**sys.dm\_exec\_query\_stats (Transact-SQL)**

* 12/17/2018
* 14 minutes to read

**APPLIES TO:** yesSQL Server (starting with 2008) yesAzure SQL Database noAzure SQL Data Warehouse noParallel Data Warehouse

Returns aggregate performance statistics for cached query plans in SQL Server. The view contains one row per query statement within the cached plan, and the lifetime of the rows are tied to the plan itself. When a plan is removed from the cache, the corresponding rows are eliminated from this view.

| **Column name** | **Data type** | **Description** |
| --- | --- | --- |
| **sql\_handle** | **varbinary(64)** | Is a token that refers to the batch or stored procedure that the query is part of.  **sql\_handle**, together with **statement\_start\_offset** and **statement\_end\_offset**, can be used to retrieve the SQL text of the query by calling the **sys.dm\_exec\_sql\_text** dynamic management function. |
| **statement\_start\_offset** | **int** | Indicates, in bytes, beginning with 0, the starting position of the query that the row describes within the text of its batch or persisted object. |
| **statement\_end\_offset** | **int** | Indicates, in bytes, starting with 0, the ending position of the query that the row describes within the text of its batch or persisted object. For versions before SQL Server 2014 (12.x), a value of -1 indicates the end of the batch. Trailing comments are no longer include. |
| **plan\_generation\_num** | **bigint** | A sequence number that can be used to distinguish between instances of plans after a recompile. |
| **plan\_handle** | **varbinary(64)** | A token that refers to the compiled plan that the query is part of. This value can be passed to the [sys.dm\_exec\_query\_plan](https://docs.microsoft.com/en-us/sql/relational-databases/system-dynamic-management-views/sys-dm-exec-query-plan-transact-sql?view=sql-server-2017) dynamic management function to obtain the query plan.  Will always be 0x000 when a natively compiled stored procedure queries a memory-optimized table. |
| **creation\_time** | **datetime** | Time at which the plan was compiled. |
| **last\_execution\_time** | **datetime** | Last time at which the plan started executing. |
| **execution\_count** | **bigint** | Number of times that the plan has been executed since it was last compiled. |
| **total\_worker\_time** | **bigint** | Total amount of CPU time, reported in microseconds (but only accurate to milliseconds), that was consumed by executions of this plan since it was compiled.  For natively compiled stored procedures, **total\_worker\_time** may not be accurate if many executions take less than 1 millisecond. |
| **last\_worker\_time** | **bigint** | CPU time, reported in microseconds (but only accurate to milliseconds), that was consumed the last time the plan was executed. 1 |
| **min\_worker\_time** | **bigint** | Minimum CPU time, reported in microseconds (but only accurate to milliseconds), that this plan has ever consumed during a single execution. 1 |
| **max\_worker\_time** | **bigint** | Maximum CPU time, reported in microseconds (but only accurate to milliseconds), that this plan has ever consumed during a single execution. 1 |
| **total\_physical\_reads** | **bigint** | Total number of physical reads performed by executions of this plan since it was compiled.  Will always be 0 querying a memory-optimized table. |
| **last\_physical\_reads** | **bigint** | Number of physical reads performed the last time the plan was executed.  Will always be 0 querying a memory-optimized table. |
| **min\_physical\_reads** | **bigint** | Minimum number of physical reads that this plan has ever performed during a single execution.  Will always be 0 querying a memory-optimized table. |
| **max\_physical\_reads** | **bigint** | Maximum number of physical reads that this plan has ever performed during a single execution.  Will always be 0 querying a memory-optimized table. |
| **total\_logical\_writes** | **bigint** | Total number of logical writes performed by executions of this plan since it was compiled.  Will always be 0 querying a memory-optimized table. |
| **last\_logical\_writes** | **bigint** | Number of buffer pool pages dirtied during the most recently completed execution of the plan.  After a page is read, the page becomes dirty only the first time it is modified. When a page becomes dirty, this number is incremented. Subsequent modifications of an already dirty page do not affect this number.  This number will always be 0 when querying a memory-optimized table. |
| **min\_logical\_writes** | **bigint** | Minimum number of logical writes that this plan has ever performed during a single execution.  Will always be 0 querying a memory-optimized table. |
| **max\_logical\_writes** | **bigint** | Maximum number of logical writes that this plan has ever performed during a single execution.  Will always be 0 querying a memory-optimized table. |
| **total\_logical\_reads** | **bigint** | Total number of logical reads performed by executions of this plan since it was compiled.  Will always be 0 querying a memory-optimized table. |
| **last\_logical\_reads** | **bigint** | Number of logical reads performed the last time the plan was executed.  Will always be 0 querying a memory-optimized table. |
| **min\_logical\_reads** | **bigint** | Minimum number of logical reads that this plan has ever performed during a single execution.  Will always be 0 querying a memory-optimized table. |
| **max\_logical\_reads** | **bigint** | Maximum number of logical reads that this plan has ever performed during a single execution.  Will always be 0 querying a memory-optimized table. |
| **total\_clr\_time** | **bigint** | Time, reported in microseconds (but only accurate to milliseconds), consumed inside Microsoft .NET Framework common language runtime (CLR) objects by executions of this plan since it was compiled. The CLR objects can be stored procedures, functions, triggers, types, and aggregates. |
| **last\_clr\_time** | **bigint** | Time, reported in microseconds (but only accurate to milliseconds) consumed by execution inside .NET Framework CLR objects during the last execution of this plan. The CLR objects can be stored procedures, functions, triggers, types, and aggregates. |
| **min\_clr\_time** | **bigint** | Minimum time, reported in microseconds (but only accurate to milliseconds), that this plan has ever consumed inside .NET Framework CLR objects during a single execution. The CLR objects can be stored procedures, functions, triggers, types, and aggregates. |
| **max\_clr\_time** | **bigint** | Maximum time, reported in microseconds (but only accurate to milliseconds), that this plan has ever consumed inside the .NET Framework CLR during a single execution. The CLR objects can be stored procedures, functions, triggers, types, and aggregates. |
| **total\_elapsed\_time** | **bigint** | Total elapsed time, reported in microseconds (but only accurate to milliseconds), for completed executions of this plan. |
| **last\_elapsed\_time** | **bigint** | Elapsed time, reported in microseconds (but only accurate to milliseconds), for the most recently completed execution of this plan. |
| **min\_elapsed\_time** | **bigint** | Minimum elapsed time, reported in microseconds (but only accurate to milliseconds), for any completed execution of this plan. |
| **max\_elapsed\_time** | **bigint** | Maximum elapsed time, reported in microseconds (but only accurate to milliseconds), for any completed execution of this plan. |
| **query\_hash** | **Binary(8)** | Binary hash value calculated on the query and used to identify queries with similar logic. You can use the query hash to determine the aggregate resource usage for queries that differ only by literal values. |
| **query\_plan\_hash** | **binary(8)** | Binary hash value calculated on the query execution plan and used to identify similar query execution plans. You can use query plan hash to find the cumulative cost of queries with similar execution plans.  Will always be 0x000 when a natively compiled stored procedure queries a memory-optimized table. |
| **total\_rows** | **bigint** | Total number of rows returned by the query. Cannot be null.  Will always be 0 when a natively compiled stored procedure queries a memory-optimized table. |
| **last\_rows** | **bigint** | Number of rows returned by the last execution of the query. Cannot be null.  Will always be 0 when a natively compiled stored procedure queries a memory-optimized table. |
| **min\_rows** | **bigint** | Minimum number of rows ever returned by the query during one execution. Cannot be null.  Will always be 0 when a natively compiled stored procedure queries a memory-optimized table. |
| **max\_rows** | **bigint** | Maximum number of rows ever returned by the query during one execution. Cannot be null.  Will always be 0 when a natively compiled stored procedure queries a memory-optimized table. |
| **statement\_sql\_handle** | **varbinary(64)** | **Applies to**: SQL Server 2014 (12.x) through SQL Server 2017.  Populated with non-NULL values only if Query Store is turned on and collecting the stats for that particular query. |
| **statement\_context\_id** | **bigint** | **Applies to**: SQL Server 2014 (12.x) through SQL Server 2017.  Populated with non-NULL values only if Query Store is turned on and collecting the stats for that particular query. |
| **total\_dop** | **bigint** | The total sum of degree of parallelism this plan used since it was compiled. It will always be 0 for querying a memory-optimized table.  **Applies to**: SQL Server 2016 (13.x) through SQL Server 2017. |
| **last\_dop** | **bigint** | The degree of parallelism when this plan executed last time. It will always be 0 for querying a memory-optimized table.  **Applies to**: SQL Server 2016 (13.x) through SQL Server 2017. |
| **min\_dop** | **bigint** | The minimum degree of parallelism this plan ever used during one execution. It will always be 0 for querying a memory-optimized table.  **Applies to**: SQL Server 2016 (13.x) through SQL Server 2017. |
| **max\_dop** | **bigint** | The maximum degree of parallelism this plan ever used during one execution. It will always be 0 for querying a memory-optimized table.  **Applies to**: SQL Server 2016 (13.x) through SQL Server 2017. |
| **total\_grant\_kb** | **bigint** | The total amount of reserved memory grant in KB this plan received since it was compiled. It will always be 0 for querying a memory-optimized table.  **Applies to**: SQL Server 2016 (13.x) through SQL Server 2017. |
| **last\_grant\_kb** | **bigint** | The amount of reserved memory grant in KB when this plan executed last time. It will always be 0 for querying a memory-optimized table.  **Applies to**: SQL Server 2016 (13.x) through SQL Server 2017. |
| **min\_grant\_kb** | **bigint** | The minimum amount of reserved memory grant in KB this plan ever received during one execution. It will always be 0 for querying a memory-optimized table.  **Applies to**: SQL Server 2016 (13.x) through SQL Server 2017. |
| **max\_grant\_kb** | **bigint** | The maximum amount of reserved memory grant in KB this plan ever received during one execution. It will always be 0 for querying a memory-optimized table.  **Applies to**: SQL Server 2016 (13.x) through SQL Server 2017. |
| **total\_used\_grant\_kb** | **bigint** | The total amount of reserved memory grant in KB this plan used since it was compiled. It will always be 0 for querying a memory-optimized table.  **Applies to**: SQL Server 2016 (13.x) through SQL Server 2017. |
| **last\_used\_grant\_kb** | **bigint** | The amount of used memory grant in KB when this plan executed last time. It will always be 0 for querying a memory-optimized table.  **Applies to**: SQL Server 2016 (13.x) through SQL Server 2017. |
| **min\_used\_grant\_kb** | **bigint** | The minimum amount of used memory grant in KB this plan ever used during one execution. It will always be 0 for querying a memory-optimized table.  **Applies to**: SQL Server 2016 (13.x) through SQL Server 2017. |
| **max\_used\_grant\_kb** | **bigint** | The maximum amount of used memory grant in KB this plan ever used during one execution. It will always be 0 for querying a memory-optimized table.  **Applies to**: SQL Server 2016 (13.x) through SQL Server 2017. |
| **total\_ideal\_grant\_kb** | **bigint** | The total amount of ideal memory grant in KB this plan estimated since it was compiled. It will always be 0 for querying a memory-optimized table.  **Applies to**: SQL Server 2016 (13.x) through SQL Server 2017. |
| **last\_ideal\_grant\_kb** | **bigint** | The amount of ideal memory grant in KB when this plan executed last time. It will always be 0 for querying a memory-optimized table.  **Applies to**: SQL Server 2016 (13.x) through SQL Server 2017. |
| **min\_ideal\_grant\_kb** | **bigint** | The minimum amount of ideal memory grant in KB this plan ever estimated during one execution. It will always be 0 for querying a memory-optimized table.  **Applies to**: SQL Server 2016 (13.x) through SQL Server 2017. |
| **max\_ideal\_grant\_kb** | **bigint** | The maximum amount of ideal memory grant in KB this plan ever estimated during one execution. It will always be 0 for querying a memory-optimized table.  **Applies to**: SQL Server 2016 (13.x) through SQL Server 2017. |
| **total\_reserved\_threads** | **bigint** | The total sum of reserved parallel threads this plan ever used since it was compiled. It will always be 0 for querying a memory-optimized table.  **Applies to**: SQL Server 2016 (13.x) through SQL Server 2017. |
| **last\_reserved\_threads** | **bigint** | The number of reserved parallel threads when this plan executed last time. It will always be 0 for querying a memory-optimized table.  **Applies to**: SQL Server 2016 (13.x) through SQL Server 2017. |
| **min\_reserved\_threads** | **bigint** | The minimum number of reserved parallel threads this plan ever used during one execution. It will always be 0 for querying a memory-optimized table.  **Applies to**: SQL Server 2016 (13.x) through SQL Server 2017. |
| **max\_reserved\_threads** | **bigint** | The maximum number of reserved parallel threads this plan ever used during one execution. It will always be 0 for querying a memory-optimized table.  **Applies to**: SQL Server 2016 (13.x) through SQL Server 2017. |
| **total\_used\_threads** | **bigint** | The total sum of used parallel threads this plan ever used since it was compiled. It will always be 0 for querying a memory-optimized table.  **Applies to**: SQL Server 2016 (13.x) through SQL Server 2017. |
| **last\_used\_threads** | **bigint** | The number of used parallel threads when this plan executed last time. It will always be 0 for querying a memory-optimized table.  **Applies to**: SQL Server 2016 (13.x) through SQL Server 2017. |
| **min\_used\_threads** | **bigint** | The minimum number of used parallel threads this plan ever used during one execution. It will always be 0 for querying a memory-optimized table.  **Applies to**: SQL Server 2016 (13.x) through SQL Server 2017. |
| **max\_used\_threads** | **bigint** | The maximum number of used parallel threads this plan ever used during one execution. It will always be 0 for querying a memory-optimized table.  **Applies to**: SQL Server 2016 (13.x) through SQL Server 2017. |
| **total\_columnstore\_segment\_reads** | **bigint** | The total sum of columnstore segments read by the query. Cannot be null.  **Applies to**: Starting with SQL Server 2016 (13.x) SP2 and SQL Server 2017 (14.x) CU3 |
| **last\_columnstore\_segment\_reads** | **bigint** | The number of columnstore segments read by the last execution of the query. Cannot be null.  **Applies to**: Starting with SQL Server 2016 (13.x) SP2 and SQL Server 2017 (14.x) CU3 |
| **min\_columnstore\_segment\_reads** | **bigint** | The minimum number of columnstore segments ever read by the query during one execution. Cannot be null.  **Applies to**: Starting with SQL Server 2016 (13.x) SP2 and SQL Server 2017 (14.x) CU3 |
| **max\_columnstore\_segment\_reads** | **bigint** | The maximum number of columnstore segments ever read by the query during one execution. Cannot be null.  **Applies to**: Starting with SQL Server 2016 (13.x) SP2 and SQL Server 2017 (14.x) CU3 |
| **total\_columnstore\_segment\_skips** | **bigint** | The total sum of columnstore segments skipped by the query. Cannot be null.  **Applies to**: Starting with SQL Server 2016 (13.x) SP2 and SQL Server 2017 (14.x) CU3 |
| **last\_columnstore\_segment\_skips** | **bigint** | The number of columnstore segments skipped by the last execution of the query. Cannot be null.  **Applies to**: Starting with SQL Server 2016 (13.x) SP2 and SQL Server 2017 (14.x) CU3 |
| **min\_columnstore\_segment\_skips** | **bigint** | The minimum number of columnstore segments ever skipped by the query during one execution. Cannot be null.  **Applies to**: Starting with SQL Server 2016 (13.x) SP2 and SQL Server 2017 (14.x) CU3 |
| **max\_columnstore\_segment\_skips** | **bigint** | The maximum number of columnstore segments ever skipped by the query during one execution. Cannot be null.  **Applies to**: Starting with SQL Server 2016 (13.x) SP2 and SQL Server 2017 (14.x) CU3 |
| **total\_spills** | **bigint** | The total number of pages spilled by execution of this query since it was compiled.  **Applies to**: Starting with SQL Server 2016 (13.x) SP2 and SQL Server 2017 (14.x) CU3 |
| **last\_spills** | **bigint** | The number of pages spilled the last time the query was executed.  **Applies to**: Starting with SQL Server 2016 (13.x) SP2 and SQL Server 2017 (14.x) CU3 |
| **min\_spills** | **bigint** | The minimum number of pages that this query has ever spilled during a single execution.  **Applies to**: Starting with SQL Server 2016 (13.x) SP2 and SQL Server 2017 (14.x) CU3 |
| **max\_spills** | **bigint** | The maximum number of pages that this query has ever spilled during a single execution.  **Applies to**: Starting with SQL Server 2016 (13.x) SP2 and SQL Server 2017 (14.x) CU3 |
| **pdw\_node\_id** | **int** | The identifier for the node that this distribution is on.  **Applies to**: Azure SQL Data Warehouse, Parallel Data Warehouse |