Leveraging Query Store for Improved Performance

Thank You!

Thank you, LA Data Platform Meetup!!

About Me

- Entered IT in Feb 2008
- MCTS SQL Server 2008 Implementation and Maintenance in May 2013
- Azure Fundamentals and Azure Data Fundamentals in Feb/March 2021
- Database Administrator since 2014

About Me

Microsoft
CERTIFIED
AZURE
FUNDAMENTALS

Microsoft
CERTIFIED
AZURE DATA
FUNDAMENTALS

- Blog at https://leemarkum.com/
- Currently a Senior DBA on the Infinity DBA team at Rocket LLC (Views expressed here are my own)
- Have spoken at a number of SQL Server events including SQL Saturday Columbus as well as Saint Louis DevUp
- leemarkum@yahoo.com
- BlueSky - @leemarkum.bsky.social
- LinkedIn – https://linkedin.com/in/leemarkum

Setting Expectations

- The goal is to provide level 200 coverage of Query Store. We’ll talk some basics and get into deeper detail on some topics.
- This is a large topic and could easily be an all-day session at a conference.
- Follow up presentations would be needed to cover and demo specific query store functionality in a level 300 and above fashion.

Agenda

Discuss features in the following categories:

- Why Query Store
- What does Query Store do
- Intelligent Query Processing and Query Store
- How to configure Query Store
- SQL Server Community tools for examining Query Store data

Why Query Store?

Capturing, storing and aggregating performance metrics was hard

SQL trace, extended events, or querying the cache are expensive to do to get all the same data Query Store provides and in pre-aggregated way.

Management Data Warehouse was an attempt to bring query performance data inside SQL Server.

Also, you couldn’t do things like plan forcing, the performance feedback features in later versions of Query Store, and other things I’ll describe later on.

What does Query Store do?

- Stores aggregated query runtime performance based on AVG, MAX, MIN, etc.

  CPU Time (ms)

  Duration(ms)

  Logical reads(kb)

  Physical Reads(kb)

  DOP

  Memory Consumption(kb)

- Wait Stats information is available in SQL 2017 and above
- Custom Capture Policies in 2019 to allow better control over what is captured so Query Store performs better.

What does Query Store do?

- Reports that show performance graphically
    - Regressed Queries
    - Overall Resource Consumption
    - Top Resource Consuming Queries*
    - Queries with Forced Plans
    - Queries with High Variation
    - Query Wait Statistics*
    - Tracked Queries*

What does Query Store do?

- Can apply query hints without changing the code. (2022)

- Compare two plans to each other side by side

- Track a stored procedure’s performance over time by queryid – assuming no one drops and recreates the procedure.

- Allows you to detect regression and force “good query plans”

What does Query Store do?

Caveats to plan forcing with Query Store https://bit.ly/3XXB2oS

There are some limitations that can prevent a plan to be enforced.

First, if the plan contains following constructions:

- Insert bulk statement
- Reference to an external table
- Distributed query or full-text operations
- Use of elastic queries
- Dynamic or keyset cursors
- Invalid star join specification

What does Query Store do?

Caveats to plan forcing with Query Store https://bit.ly/3XXB2oS

There are some limitations that can prevent a plan to be enforced.

Second, when objects that plan relies on, are no longer available:

- Database (if database, where plan originated, doesn't exist anymore)
- Index (no longer there or disabled)

Finally, problems with the plan itself:

- Not legal for query
- Query Optimizer exceeded number of allowed operations
- Incorrectly formed plan XML

The point is, plan forcing isn't full-proof!

What does Query Store do?

Permissions for using Query Store:

To view/execute Query Store reports, the minimum permission needed is VIEW DATABASE STATE in each database where it is enabled.

The VIEW SERVER STATE permission is at the server level and includes VIEW DATABASE STATE for all databases.

Keep the issue of permissions in mind for Developers who may want access to view Query Store information.

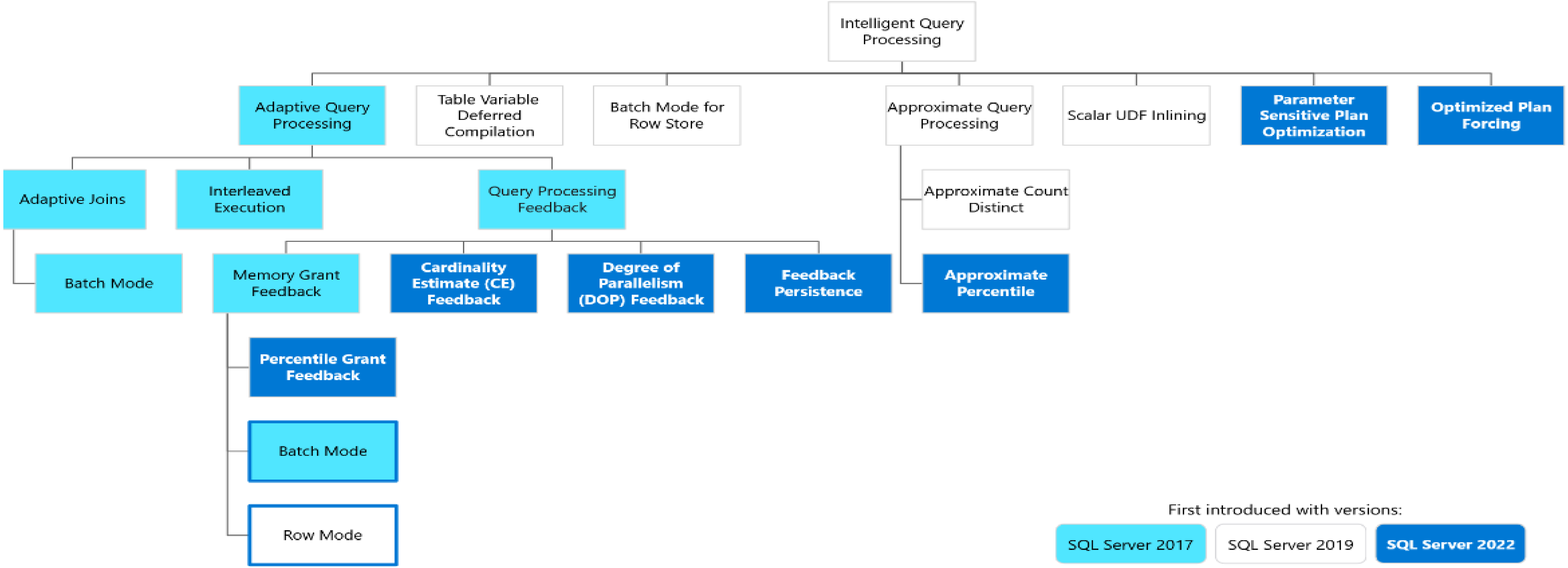
Leveraging Query Store for Improved Performance

Query Store:

How does it relate to Intelligent Query Processing?

Intelligent Query Processing and Query Store

Intelligent Query Processing Picture from MS Learn
https://bit.ly/3C0Zn1B


Intelligent Query Processing
Adaptive Query Processing
Table Variable Deferred Compilation
Batch Mode for Row Store
Approximate Query Processing
Scalar UDF Inlining
Parameter Sensitive Plan Optimization
Optimized Plan Forcing
Adaptive Joins
Interleaved Execution
Query Processing Feedback
Approximate Count Distinct
Batch Mode
Memory Grant Feedback
Cardinality Estimate (CE) Feedback
Degree of Parallelism (DOP) Feedback
Feedback Persistence
Approximate Percentile
Percentile Grant Feedback
Batch Mode
Row Mode
First introduced with versions:
SQL Server 2017
SQL Server 2019
SQL Server 2022

Intelligent Query Processing and Query Store

- Memory Grant Feedback – Batch mode (SQL Server 2017)
- Memory Grant Feedback - Row Mode (SQL Server 2019)
- Memory Grant Feedback – Percentile and Persisted (SQL Server 2022)
- DOP_Feedback (SQL Server 2022)
- CE Feedback (SQL Server 2022)
- Query Store Hints (SQL Server 2022)

Intelligent Query Processing and Query Store

- Memory grant feedback (SQL Server 2017)

- Can reduce spills to disk by increasing memory grants

- Can reduce wasted memory by lowering memory grants that are too high

- Can result in a “zig-zag” effect where memory grants are constantly shifting up and down based on parameters passed to the affected query

Intelligent Query Processing and Query Store

- Memory grant feedback (SQL Server 2017)

- Feedback is lost on SQL Server restart or use of RECOMPILE

- Needs database compatibility level 140 for batch mode and 150 for row mode

Leveraging Query Store for Improved Performance

- Persisted and Percentile memory grant feedback (SQL Server 2022)

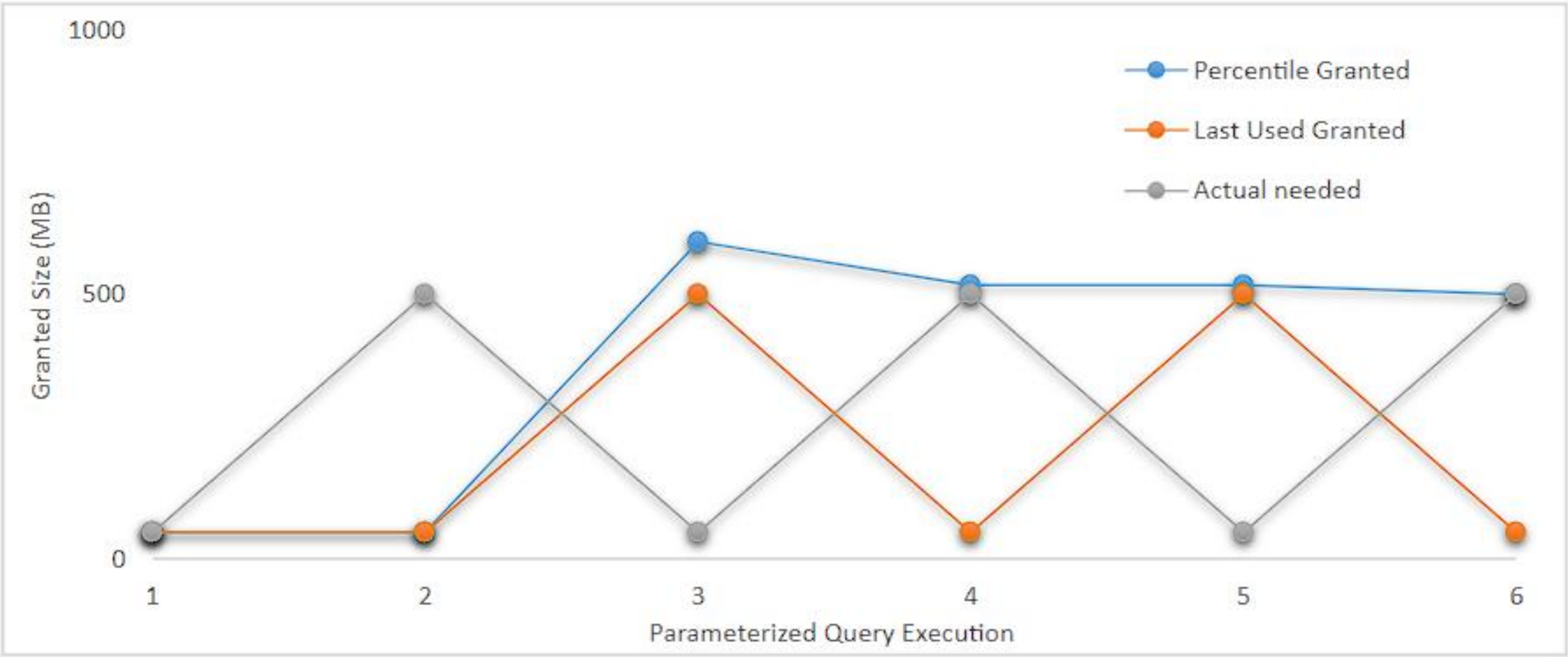
- Needs compat level 140 and higher.

- Needs Query Store enabled and in read-write mode.

- Feedback will be persisted in the sys.query_store_plan_feedback catalog view

Leveraging Query Store for Improved Performance

Persisted And Percentile Memory Grant Feedback (SQL Server 2022)



Leveraging Query Store for Improved Performance

- DOP_Feedback (SQL Server 2022)

- Needs compat level 160

- Needs Query Store enabled and in read/write mode

- Only verified feedback is persisted

- Feedback will be persisted in the sys.query_store_plan_feedback catalog view when we reach a stable degree of parallelism feedback value.

Leveraging Query Store for Improved Performance

How is Query Store Configured?

How is Query Store Configured?

```
ALTER DATABASE MyDB SET QUERY_STORE = ON;
```

It’s that simple to enable, but there are 8 settings that need to be understood.

How is Query Store Configured?

Operation_Mode: Off, Read_Write, Read_only. Default is off in the on-prem version for 2016-2019. On is the default in 2022.

Cleanup_Policy/Stale_Query_Threshold_Days: Number of days to hold data for. Default is 30 days in the on-prem version

How is Query Store Configured?

Data_Flush_Interval_Seconds: How often will in-memory data be flushed to disk. Default is 900 seconds(15 minutes)

Max_Storage_Size_MB: The largest amount of data to be held on disk. Defaults are all too small for most scenarios(100 MB in 2016/2017, 1000 in 2019)

How is Query Store Configured?

Interval_Length_Minutes: Controls the span of aggregated data. Default is 60. The smaller the interval, the more disk space used, but the more granular the data will be.

Size_Based_Cleanup_Mode: Attempt to cleanup as max size is approached. Default is Auto, which means this is enabled by default.

How is Query Store Configured?

Max_Plans_Per_Query: How many distinct plans to keep per query. Default is 200. Option doesn’t exist in 2016.

Wait_Statistics_Capture_Mode: Captures wait stats at database and query level. Default is on. Not available in 2016.

How is Query Store Configured?

Query_Store_Capture_Mode: Determines the behavior Query Store uses to capture queries.

Default is All for 2016/17.

Auto for 2019.

Also has Custom mode available in 2019.

How is Query Store Configured?

Query_Store_Capture_Mode: Custom mode details -

Help control how data will be stored and which data, allowing granular capture control for ad-hoc workloads.

Stale_Capture_Policy_Threshold (SCPT): Defines the time span during which a query must exceed one of the values in the next three options in order for it to be captured. Valid values are 1 hour up to 7 days.

How is Query Store Configured?

Query_Store_Capture_Mode: Custom mode details -

Execution_Count: How many times a query must be executed within the Stale Capture Policy Threshold.

Total_Compile_CPU_Time_ms: Total cpu compile time a query must use within the Stale Capture Policy Threshold.

Total_Execution_CPU_Time_ms: The total cpu execution time the query must use within the Stale Capture Policy Threshold.

How is Query Store Configured?

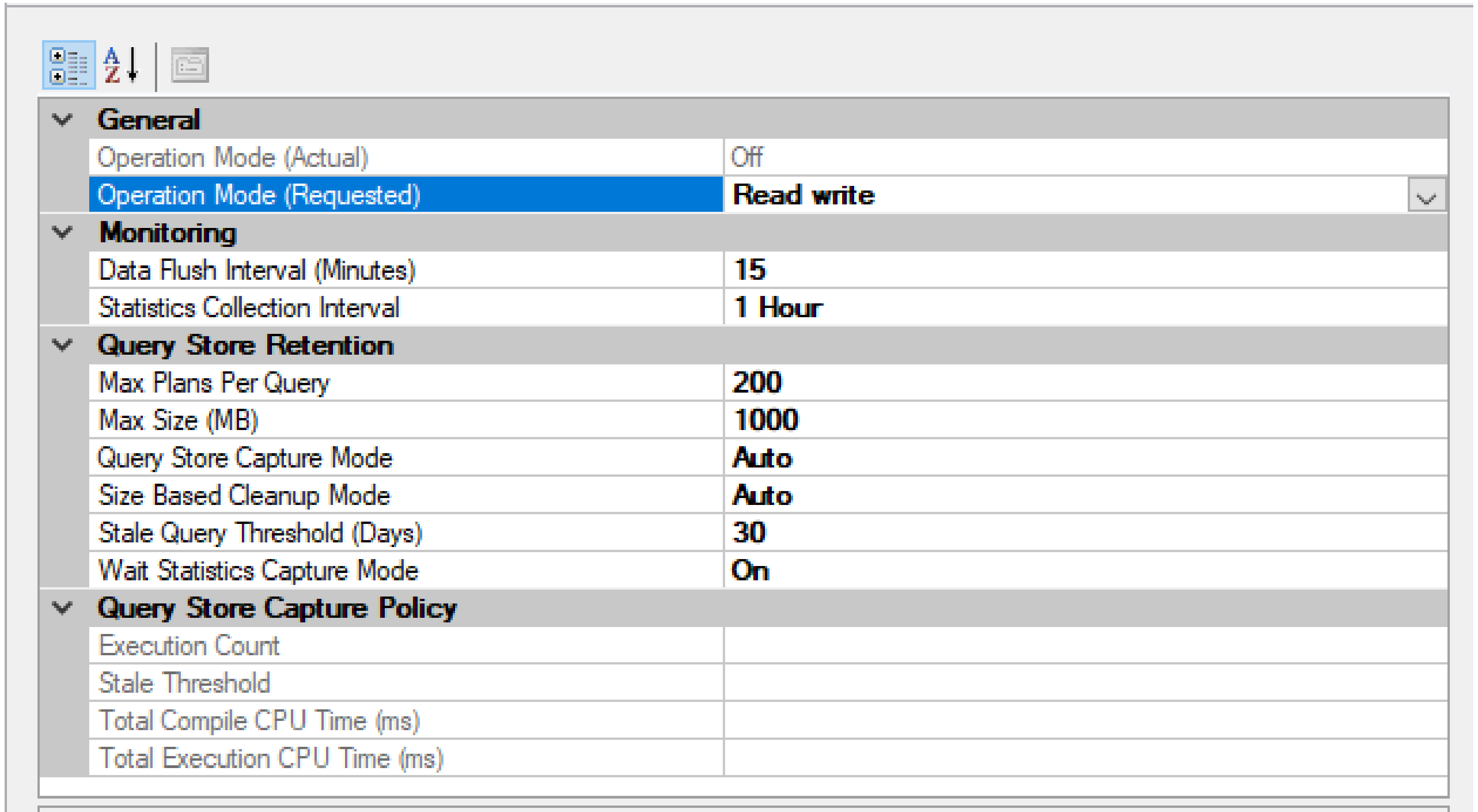
Trace Flags

7745 - prevents Query store data from writing to disk prior to shutdown or failover process so it doesn’t delay a shutdown or failover.

7752 - Loads Query Store data to memory asynchronously from query execution.

How is Query Store Configured?

Defaults from 2019.



General	
Operation Mode (Actual)	Off
Operation Mode (Requested)	Read write
Monitoring	
Data Flush Interval (Minutes)	15
Statistics Collection Interval	1 Hour
Query Store Retention	
Max Plans Per Query	200
Max Size (MB)	1000
Query Store Capture Mode	Auto
Size Based Cleanup Mode	Auto
Stale Query Threshold (Days)	30
Wait Statistics Capture Mode	On
Query Store Capture Policy	
Execution Count	
Stale Threshold	
Total Compile CPU Time (ms)	
Total Execution CPU Time (ms)	

Community Scripts for Query Store

What community scripts exist for querying the Query Store system tables?

Sp_BlitzQueryStore from the First Responder Kit on GitHub (deprecated – On Slack, Brent said to point people to sp_QuickieStore)

https://bit.ly/42aqzc7 (sp_QuickieStore from Erik Darling Data on GitHub)

https://bit.ly/4i6ngY5 (Erin Stellato’s Query Store GitHub)

https://bit.ly/4ifL0t1 (David Schanzer Query Store Toolbox)

https://bit.ly/3Ei13bP (MS Docs Tune Performance with query Store)

Learning More About Query Store

- https://bit.ly/2JGUC5J (Blog:Lee Markum - Overview of Query Store)
- http://bit.ly/3G2PhTj (Blog:Query Store category on SQLSkills.com)
- https://bit.ly/4cUUgjO (Video: Erin Stellato - Why you Need Query Store)
- https://bit.ly/3W61tqw (Video: Erin Stellato - Query Store Best Practices)
- https://bit.ly/4i6ngY5 (Erin Stellato’s Query Store Github)

Learning More About Query Store

- https://bit.ly/3Ei13bP (MS Docs Tune Performance with query Store)
- “Query Store for SQL Server 2019” Tracy Boggiano & Grant Fritchey
- “SQL Server 2022 Revealed” Bob Ward
- “SQL Server 2022 Query Performance Tuning” Grant Fritchey

Leveraging Query Store for Improved Performance

Summary

- Why Query Store?
- What does it do?
- How does Query Store work with intelligent Query Processing?
- How is it configured?
- What community scripts currently exist for Query Store?

Leveraging Query Store for Improved Performance

Thank you for attending!

Leveraging Query Store for Improved Performance

QUESTIONS?

Contacting Me

- leemarkum@yahoo.com
- BlueSky - @leemarkum.bsky.social
- LinkedIn – https://linkedin.com/in/leemarkum