Unexpected increase in resource consumption or DTUs

Last updated by | Amie Coleman | Oct 5, 2022 at 9:02 AM PDT

Contents

- Issue
- Investigation/Analysis
  - Troubleshoot using ASC
  - Additional Checks
  - Best Practices
- Mitigation Steps
- Public Doc Reference
- Internal Reference
- Root Cause Classification

Issue

Customers may report an unexpected increase in resource utilisation for their database. There are multiple resources and tools available for the customer to monitor usage.
Monitoring Azure SQL Database

Investigation/Analysis

NOTE: If the Service Level Objective (SLO) of the database is equal to S2 or lower, it is considered a Sub-Core SLO. The Basic, S0, S1 and S2 service objectives provide less than one vCore (CPU). For CPU-intensive workloads, a service objective of S3 or greater is recommended. The best solution is to ask customer to scale up, see Sub-Core SLO.

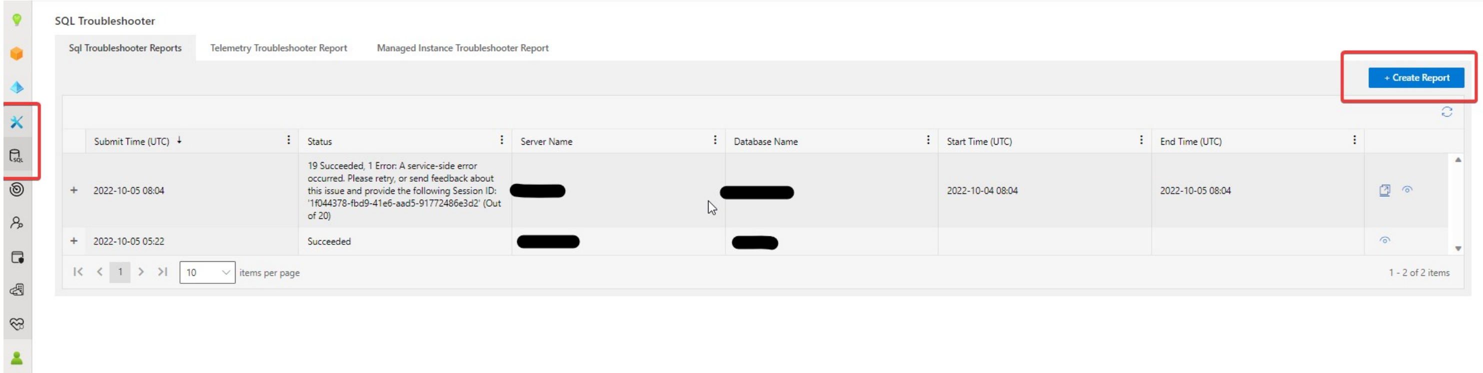
Unexpected increase in resource consumption scoping:

1. Server Name/Database Name/Elastic Pool
2. Resource SLO
3. Determine an accurate timeframe (start time and end time, unless the issue is on-going). An accurate timeframe can be used to compare before and after usage to understand what is a "normal" workload. Determine a workload baseline
4. Utilise telemetry in ASC and work with the customer to understand if there were any changes around the start time of the problem (workload increases or changes, query/code changes, SLO changes, database configuration changes)

Troubleshoot using ASC

Leverage ASC using the following reports:

- Generate Sql Troubleshooter reports to capture telemetry before/after issue (ASC > Tools > Sql Troubleshooter > Create Report)



SQL Troubleshooter
Sql Troubleshooter Reports
Telemetry Troubleshooter Report
Managed Instance Troubleshooter Report
+ Create Report

	Submit Time (UTC)	Status	Server Name	Database Name	Start Time (UTC)	End Time (UTC)
+	2022-10-05 08:04	19 Succeeded, 1 Error: A service-side error occurred. Please retry, or send feedback about this issue and provide the following Session ID: '1f044378-fbd9-41e6-aad5-91772486e3d2' (Out of 20)	[redacted]	[redacted]	2022-10-04 08:04	2022-10-05 08:04
+	2022-10-05 05:22	Succeeded	[redacted]	[redacted]		

- Review the generated Insights, focusing on those that != AllClear (Select Report > Performance > Insights)
- Check the Workload details to identify resource throttling, including Resource Utilisation patterns, Query Execution count, Database Wait Statistics (Performance > Overview/Last 7 day Overview)
- If you cannot identify any change in Query Execution Count/Workload increase that might explain the unexpected resource consumption, continue to review individual Queries and review for any changes in executions (Performance > Queries)
- Check for any Long Running Transactions and/or Blocking occurrences during the time frame (Performance > Blocking and Deadlocking)
- Check for any Configuration Changes before or during the time frame (Performance > Config & Change History)

Additional Checks

Review further performance details:
For standalone databases: Performance Stand Alone DB's
For elastic pool databases: Performance: Elastic Pool's

Capturing logs:
PerfStats: Collecting Perf Stats
Xevents: Collecting XEvents

Best Practices

Ensure customer is performing regular database maintenance tasks, including:

1. Updating Statistics
2. Index maintenance

We can provide scripts to help the customer perform regular maintenance activities:
Maintenance solution
Automating Maintenance solution

Mitigation Steps

There are a few scenarios that can cause an unexpected increase in resource consumption, therefore, it depends on the outcome of the investigation;

1. For scenarios where you find that the overall workload and query execution has increased, then the customer can look at increasing the Service Level Objective (SLO) of the database to accomodate the higher workload.
2. If there was a specific query that regressed and consumed additional resources, focus on tuning the query (review query execution, check for execution plan changes, wait statistics, capture an actual execution plan).
3. Implement best practice recommendations to ensure that statistics and indexes are maintained regularly.
4. If you found that there were any ad-hoc long running queries during the time frame that caused the increase in resource consumption, highlight these to the customer for review (is it expected for that query to run for extended periods of time, if it used to run faster then refer to point 2 for query tuning).

Public Doc Reference

Monitoring Azure SQL Database
Sub-Core SLO
Maintain Statistics and Indexes
Automate Statistics and Index Maintenance Task

Internal Reference

Sub-Core SLO
Index and Statistics Maintenance

Root Cause Classification

Cases resolved by this TSG should be coded to the following root cause:
Root Cause: Azure SQL DB v2\Workload Performance\User Issue/Error\DTU Limit\CPU

How good have you found this content?

🙂 🙁