ComparingPerformance_betweenenvironments

Last updated by | Joao Antunes | Dec 2, 2022 at 5:17 AM PST

Contents

Issue

A user may report issue that different environments with identical workloads, schema, data and configuration are running differently. A typical example is that a query runs faster on one environment but slower on secondary or vice versa. This TSG defines steps to collect information and basic analysis for this type of analysis.

Investigation/Analysis

If the issue is specific to a query and you are looking to capture information relative to that please follow

Step 1: Compare using ASC.

Run ASC for both the environments and fo to SQL Troubleshooter >>> Performance TAB:

1. Make sure that the code package version under Database version change history is same for both the enviromnets.
2. Compare the resource utilization. If there is a difference in utilization it will help you determine where the specific issue is (CPU\IO\Memory etc).
3. Check waits under top waiting queries and you should be able to identify the top slow queries.

Step 2: Compare workload for a particular timeframe using Kusto:

Identify if there is any difference in exectuion counts.

```
MonWiQdsExecStats
| where (LogicalServerName =~ '' and database_name =~ '' )
| where TIMESTAMP < datetime() and TIMESTAMP > datetime()
| summarize sum(execution_count) by query_plan_hash
| order by sum_execution_count desc
```

Step 3: Compare environments.

1. Indexes: You can compare indexes in both the evironemtns from adhocquerytobackendinstance.xts view or sys.indexes from customer end.
2. Compare query plans and identify if there is difference in specific qurey plans for slow running query.
3. Check if auditing is enabled in either of the environments. If auditing is enabled only on one environment, it could be reson for slowness.

Investigating further

CPU Issues

In case of CPU issues please proceed with this TSG Workflow for High CPU troubleshooting

IO Issues

In case of IO issues please proceed with this TSG Workflow-for-IO-troubleshooting

For Query tunning

In case of IO issues please proceed with this TSG Workflow-for-query-tunning

Mitigation

At this point, if you are still not able to identify the difference please engage PG: SQL DB Perf : HighDtu, SlowQuery, HighWaits, unexpectedSQLBehavior.

Internal Links

192072515

Root Cause Classification

Cases resolved by this TSG should be coded to the following root cause: Workload performance/User issue/error/Specific query slow

How good have you found this content?