MonWiQdsExecStats (QDS Execution Stats)

Last updated by | Radhika Shah | May 31, 2022 at 7:01 AM PDT

Contents

Friday, November 10, 2017

6:28 PM

Type of data

Execution statistics per query plan over time

Included info

- Execution count
- Execution type (0 - regular, 3 - aborted, 4 - exception)
- Statement type (select/insert/update/delete/...)
- CPU time (in microseconds)
- Elapsed time (in microseconds)
- DOP
- Memory (in number of 8KB pages)
- Logical reads (in number of 8KB pages)
- Logical writes
- Physical reads (in number of 8KB pages)
- Row count
- CLR time (in microseconds)
- Log bytes used
- TempDB space used

Notable columns to summarize by

- originalEventTimestamp/TIMESTAMP

- Query_id - id of the query in QDS (specific to database)
- Plan_id - id of the plan in QDS (specific to database)
- Query_hash - represents the hash value of the query shape
- Query_plan_hash - represents the hash value of the plan shape
- Statement_sql_hash - represents the hash value of the query text

For each metric the following data is available:

- Total over the exhaust interval (usually 15 minutes)
  - Average could be calculated as total / execution_count
- Minimum over the aggregation interval (usually the full hour)
- Maximum over the aggregation interval (usually the full hour)
- Sum of squares over the exhaust interval (usually 15 minutes)

For more details on how to interpret this data and the execution times, please refer to TRQDS0002: Using Query Store (QDS) data exhaust

Sample queries

Get top 10 queries by total CPU time over the past 7 days:

```
MonWiQdsExecStats | where LogicalServerName == "nc-bacmpshardprod9" and database_name ==
"Campaign_MainShard_P124" | where originalEventTimestamp > now(-7d) | summarize sum(execution_count),
sum(cpu_time), dcount(plan_id) by query_id | top 10 by sum_cpu_time desc | extend sum_cpu_time_milliseconds
= sum_cpu_time / 1000 | extend avg_cpu_time_milliseconds = 1.0 * sum_cpu_time_milliseconds /
sum_execution_count | extend distinct_plans = dcount_plan_id | project query_id, distinct_plans,
sum_execution_count, avg_cpu_time_milliseconds, sum_cpu_time_milliseconds
```

query_id	distinct_plans	sum_execution_count	avg_cpu_time_milliseconds	sum_cpu_time_milliseconds
14845172	2	212281	140.869889438998	29904001
13697	2	185260	52.671764007341	9757971
14845393	1	4147	831.964793826863	3450158
14835127	46	9914	308.085333871293	3054358
12774	2	1841622	1.64925592765508	3037306
14868893	2	55852	50.8963510706868	2842663
14978611	1	10086808	0.206022460227259	2078109
15020757	1	9928	202.885676873489	2014249
12841	1	10075256	0.190791578893876	1922274
14851173	29	9921	193.161374861405	1916354

Compare the average CPU time for all plans of a query

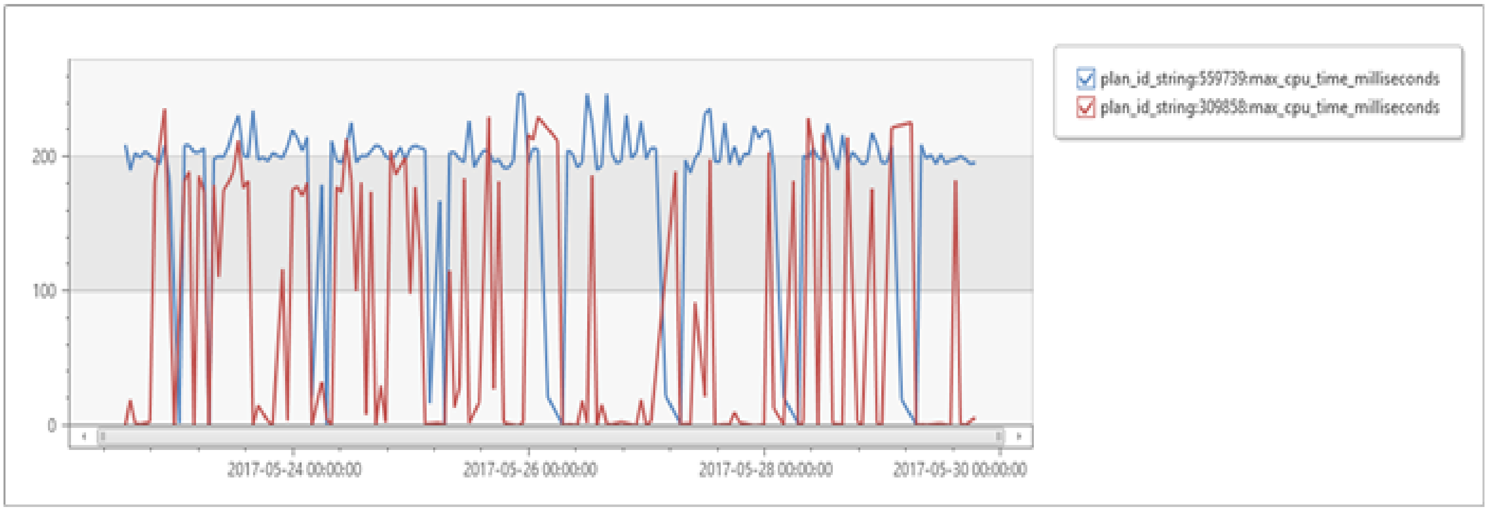
```
MonWiQdsExecStats | where LogicalServerName == "nc-bacmpshardprod9" and database_name ==
"Campaign_MainShard_P124" | where query_id == 14845172 | summarize sum(execution_count), sum(cpu_time)
by plan_id | extend avg_cpu_time_milliseconds = 1.0 * sum_cpu_time / sum_execution_count / 1000 | project
plan_id, sum_execution_count, avg

_cpu_time_milliseconds
```

plan_id	sum_execution_count	avg_cpu_time_milliseconds
559739	205177	141.518825716333
309858	7104	122.127506897523

Compare the maximum CPU time for all plans of a query over 1-hour intervals

```
MonWiQdsExecStats | where LogicalServerName == "nc-bacmpshardprod9" and database_name ==
"Campaign_MainShard_P124" | where query_id == 14845172 | extend plan_id_string = tostring(plan_id) |
summarize max(max_cpu_time) by bin(TIMESTAMP, 1h), plan_id_string | extend max_cpu_time_milliseconds =
1.0 * max_max_cpu_time / 1000 | project TIMESTAMP, max_cpu_time_milliseconds, plan_id_string | render
timechart
```



How good have you found this content?