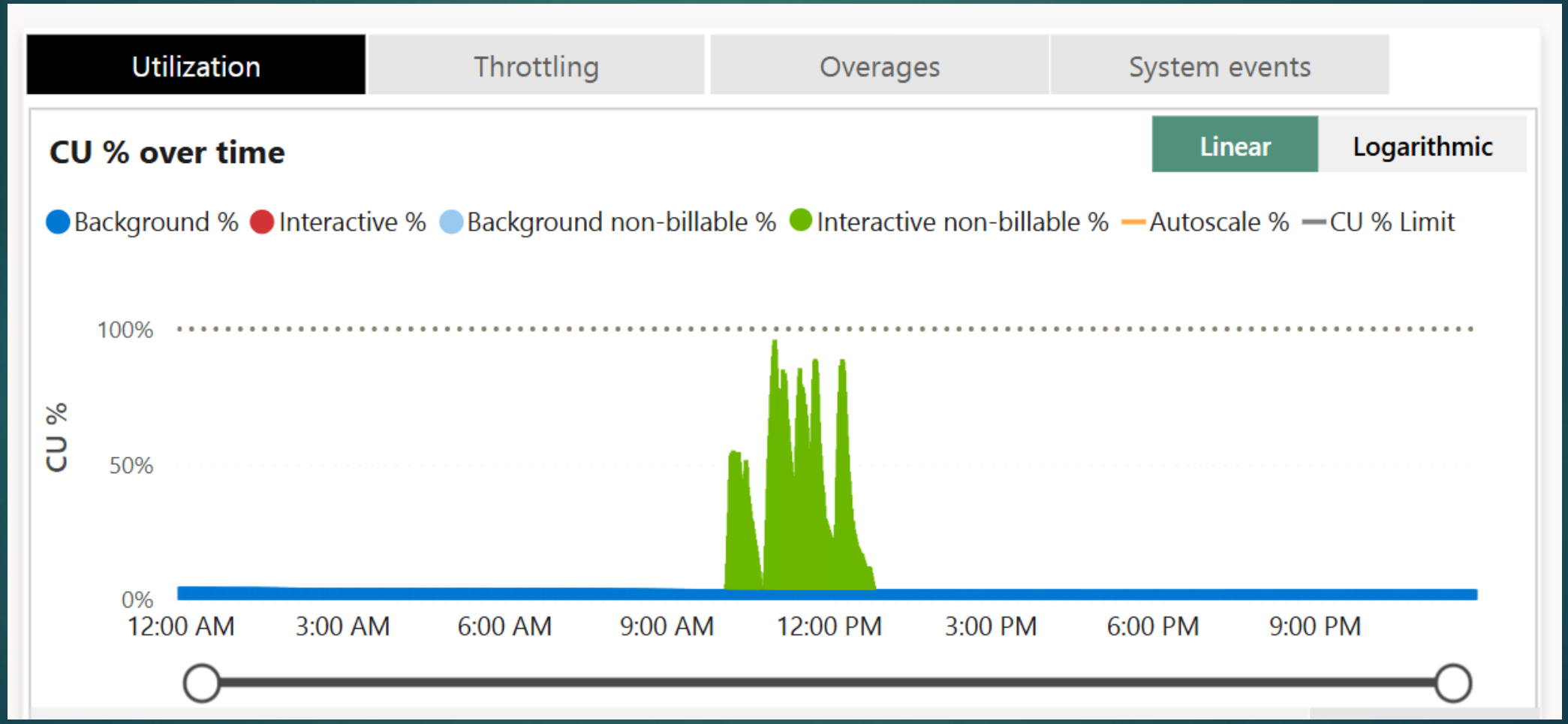
Your data model is still very important!

Monitoring

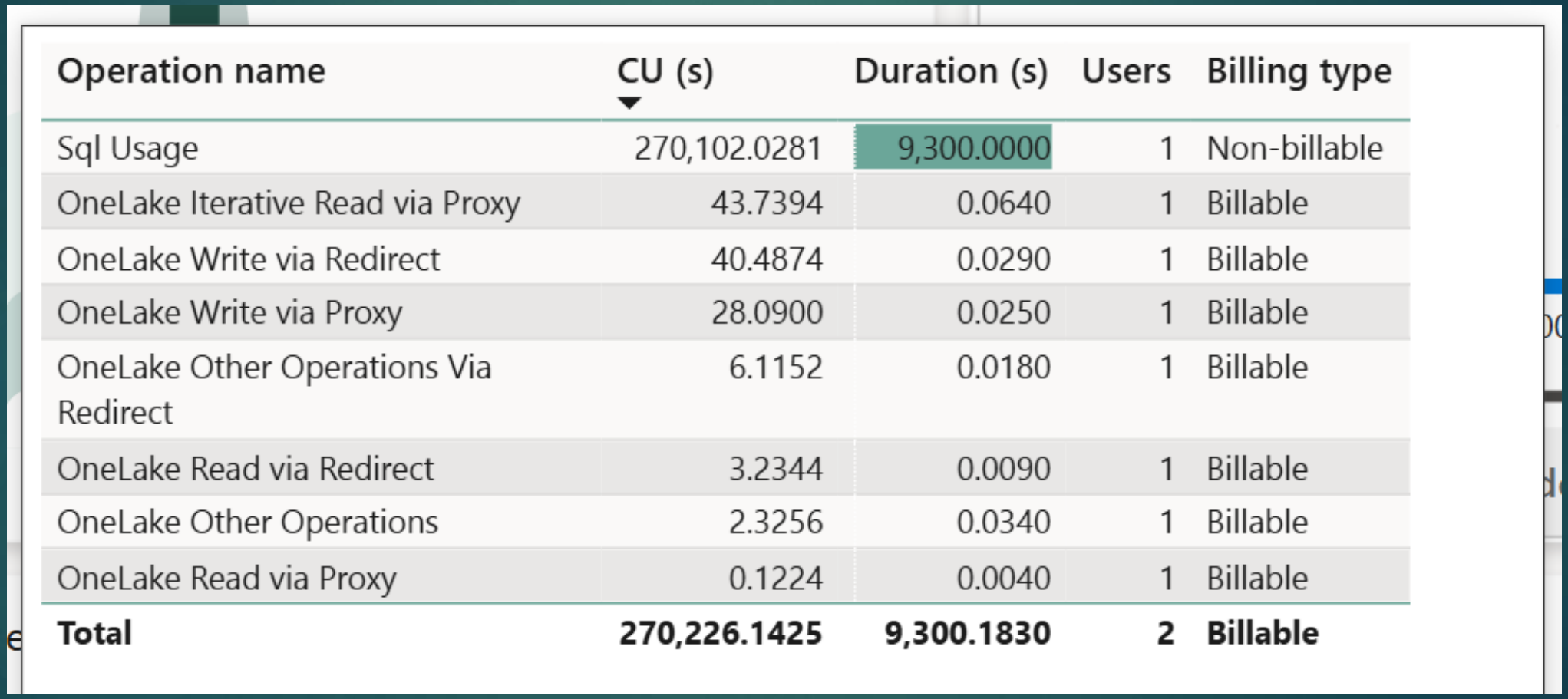
Built in monitoring: alerts



Capacity Metrics App

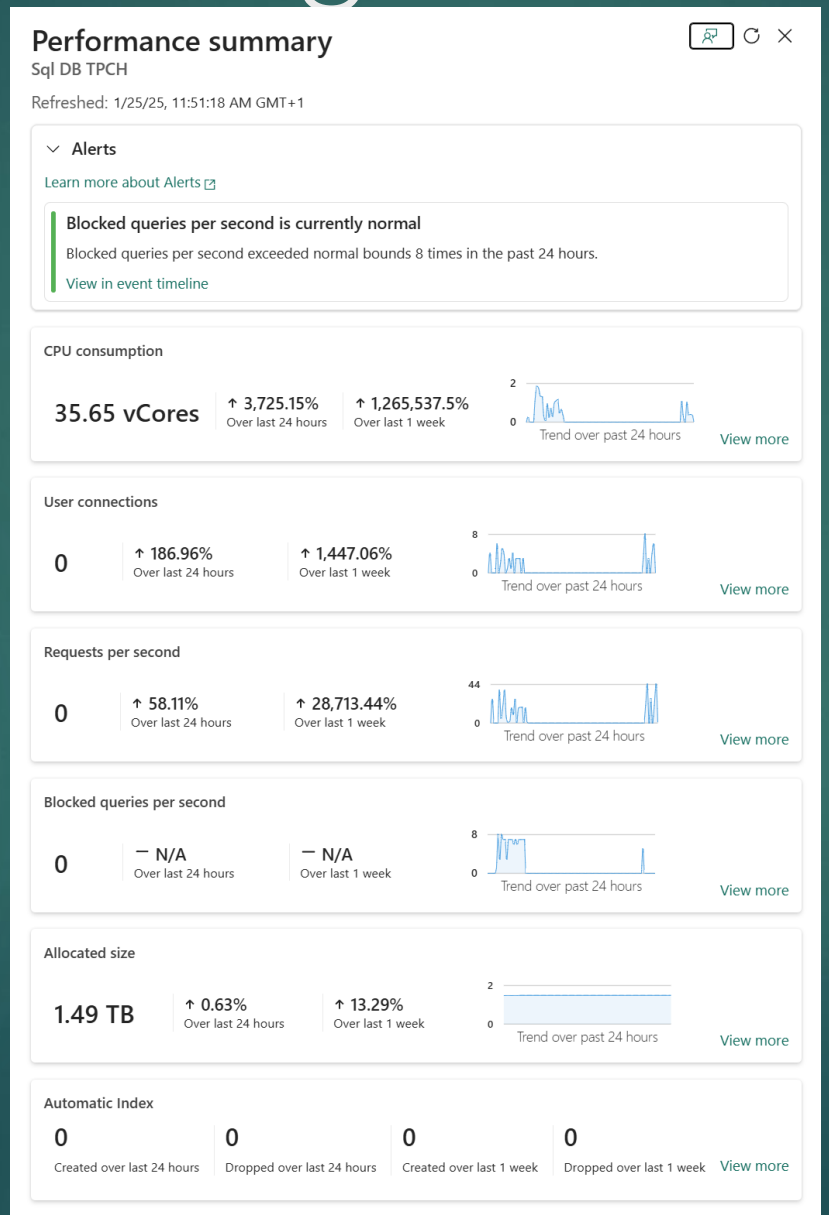


Capacity Metrics App

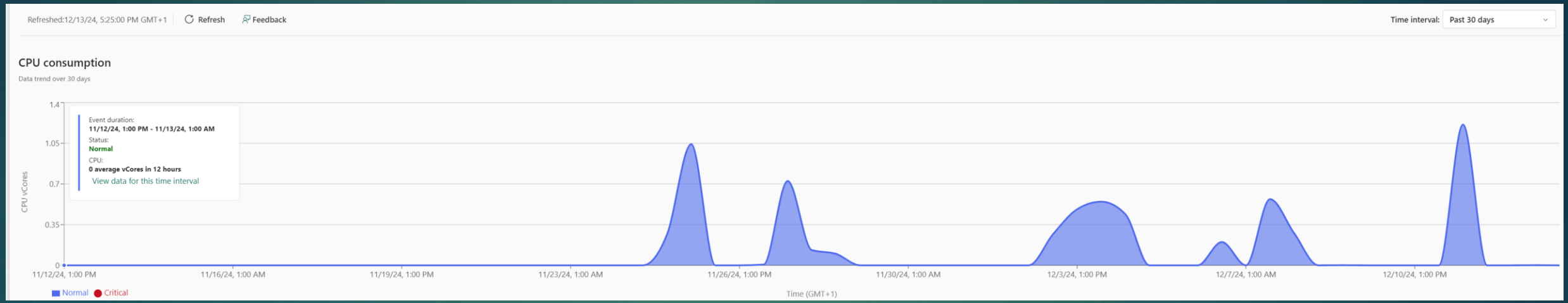


Operation name	CU (s)	Duration (s)	Users	Billing type
Sql Usage	270,102.0281	9,300.0000	1	Non-billable
OneLake Iterative Read via Proxy	43.7394	0.0640	1	Billable
OneLake Write via Redirect	40.4874	0.0290	1	Billable
OneLake Write via Proxy	28.0900	0.0250	1	Billable
OneLake Other Operations Via Redirect	6.1152	0.0180	1	Billable
OneLake Read via Redirect	3.2344	0.0090	1	Billable
OneLake Other Operations	2.3256	0.0340	1	Billable
OneLake Read via Proxy	0.1224	0.0040	1	Billable
Total	**270,226.1425**	**9,300.1830**	**2**	**Billable**

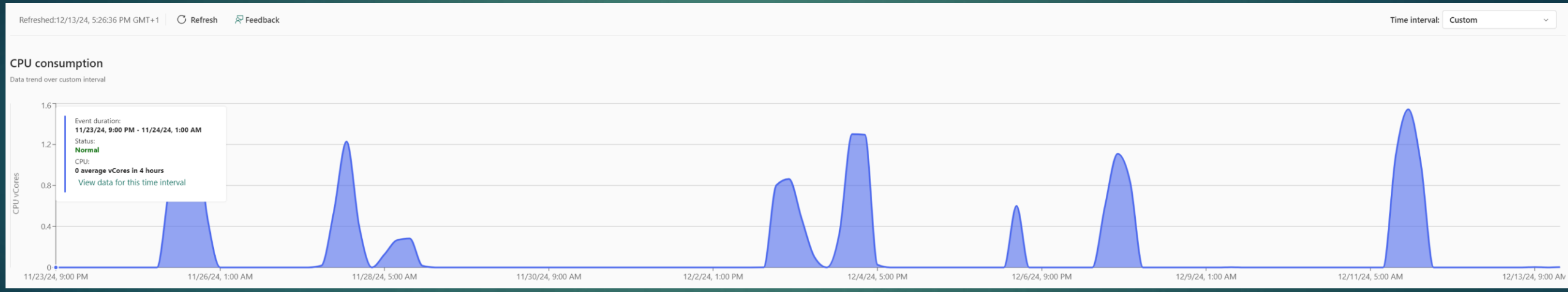
Built in monitoring: one level deeper



Built in monitoring: zoom levels matter

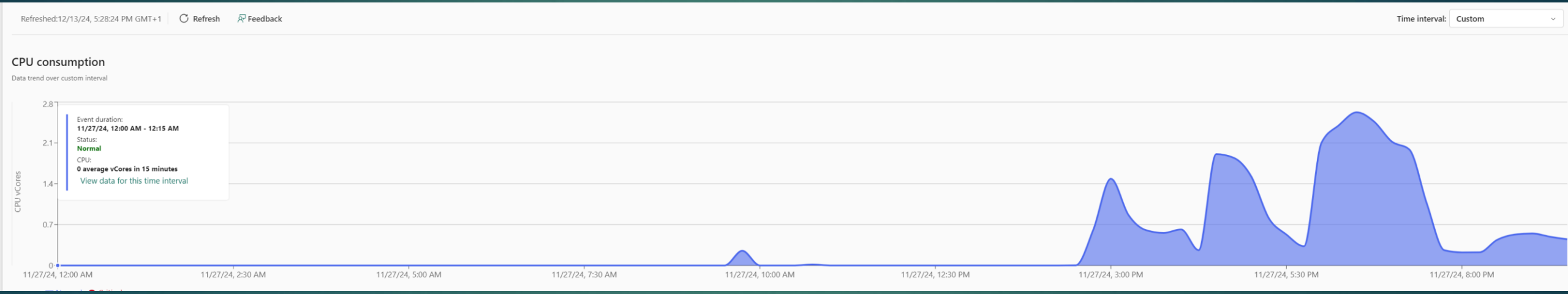


Built in monitoring: zoom levels matter



Built in monitoring: zoom levels matter

Built in monitoring: zoom levels matter



Monitoring, needs a little love

THE BASICS ARE THERE

SOME CONFUSING RESULTS

LOG FEEDBACK WITH MICROSOFT WITH WHAT YOU WANT TO SEE

My conclusions

Ingestion techniques matter

Some work to be done with settings and monitoring

Hello DBA skills!

The big question, does it fit your use cases?

WHAT DO YOU THINK, RAISE YOUR HANDS

Share **your thoughts** and help our speakers!

fabfeb.app/feedback

#FabricFebruary