Introduction to the Query Store

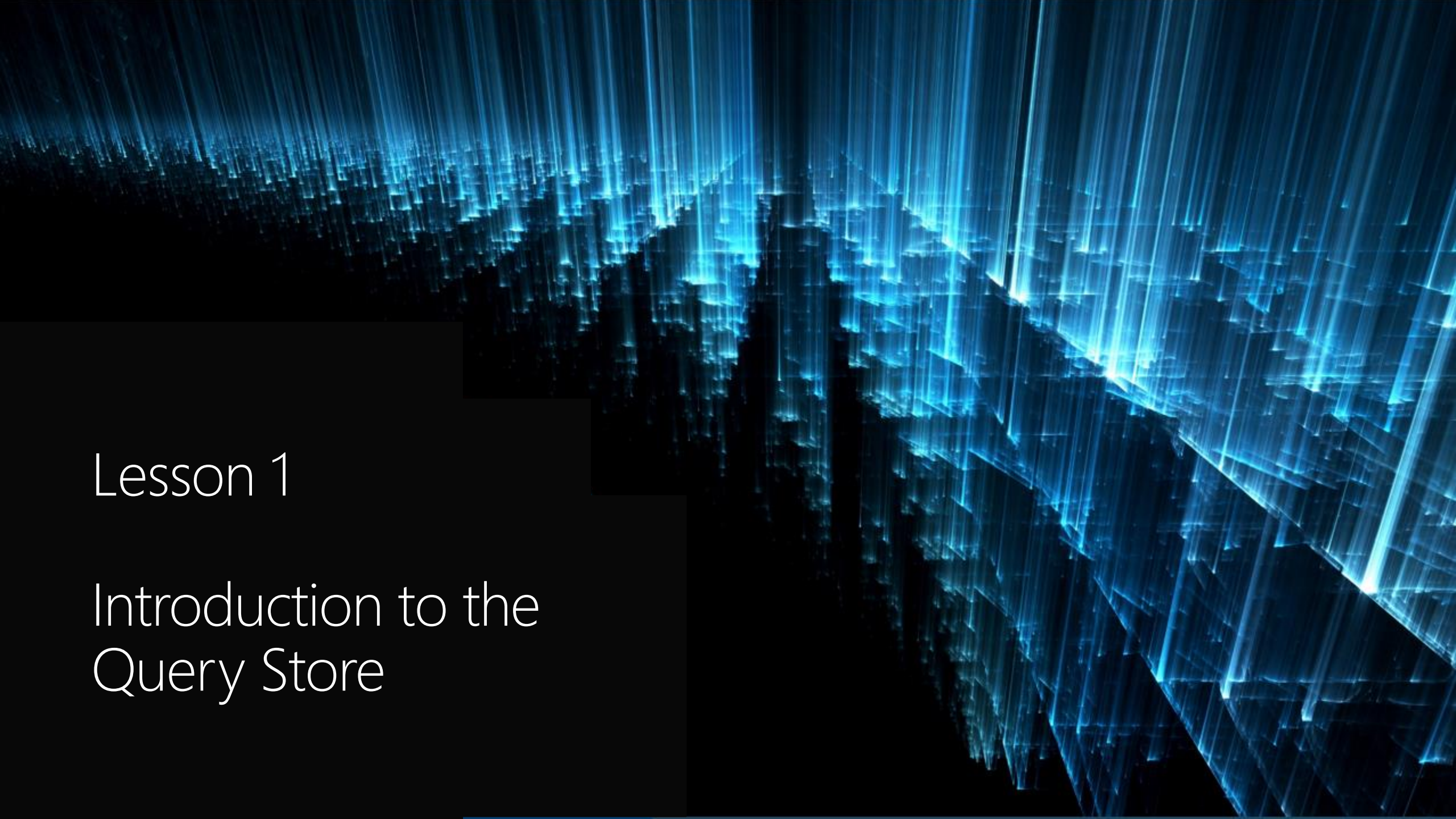
John Deardurff

John Deardurff

Microsoft Customer Engineer (Global Technical Team)
Microsoft Certified Trainer (Regional Lead)
MVP: Data Platform (2016 – 2018)
Email: John.Deardurff@Microsoft.com
Twitter: @SQLMCT
Website: www.SQLMCT.com
GitHub: github.com\SQLMCT

What does this session cover?

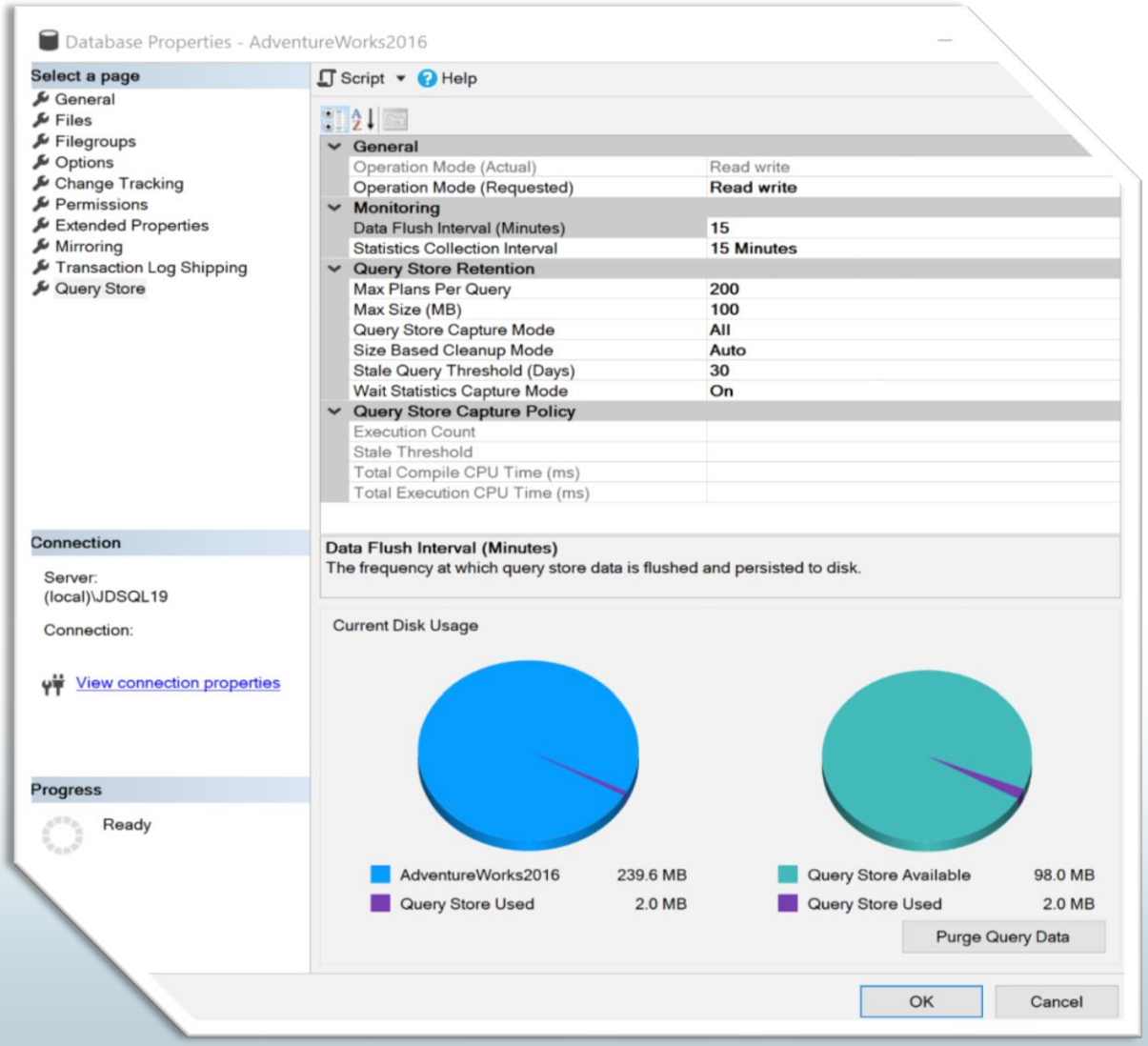
- Introduction to the Query Store
- Query Store Settings
- Troubleshooting with Query Store Reports
- Query Store Catalog Views



Lesson 1

Introduction to the Query Store

Introducing the Query Store



Query Store is set at the database level

Cannot be used for Master or TempDB system databases but can be enabled for the Model and MSDB system databases.

The user database stores the data in internal tables that can be accessed by using built-in Query Store views.

SQL Server retains this data until the space allocated to Query Store is full or manually purged.

Why use Query Store?

Before Query Store

- Requires manual proactive monitoring to identify execution plan problems.
- Only the latest plan was stored in the procedure cache
- Restart caused data to be lost
- Frequent recompiles of procedures or use of DBCC FREEPROCACHE
- No history or aggregated gathering of data available.

With Query Store

- It stores the history of the execution plans for each query
- It establishes a performance baseline for each plan over time
- It identifies queries that may have regressed
- It is possible to force plans quickly and easily
- It works across server restarts, upgrades, and query recompilation

Query Store contains three stores

Plan Store

- Persists execution plan information
- Stores Query Text and Query Plan information
- You can limit the number of unique plans by using the max_plans_per_query configuration.

Runtime Stats Store

- Persists execution statistics information
- Captures Compile Time, Duration, CPU usage, Writes, and both Logical and Physical Reads
- To minimize space usage, the runtime stats are aggregated over a fixed time window.

Wait Stat Store

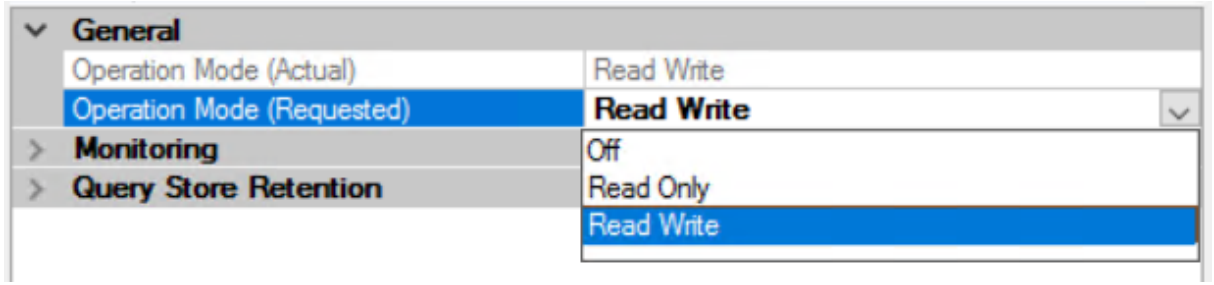
- Persists wait statistics information
- Introduced in SQL Server 2017 and Azure SQL Database
- Organizes wait stats into different wait categories to simplify troubleshooting similar wait types.

Lesson 2

Query Store Settings

Query Store Operation Modes

Operation Mode can be set under database properties



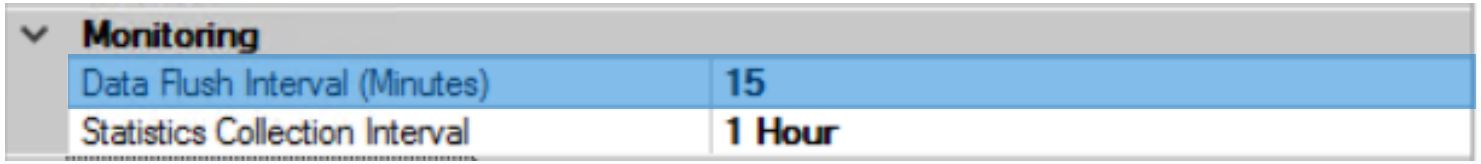
Operation Mode can be enabled two ways using T-SQL. If only using the ON option, the Mode defaults to **Read_Write**

```
ALTER DATABASE [AdventureWorks2016] SET QUERY_STORE = ON;
```

```
ALTER DATABASE [AdventureWorks2016] SET QUERY_STORE
(OPERATION_MODE = READ_WRITE);
```

Query Store Monitoring Settings

Data Flush Interval determines the frequency at which data written to the query store is persisted to disk. (Default is **15 Minutes**).




```
ALTER DATABASE [AdventureWorks2016] SET QUERY_STORE
(INTERVAL_LENGTH_MINUTES = 1,
DATA_FLUSH_INTERVAL_SECONDS = 60)
```

Query Store Monitoring Settings

Statistics Collection Interval determines the time interval at which runtime execution statistics data is aggregated into the query store. Only the values of 1, 5, 10, 14, 60, and 1440 minutes is allowed. (Default is **60**).

```
ALTER DATABASE [AdventureWorks2016] SET QUERY_STORE
(INTERVAL_LENGTH_MINUTES = 1,
DATA_FLUSH_INTERVAL_SECONDS = 60)
```

Query Store Retention Settings

Max Plans Per Query is a new retention setting introduced in SQL Server 2017 and is an integer representing the maximum number of plans maintained for each query. (Default is **200**).

Query Store Retention	
Max Plans Per Query	200
Max Size (MB)	100
Query Store Capture Mode	Custom
Size Based Cleanup Mode	Auto
Stale Query Threshold (Days)	30
Wait Statistics Capture Mode	On

```
ALTER DATABASE AdventureWorks2016 SET QUERY_STORE
(MAX_PLANS_PER_QUERY = 200,
 MAX_STORAGE_SIZE_MB = 100,
 QUERY_CAPTURE_MODE = AUTO,
 SIZE_BASED_CLEANUP_MODE = AUTO,
 CLEANUP_POLICY = (STALE_QUERY_THRESHOLD_DAYS = 367),
 WAIT_STATS_CAPTURE_MODE = ON);
GO
```

Query Store Retention Settings

Max Size (MB) configures the maximum storage size for the query store. (Default is **100MB**) When the query store limit is reached, query store changes the state from read-write to read-only.

Query Store Retention	
Max Plans Per Query	200
Max Size (MB)	100
Query Store Capture Mode	Custom
Size Based Cleanup Mode	Auto
Stale Query Threshold (Days)	30
Wait Statistics Capture Mode	On

```
ALTER DATABASE AdventureWorks2016 SET QUERY_STORE
(MAX_PLANS_PER_QUERY = 200,
 MAX_STORAGE_SIZE_MB = 100,
 QUERY_CAPTURE_MODE = AUTO,
 SIZE_BASED_CLEANUP_MODE = AUTO,
 CLEANUP_POLICY = (STALE_QUERY_THRESHOLD_DAYS = 367),
 WAIT_STATS_CAPTURE_MODE = ON);
GO
```

Query Store Retention Settings

Query Store Capture Mode determines to capture all the queries (Default is **ALL**), or relevant queries based on execution count and resource consumption (**AUTO**) or stop capturing queries (**NONE**). SQL Server 2019 introduces an additional (**CUSTOM**) setting.

Query Store Retention	
Max Plans Per Query	200
Max Size (MB)	100
Query Store Capture Mode	Custom
Size Based Cleanup Mode	Auto
Stale Query Threshold (Days)	30
Wait Statistics Capture Mode	On

```
ALTER DATABASE AdventureWorks2016 SET QUERY_STORE
(MAX_PLANS_PER_QUERY = 200,
 MAX_STORAGE_SIZE_MB = 100,
 QUERY_CAPTURE_MODE = AUTO,
 SIZE_BASED_CLEANUP_MODE = AUTO,
 CLEANUP_POLICY = (STALE_QUERY_THRESHOLD_DAYS = 367),
 WAIT_STATS_CAPTURE_MODE = ON);
GO
```

Query Store Retention Settings

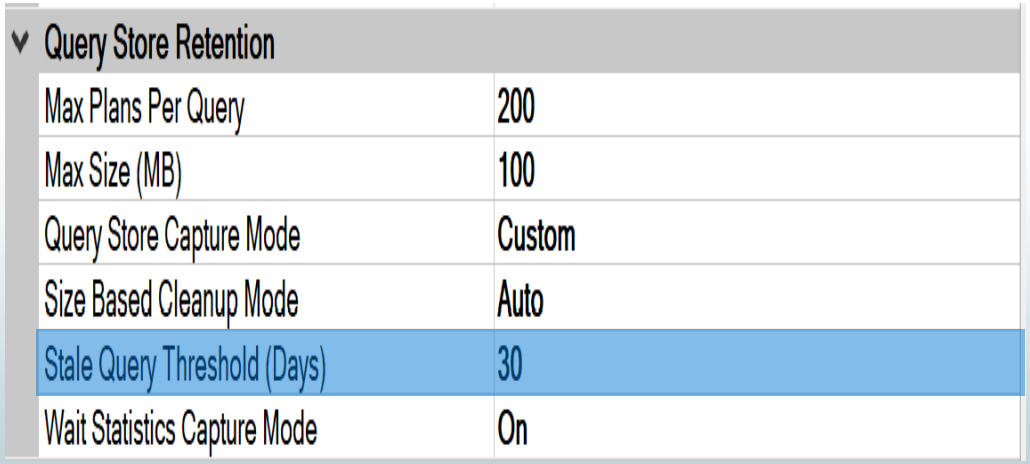
Size Based Cleanup Mode determines whether the cleanup process will be automatically activated when the total amount of data gets close to the maximum size. (Default is **Auto**).

Query Store Retention	
Max Plans Per Query	200
Max Size (MB)	100
Query Store Capture Mode	Custom
Size Based Cleanup Mode	Auto
Stale Query Threshold (Days)	30
Wait Statistics Capture Mode	On

```
ALTER DATABASE AdventureWorks2016 SET QUERY_STORE
(MAX_PLANS_PER_QUERY = 200,
 MAX_STORAGE_SIZE_MB = 100,
 QUERY_CAPTURE_MODE = AUTO,
 SIZE_BASED_CLEANUP_MODE = AUTO,
 CLEANUP_POLICY = (STALE_QUERY_THRESHOLD_DAYS = 367),
 WAIT_STATS_CAPTURE_MODE = ON);
GO
```

Query Store Retention Settings

Stale Query Threshold (Days) determines the number of days to retain data in the query store. (Default is **30 days** and Maximum is **367 days**).



Query Store Retention	
Max Plans Per Query	200
Max Size (MB)	100
Query Store Capture Mode	Custom
Size Based Cleanup Mode	Auto
Stale Query Threshold (Days)	30
Wait Statistics Capture Mode	On

```
ALTER DATABASE AdventureWorks2016 SET QUERY_STORE
(MAX_PLANS_PER_QUERY = 200,
 MAX_STORAGE_SIZE_MB = 100,
 QUERY_CAPTURE_MODE = AUTO,
 SIZE_BASED_CLEANUP_MODE = AUTO,
 CLEANUP_POLICY = (STALE_QUERY_THRESHOLD_DAYS = 367),
 WAIT_STATS_CAPTURE_MODE = ON);
GO
```

Query Store Retention Settings

Wait Statistics Capture Mode is a new retention setting introduced in SQL Server 2017 that controls if Query Store captures wait statistics information.

(Default = **ON**).

Query Store Retention	
Max Plans Per Query	200
Max Size (MB)	100
Query Store Capture Mode	Custom
Size Based Cleanup Mode	Auto
Stale Query Threshold (Days)	30
Wait Statistics Capture Mode	On

```
ALTER DATABASE AdventureWorks2016 SET QUERY_STORE
(MAX_PLANS_PER_QUERY = 200,
 MAX_STORAGE_SIZE_MB = 100,
 QUERY_CAPTURE_MODE = AUTO,
 SIZE_BASED_CLEANUP_MODE = AUTO,
 CLEANUP_POLICY = (STALE_QUERY_THRESHOLD_DAYS = 367),
 WAIT_STATS_CAPTURE_MODE = ON);
GO
```

Query Store Capture Policy Settings

Introduced in SQL Server 2019 and available if the Query Store Capture Mode setting has been set to **CUSTOM.**

The value for the **EXECUTION COUNT** is the value a query must exceed within the Stale Threshold time period to be captured by the Query Store.

Query Store Capture Policy	
Execution Count	30
Stale Threshold	1 Hour
Total Compile CPU Time (ms)	1000
Total Execution CPU Time (ms)	100

```
ALTER DATABASE AdventureWorks2016 SET QUERY_STORE
(QUERY_CAPTURE_POLICY =
 (EXECUTION_COUNT = 100,
 STALE_CAPTURE_POLICY_THRESHOLD = 24 Hours,
 TOTAL_COMPILE_CPU_TIME_MS = 2000,
 TOTAL_EXECUTION_CPU_TIME_MS = 1000));
GO
```

Query Store Capture Policy Settings

Introduced in SQL Server 2019 and available if the Query Store Capture Mode setting has been set to **CUSTOM.**

The value for the **Stale Threshold** can be from 1 hour up to 7 days. This setting specifies the time given to exceed the values of the three other settings for a query to be captured.

Query Store Capture Policy	
Execution Count	30
Stale Threshold	1 Hour
Total Compile CPU Time (ms)	1000
Total Execution CPU Time (ms)	100

```
ALTER DATABASE AdventureWorks2016 SET QUERY_STORE
(QUERY_CAPTURE_POLICY =
 (EXECUTION_COUNT = 100,
 STALE_CAPTURE_POLICY_THRESHOLD = 24 Hours,
 TOTAL_COMPILE_CPU_TIME_MS = 2000,
 TOTAL_EXECUTION_CPU_TIME_MS = 1000));
GO
```

Query Store Capture Policy Settings

Introduced in SQL Server 2019 and available if the Query Store Capture Mode setting has been set to **CUSTOM.**

The value for the **Total Compile CPU Time (ms)** is the value in milliseconds that a query must exceed within the **Stale Threshold** time period to be captured by the Query Store.

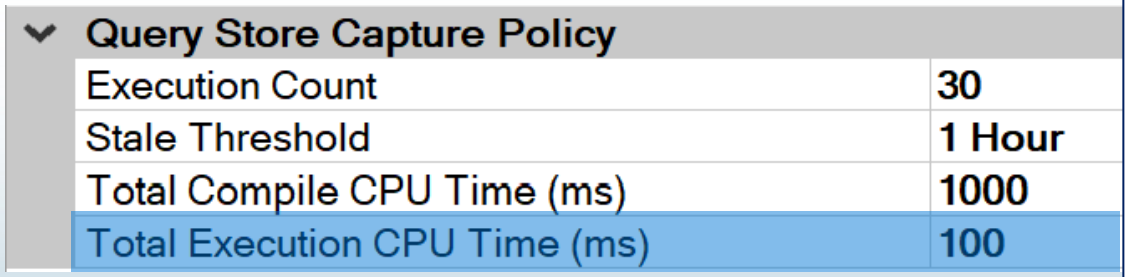
Query Store Capture Policy	
Execution Count	30
Stale Threshold	1 Hour
Total Compile CPU Time (ms)	1000
Total Execution CPU Time (ms)	100

```
ALTER DATABASE AdventureWorks2016 SET QUERY_STORE
(QUERY_CAPTURE_POLICY =
 (EXECUTION_COUNT = 100,
 STALE_CAPTURE_POLICY_THRESHOLD = 24 Hours,
 TOTAL_COMPILE_CPU_TIME_MS = 2000,
 TOTAL_EXECUTION_CPU_TIME_MS = 1000));
GO
```

Query Store Capture Policy Settings

Introduced in SQL Server 2019 and available if the Query Store Capture Mode setting has been set to **CUSTOM.**

The value for the **Total Execution CPU Time (ms)** is the value in milliseconds that a query must exceed within the **Stale Threshold** time period to be captured by the Query Store.

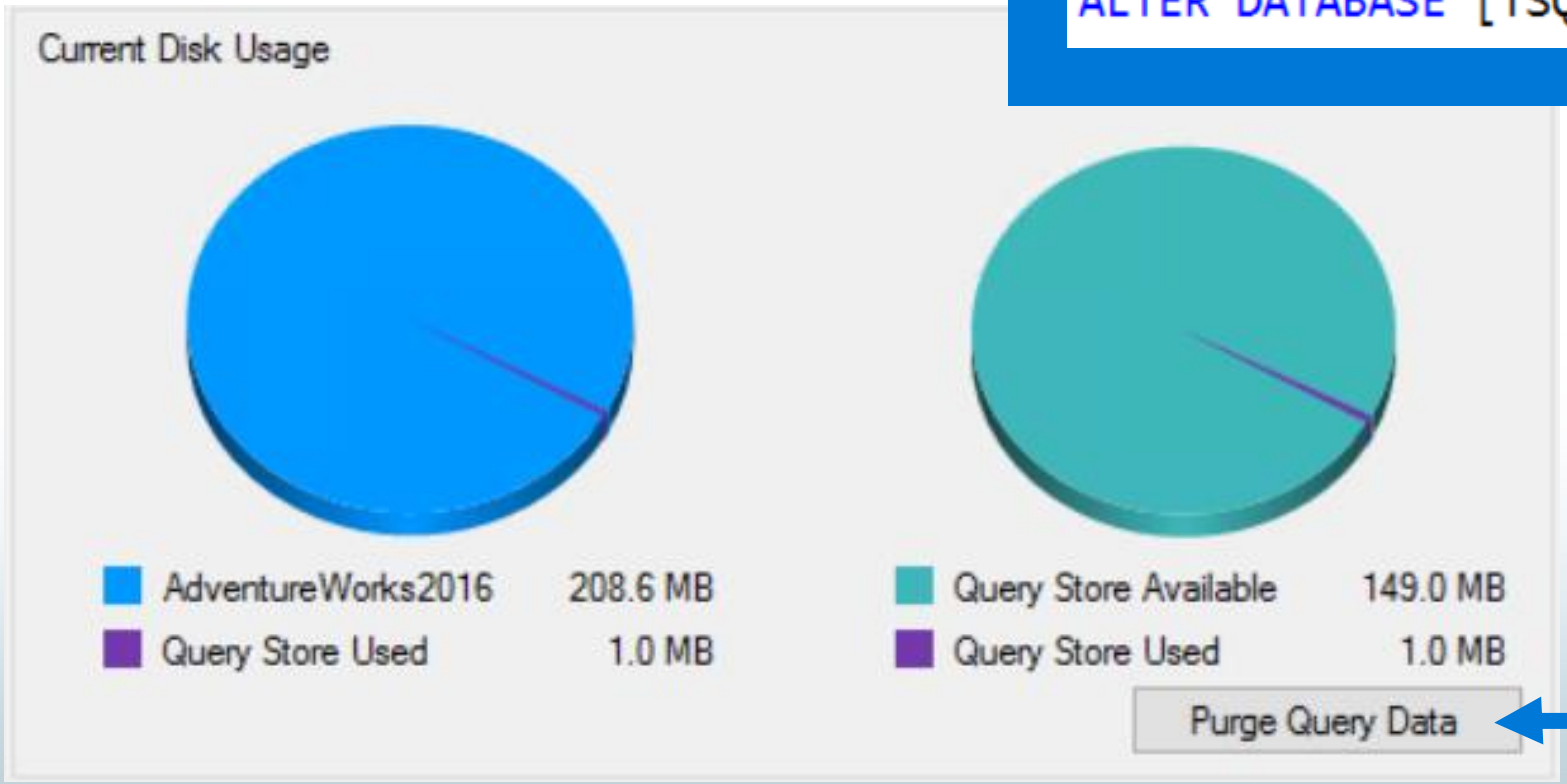



```
ALTER DATABASE AdventureWorks2016 SET QUERY_STORE
(QUERY_CAPTURE_POLICY =
 (EXECUTION_COUNT = 100,
 STALE_CAPTURE_POLICY_THRESHOLD = 24 Hours,
 TOTAL_COMPILE_CPU_TIME_MS = 2000,
 TOTAL_EXECUTION_CPU_TIME_MS = 1000));
GO
```

Purge Query Data

Data can be manually purged from the Query Store.

```
ALTER DATABASE [TSQL] SET QUERY_STORE CLEAR;
```



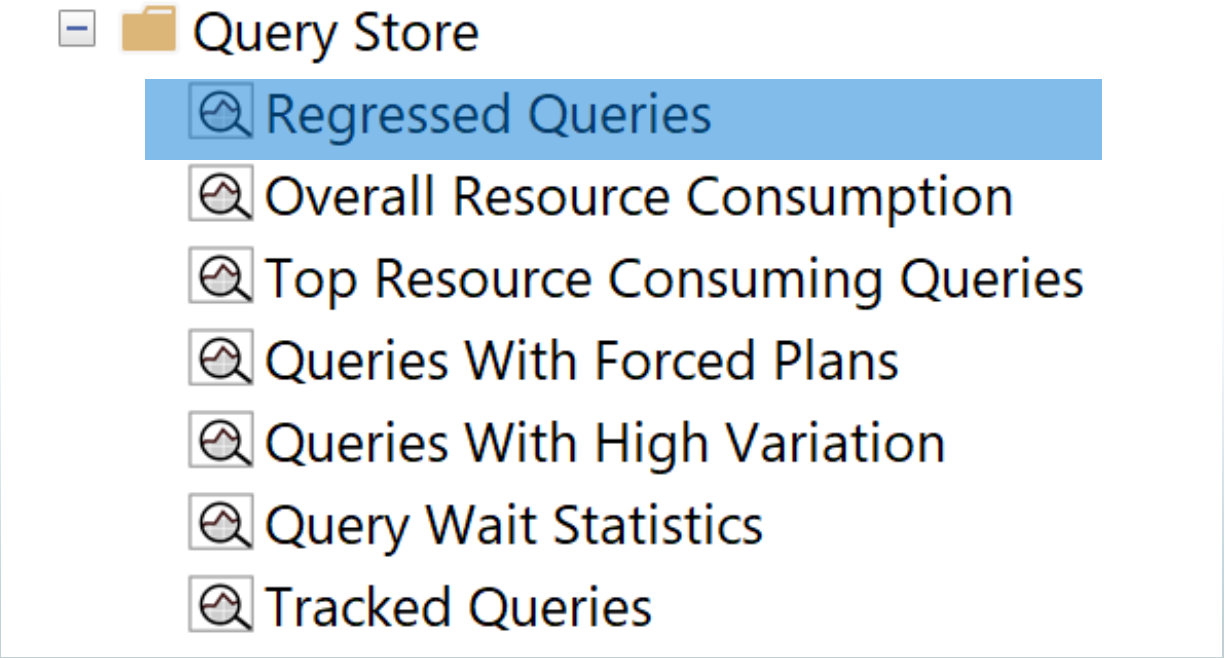
Demonstration

Lesson 3

Troubleshooting with Query Store Reports

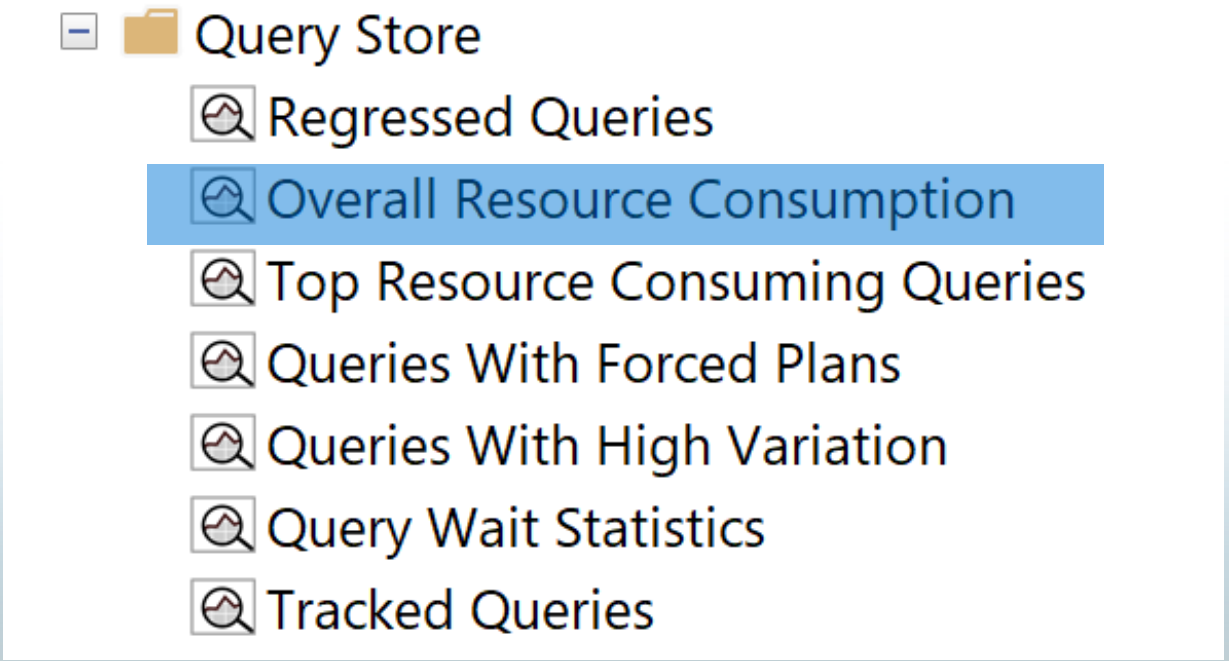
Query Store Reports

Regressed Queries: Use this dashboard to review queries that might have regressed because of execution plan changes



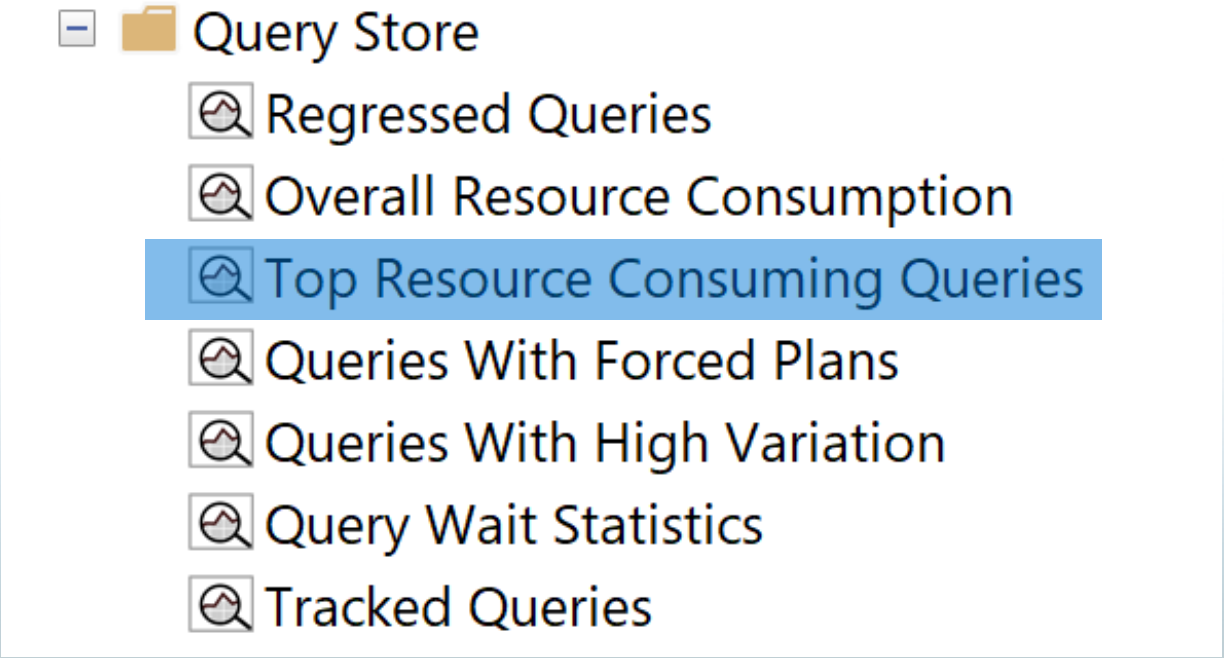
Query Store Reports

Overall Resource Consumption: Use this dashboard to visualize overall resource consumption during the last month in four charts: duration, execution count, CPU time, and logical reads



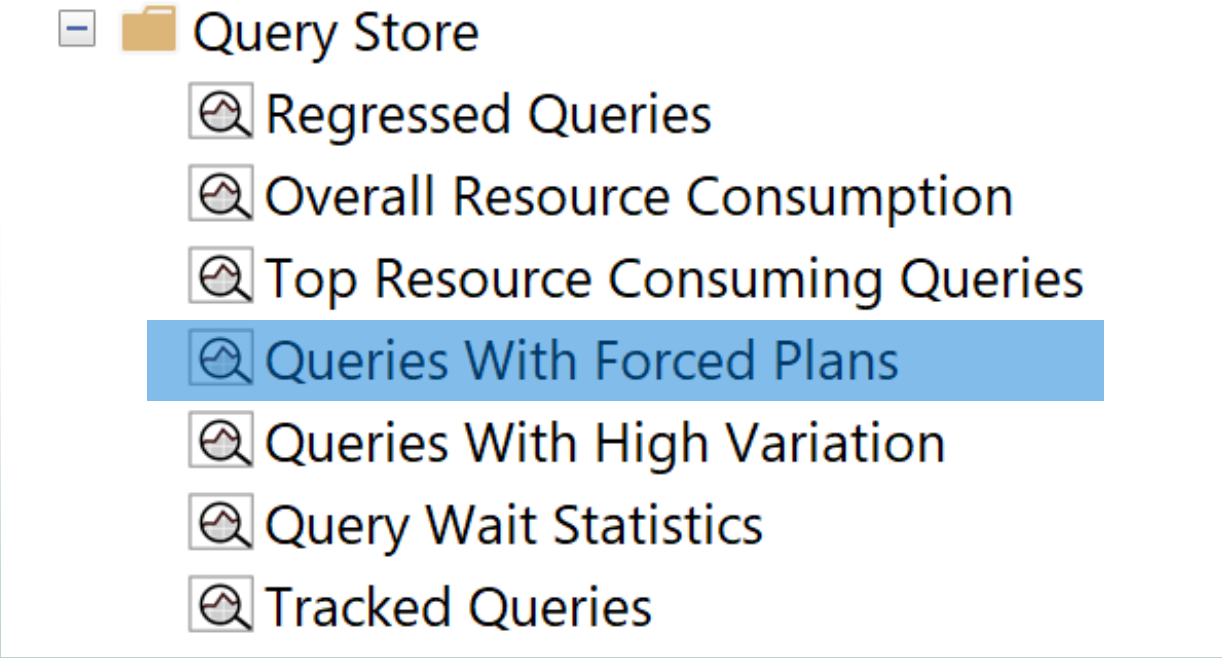
Query Store Reports

Top Resource Consuming Queries: Use this dashboard to review queries in the set of top 25 resource consumers during the last hour



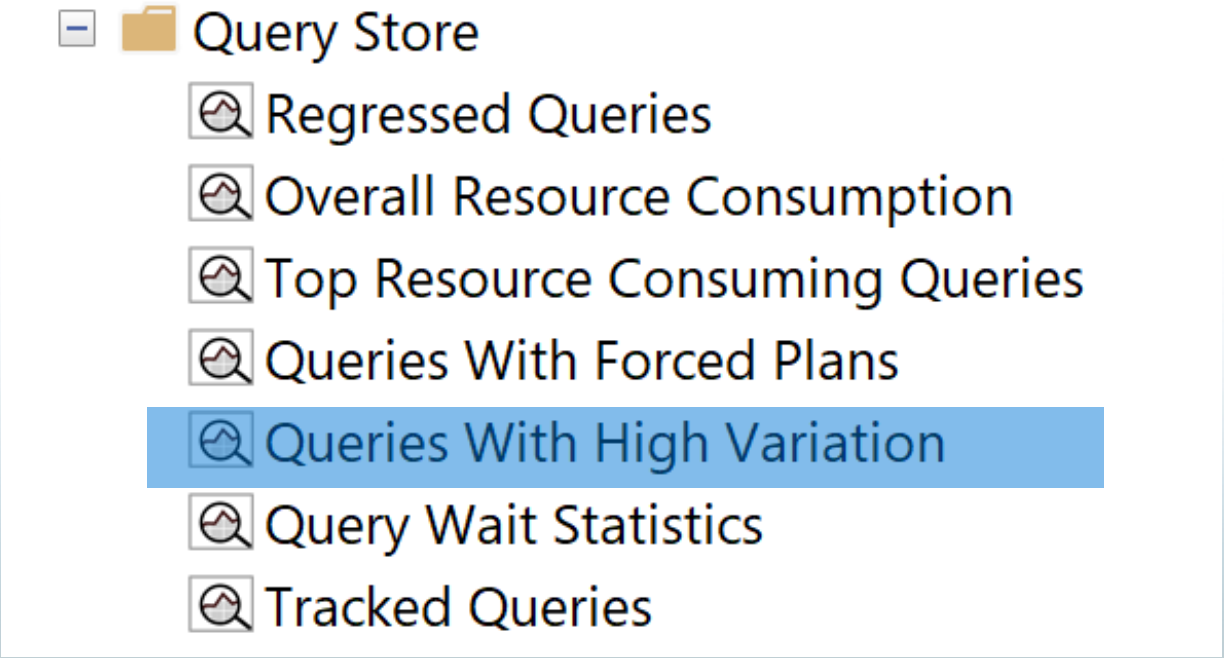
Query Store Reports

Queries With Forced Plans: Used to isolate queries that have been given a forced plan. Requires SQL Server 2016 SP1 or later.



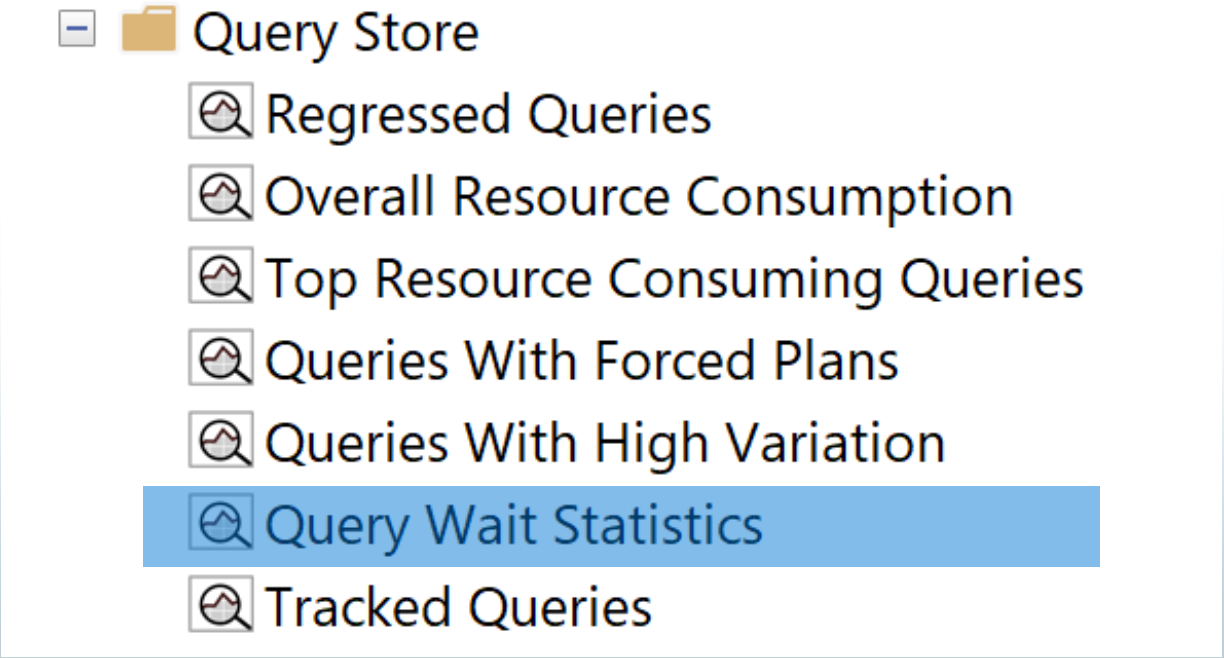
Query Store Reports

Queries With High Variation: Used to locate queries with high variation in query execution. Useful to locate queries with parameterization problems. Requires SQL Server 2016 SP1 or later.



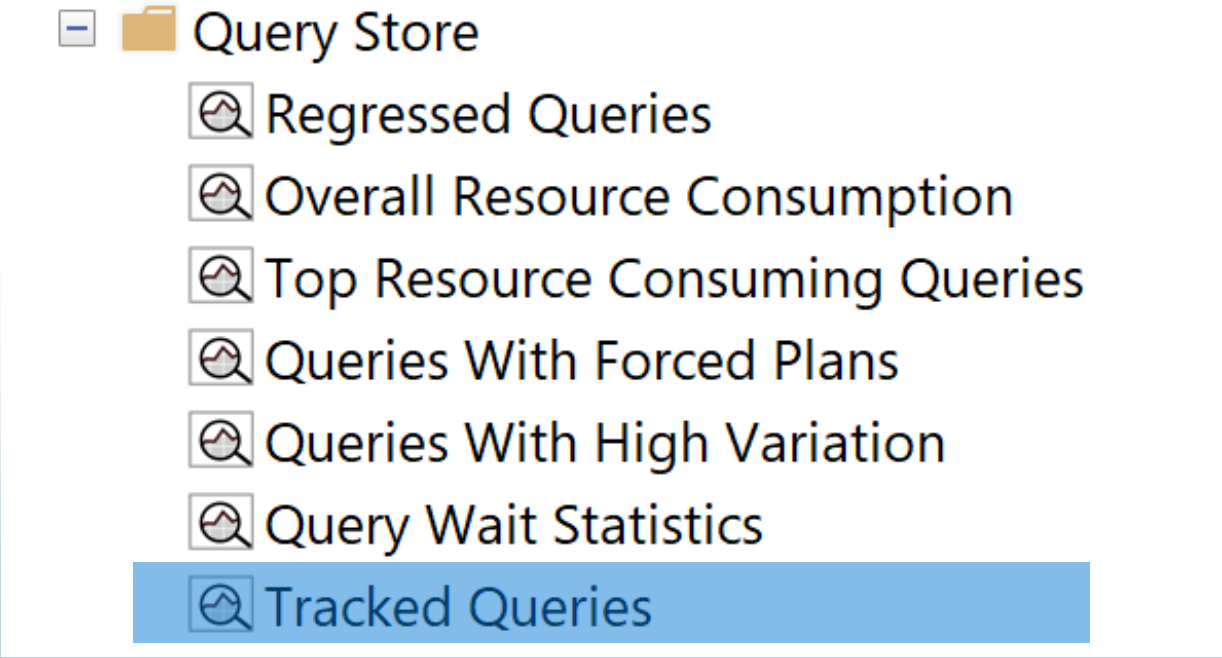
Query Store Reports

Query Wait Statistics shows a bar chart containing the top wait categories in the Query Store. Use the drop down at the top to select an aggregate criteria for the wait time: avg, max, min, std dev, and **total** (default). Requires SQL Server 2017.

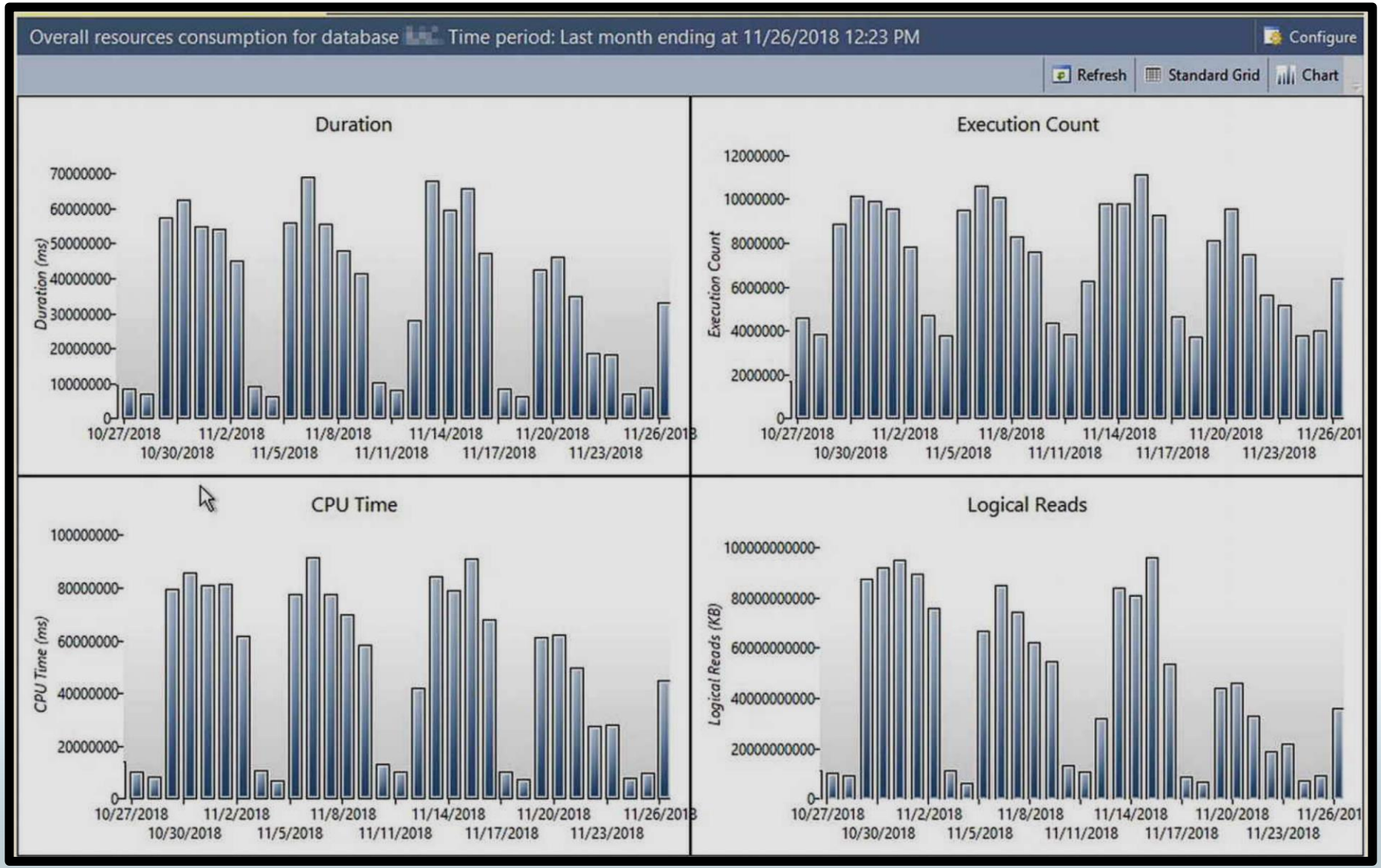


Query Store Reports

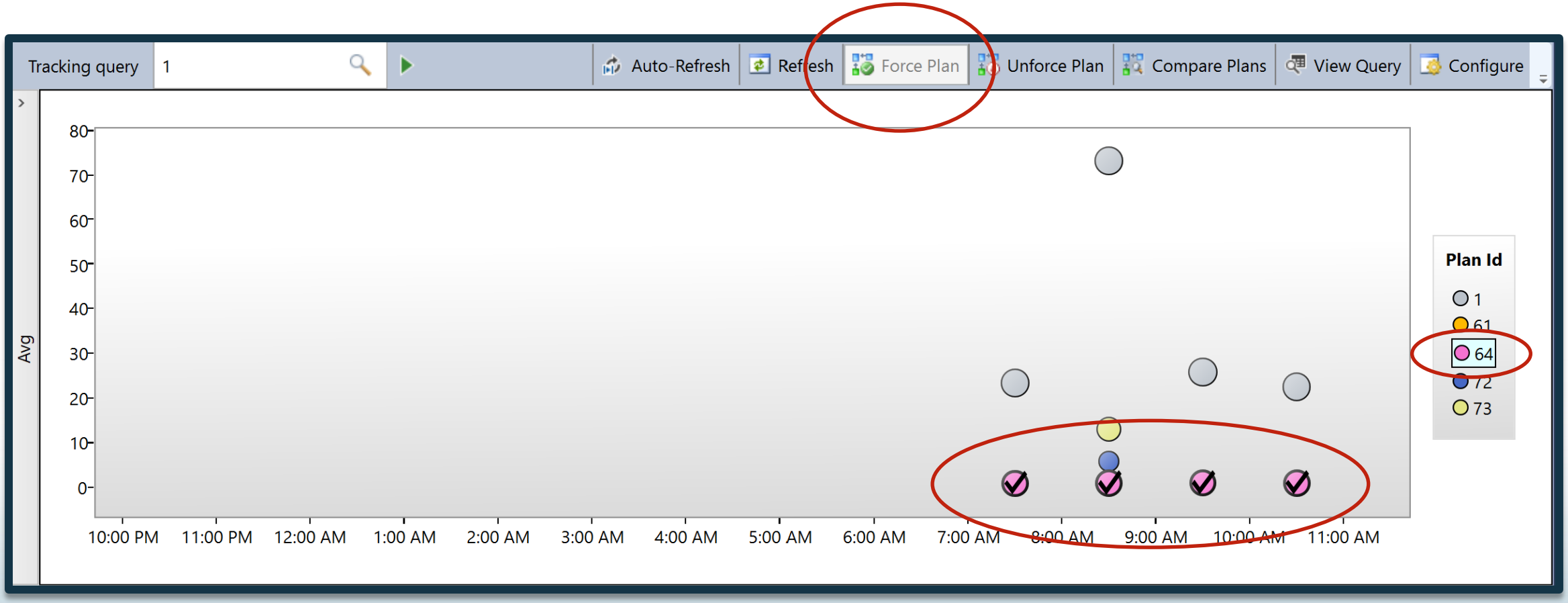
Tracked Queries: Use this dashboard to monitor the execution plans and regression of a specify query



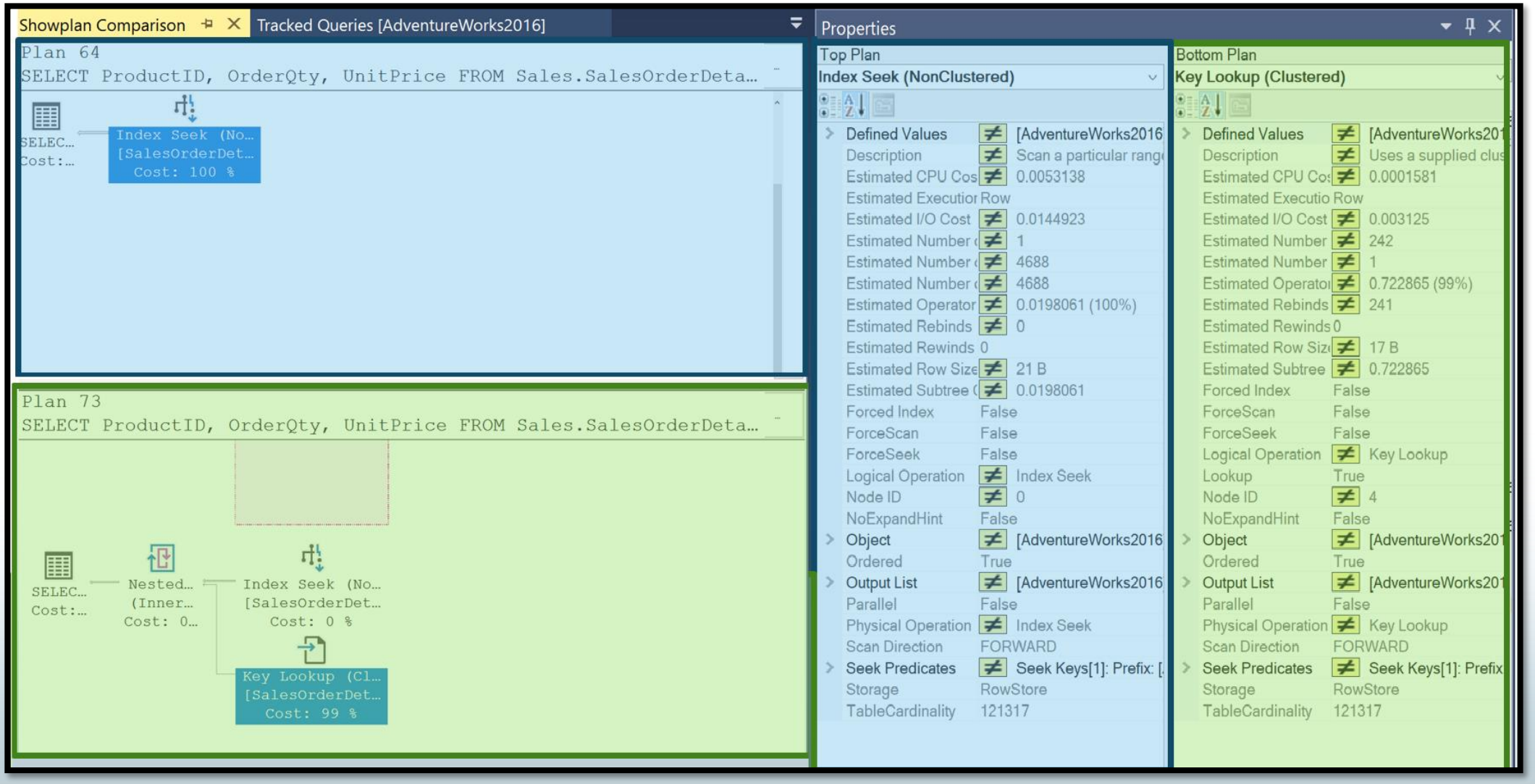
Establishing a Baseline



Force Plan


Tracking query
1
Auto-Refresh
Refresh
Force Plan
Unforce Plan
Compare Plans
View Query
Configure
Avg
80
70
60
50
40
30
20
10
0
10:00 PM
11:00 PM
12:00 AM
1:00 AM
2:00 AM
3:00 AM
4:00 AM
5:00 AM
6:00 AM
7:00 AM
8:00 AM
9:00 AM
10:00 AM
11:00 AM
Plan Id
1
61
64
72
73

Plan Compare



Questions?

Lesson 4

Query Store Catalog Views

Query Store contains three stores

Plan Store

- Persists execution plan information
- Stores Query Text and Query Plan information
- You can limit the number of unique plans by using the max_plans_per_query configuration.

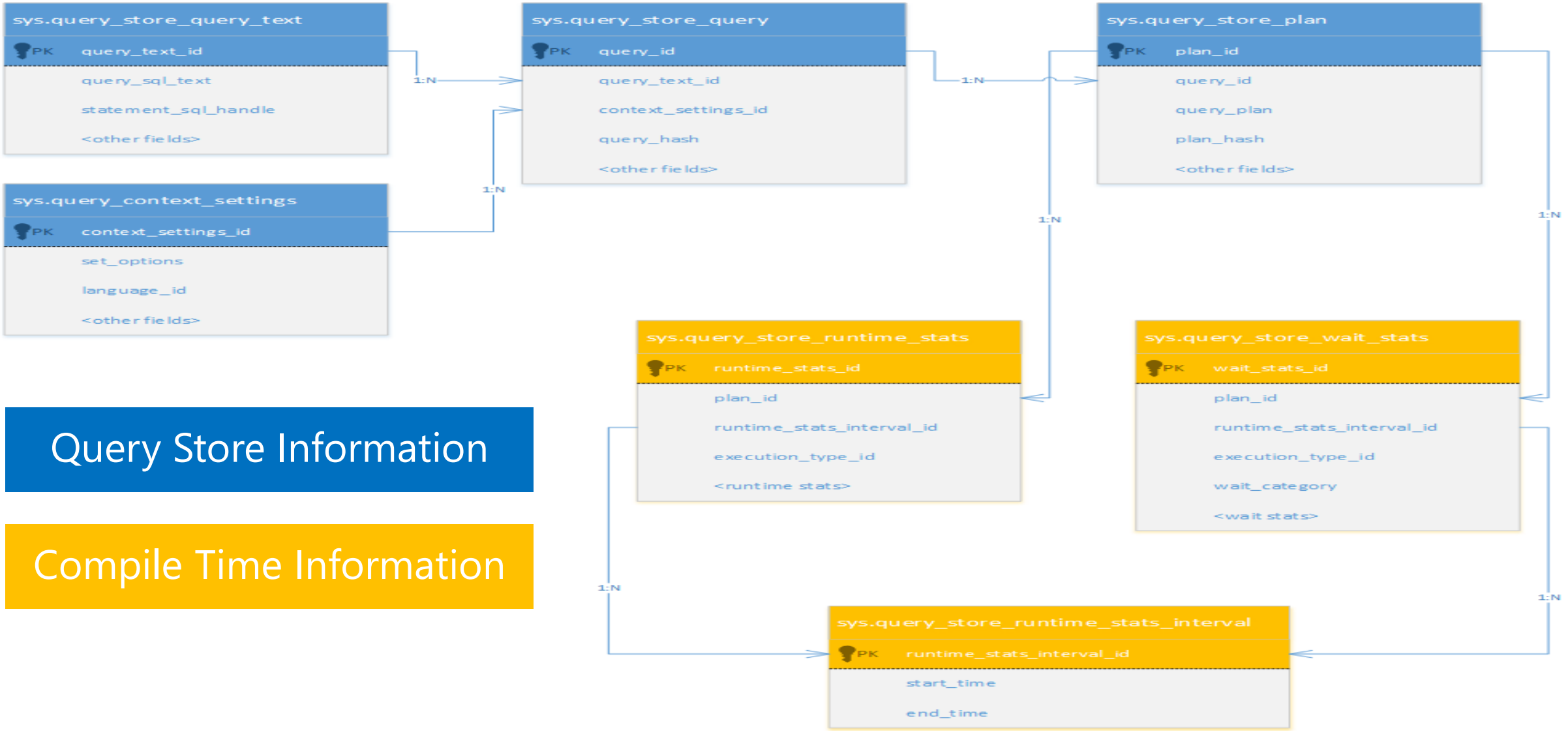
Runtime Stats Store

- Persists execution statistics information
- Captures Compile Time, Duration, CPU usage, Writes, and both Logical and Physical Reads
- To minimize space usage, the runtime stats are aggregated over a fixed time window.

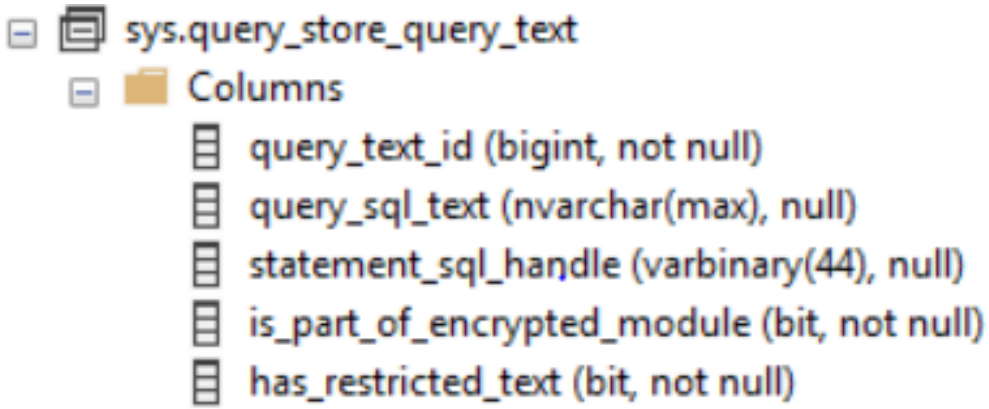
Wait Stat Store

- Persists wait statistics information
- Introduced in SQL Server 2017 and Azure SQL Database
- Organizes wait stats into different wait categories to simplify troubleshooting similar wait types.

Query Store Catalog Views



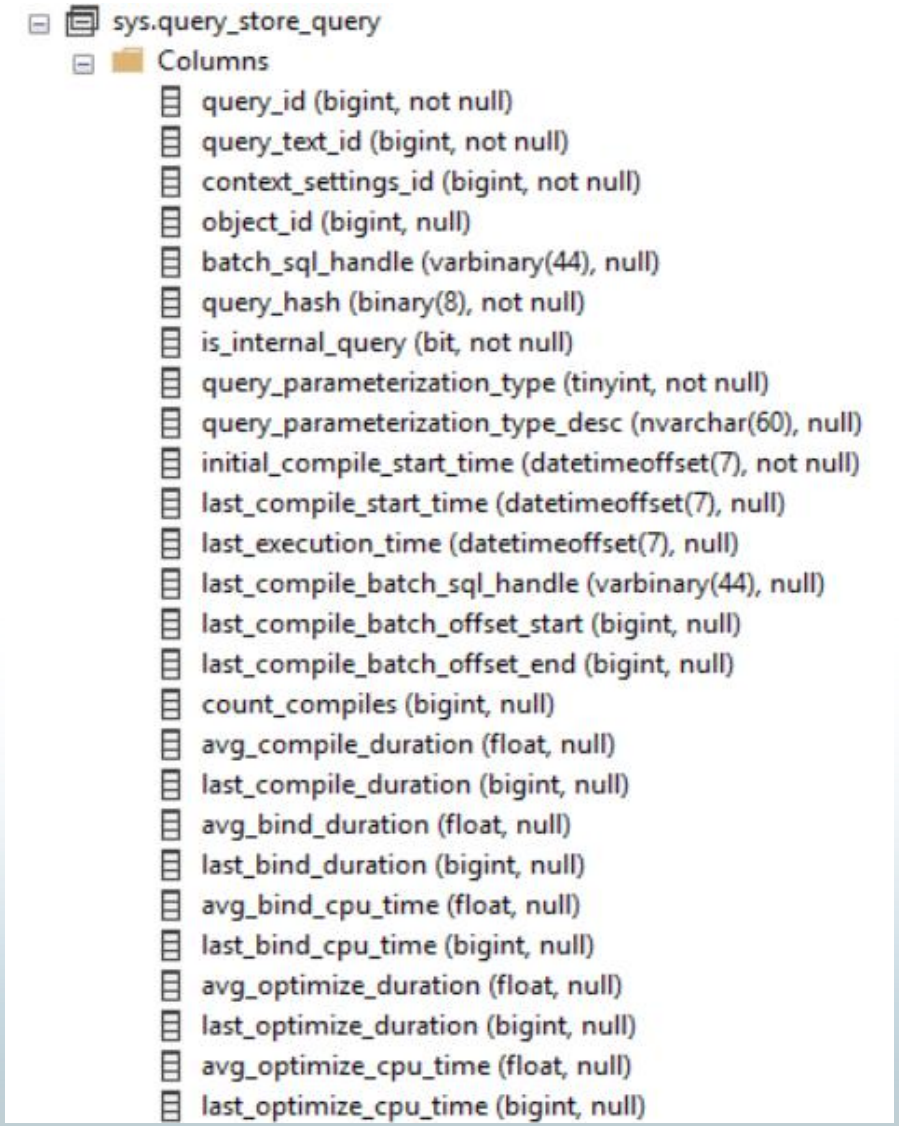
sys.query_store_query_text



Presents the unique query text that is executed against the database.

Every statement in a batch generates a separate query text entry.

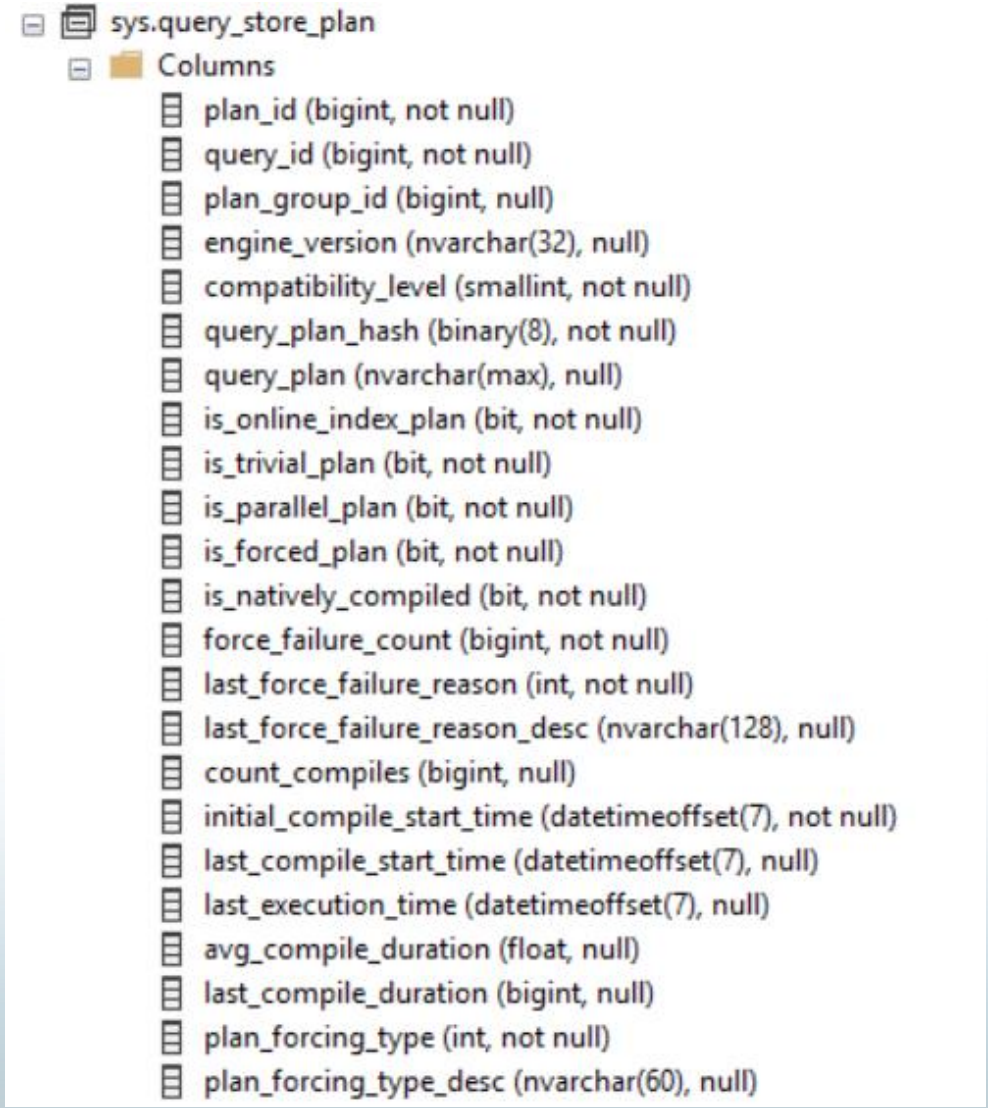
sys.query_store_query



Presents the query entries that are tracked and forced separately in the Query Store.

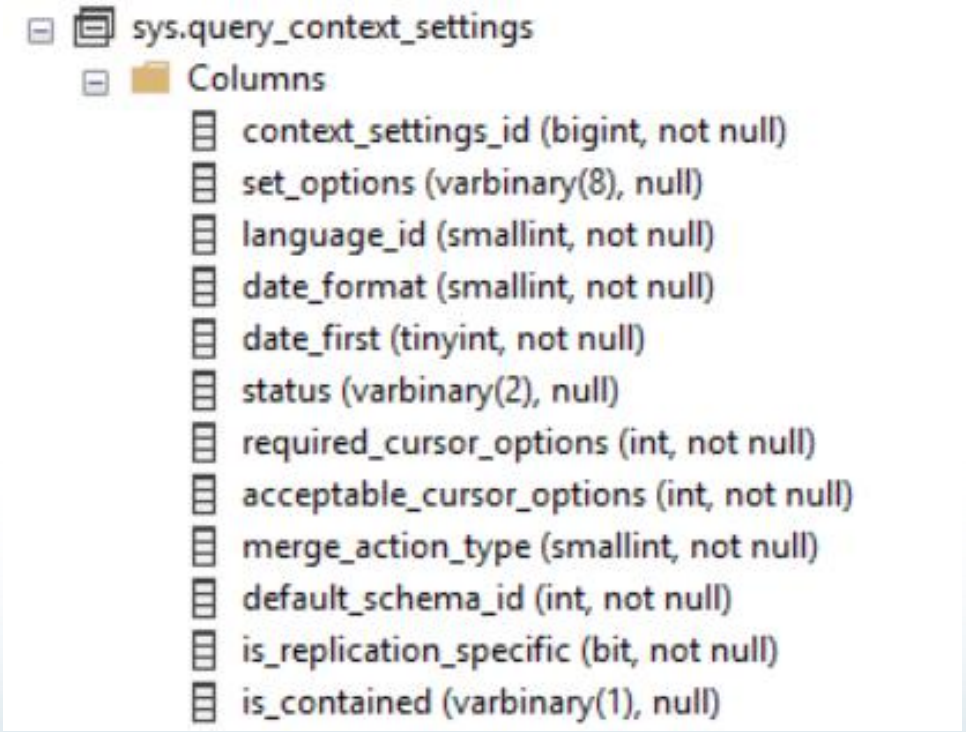
A single query text can produce multiple query entries that are executed under different context settings.

sys.query_store_plan



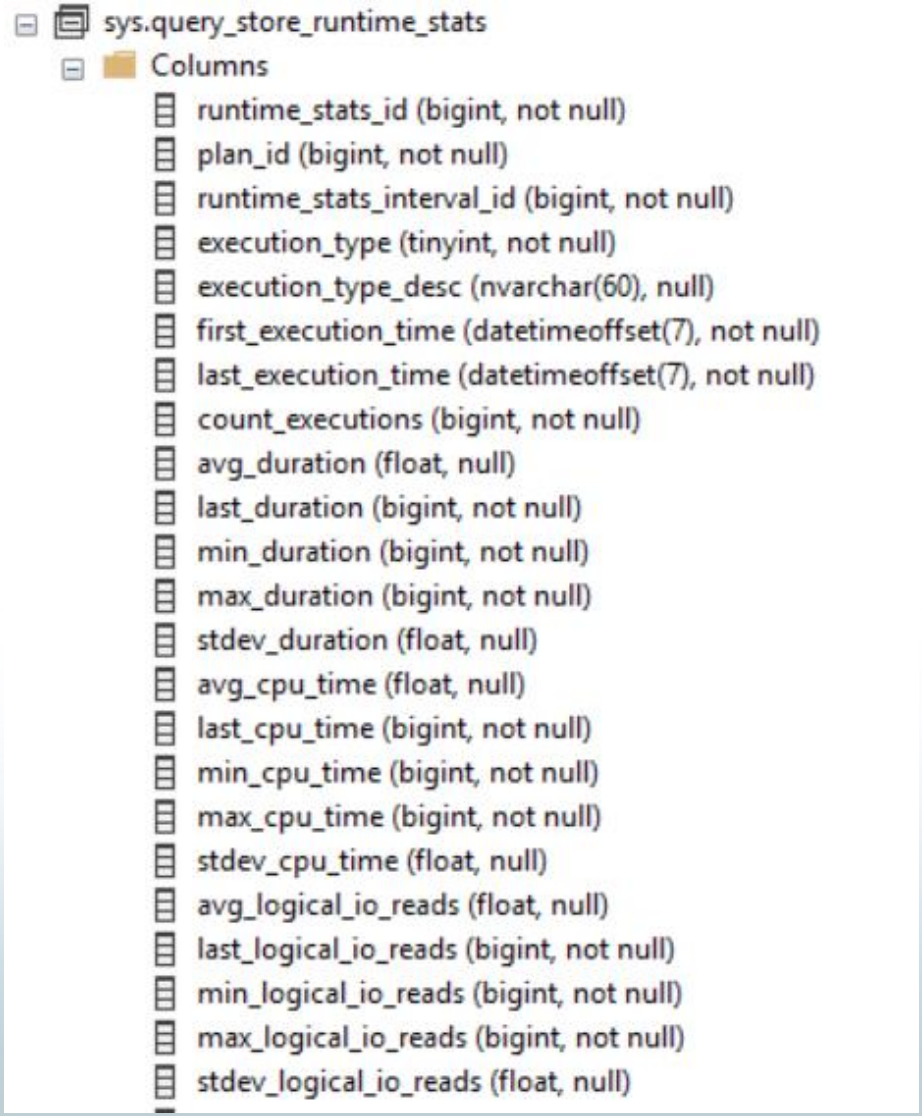
Presents the estimated plan for the query with the compile time statistics.

sys.query_context_settings



Presents the unique combinations of settings that affect a plan under which queries are executed.

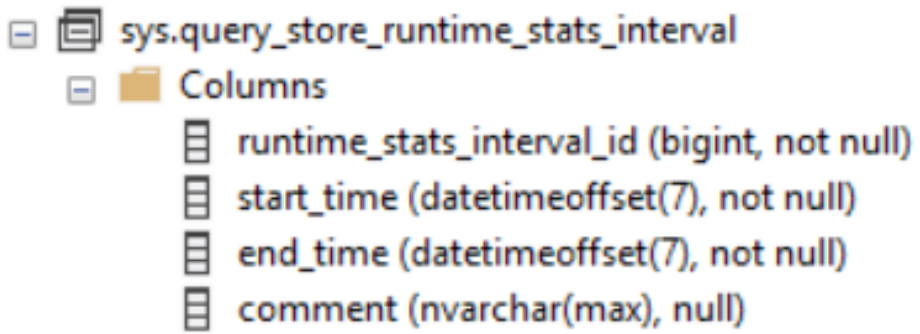
sys.query_store_runtime_stats



Aggregated runtime statistics for executed plans.

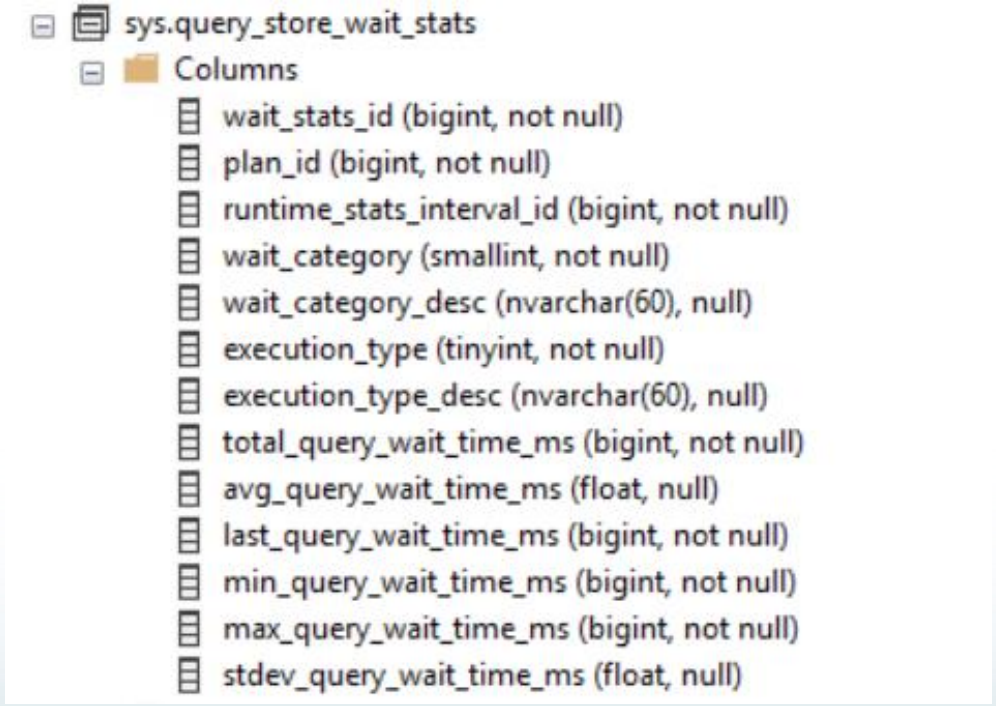
Metrics are expressed in the form of Four statistic functions. (Average, Minimum, Maximum, and Standard Deviation).

sys.query_store_runtime_stats_interval



Query Store divides time into automatically generated time windows (intervals) and stores aggregated statistics on that interval for every executed plan.

sys.query_store_plan_wait_stats



Presents wait category information for past runtime statistics interval.

Wait types are categorized and then wait time is aggregated across those wait categories.

Using Query Store Catalog Views

Finding the TOP 10 most frequently executed SQL Server Queries in the Query Store.

```
SELECT TOP 10 T.query_sql_text, Q.query_id,
    SUM(RS.count_executions) AS TotalExec
FROM sys.query_store_query_text as T
JOIN sys.query_store_query as Q
    ON T.query_text_id = Q.query_text_id
JOIN sys.query_store_plan AS P
    ON Q.query_id = P.query_id
JOIN sys.query_store_runtime_stats as RS
    ON P.plan_id = RS.plan_id
WHERE RS.count_executions > 1
GROUP BY T.query_sql_text, Q.query_id
ORDER By SUM(RS.count_executions)
```