Microsoft

Monitoring and Tuning Azure SQL Database

Module 6

Learning Units covered in this Module

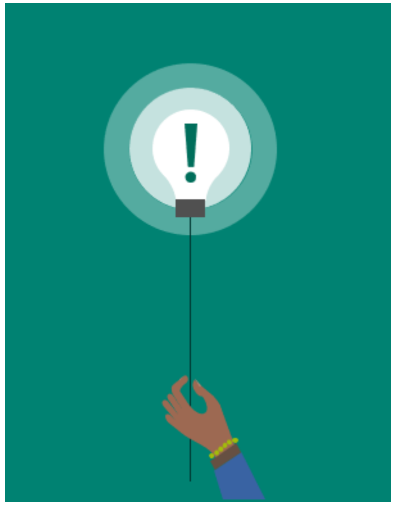
- Lesson 1: Monitoring and Troubleshooting Azure SQL Database
- Lesson 2: Configure Alerts through Azure Portal
- Lesson 3: Monitoring Query Performance using Query Performance Insight
- Lesson 4: Azure SQL Database Tuning using Automatic Tuning
- Lesson 5: Monitoring Azure SQL Database Performance using Database Watcher

Lesson 1: Monitoring and Troubleshooting Azure SQL Database

Objectives

After completing this learning, you will be able to:

- Know the various options to monitor and troubleshoot the Azure SQL Database.



Common Issues on Azure SQL Database

Monitoring for Azure SQL Database is scoped at database level.

Here is list of most faced issues:

- Database Connectivity
- High DTU Percentage
- Query Timeouts
- Deadlocks
- Database Storage consumption
- Slow Queries

Tools to Monitor & Troubleshoot Issues

Query Performance Insight

Automatic Tuning

Database Watcher

Dynamic Management Views (DMVs)

Azure Database Portal Dashboard

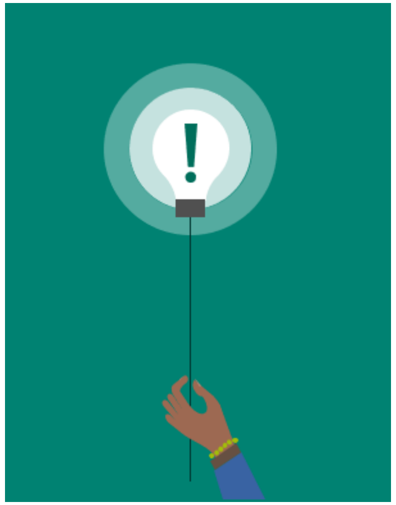
Questions?

Lesson 2: Configure Alerts through Azure Portal

Objectives

After completing this learning, you will be able to:

- Configure alerts using Azure Management Portal.



Purpose of Alerts for Azure SQL Database

Database alerts can help to proactively trigger various events related to database connectivity, high DTU usage or deadlocks, etc.

It helps to proactively resolve underlying issues to avoid application outages and improve user experience.

Receiving an alert based on monitoring metrics or events on

Metric values

- The alert triggers when the value of a specified metric crosses a threshold you assigned in either direction. It triggers when the condition is first met and then when that condition is no longer being met.

Activity log events

- An alert can trigger on every event, or, only when a certain number of events occur.

Purpose of Alerts for Azure SQL Database

You can configure an alert to do the following when it triggers:

- Send email notifications to the service administrator and co-administrators.
- Send email to additional emails that you specify.
- Call a webhook

You can configure and get information about alert rules using

- Azure portal
- PowerShell
- command-line interface (CLI).
- Azure Monitor REST API.

SQL Database alert values

Metric Name	Aggregation Type	Minimum Alert Time Window
CPU percentage	Average	5 minutes
Data IO percentage	Average	5 minutes
Log IO percentage	Average	5 minutes
DTU percentage	Average	5 minutes
Total database size	Maximum	30 minutes
Successful Connections	Total	10 minutes
Failed Connections	Total	10 minutes
Blocked by Firewall	Total	10 minutes
Deadlocks	Total	10 minutes
Database size percentage	Maximum	30 minutes
In-Memory OLTP storage percent(Preview)	Average	5 minutes
Workers percentage	Average	5 minutes
Sessions percent	Average	5 minutes
DTU limit	Average	5 minutes
DTU used	Average	5 minutes

Demonstration

Configure Alerts through Azure Portal

- Configure alerts through Azure Portal.

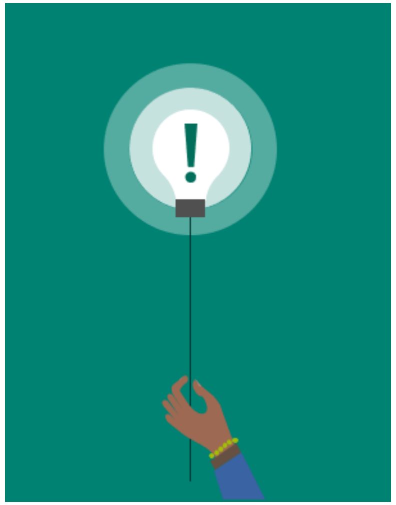
Questions?

Lesson 3: Monitoring Query Performance using Query Performance Insight

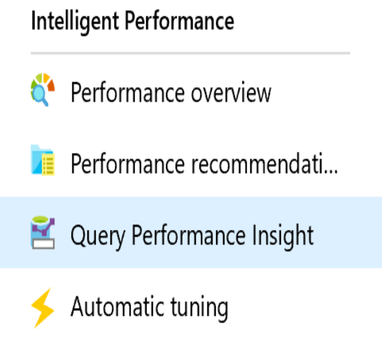
Objectives

After completing this learning, you will be able to:

- Know how to troubleshoot the performance of your queries by using Query Performance Insight.

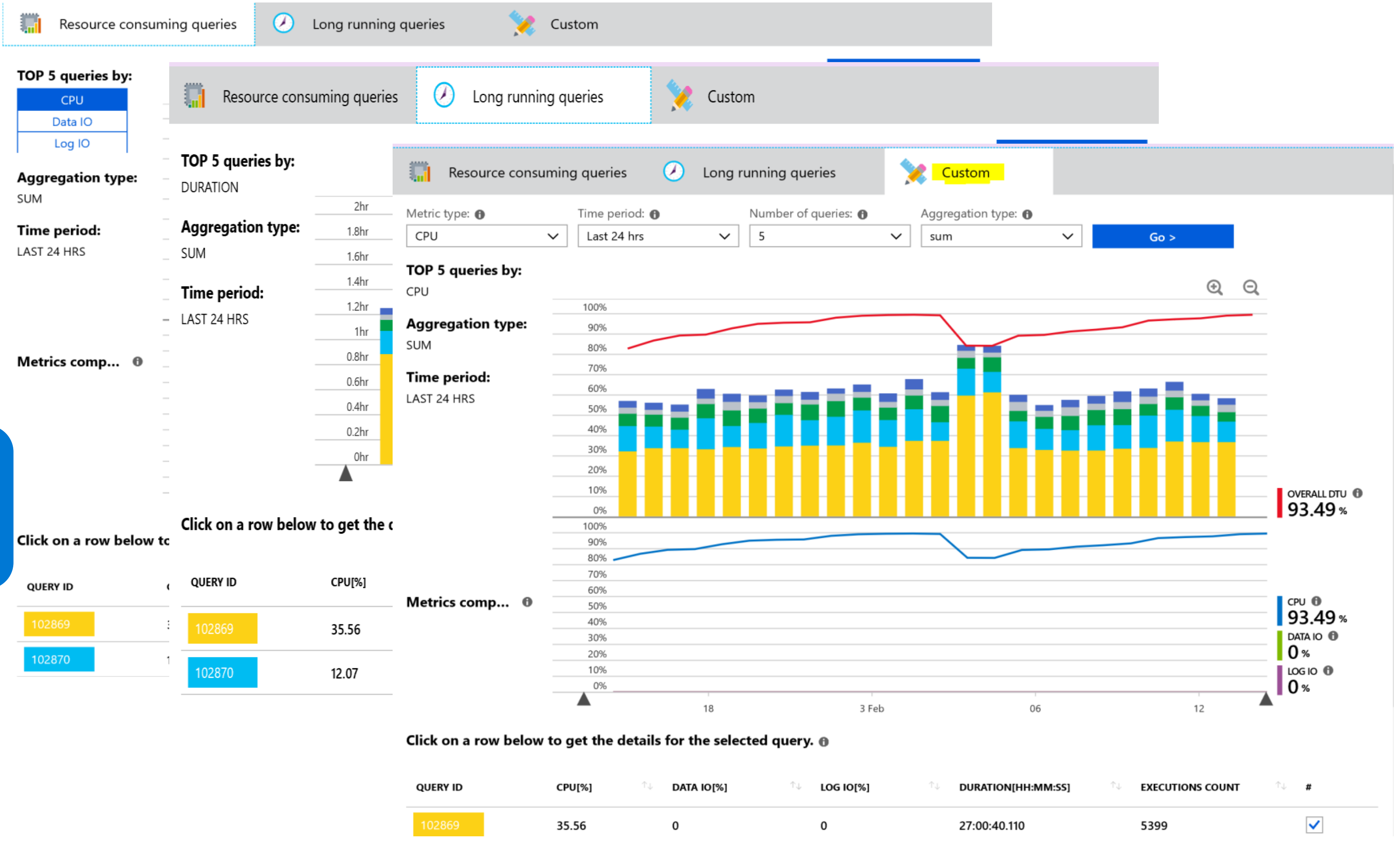


Query Performance Insight



Intelligent Performance

- Performance overview
- Performance recommendati...
- Query Performance Insight
- Automatic tuning



Click on a row below to get the details for the selected query.

QUERY ID	CPU[%]	DATA IO[%]	LOG IO[%]	DURATION[HH:MM:SS]	EXECUTIONS COUNT	#
102869	35.56	0	0	27:00:40.110	5399	☑

Custom options – Insights based upon custom selection:

- **Metric Type** – CPU , Data IO, Log IO, Queries and Execution Count.
- **Time Period** – Last 6 hrs, Last 24 hrs, Past Week, Past Month and Custom.
- **Number of Queries** – 5, 10, 20.
- **Aggregation type** – sum, max and avg.

most resource consuming queries, long running queries and custom configuration.

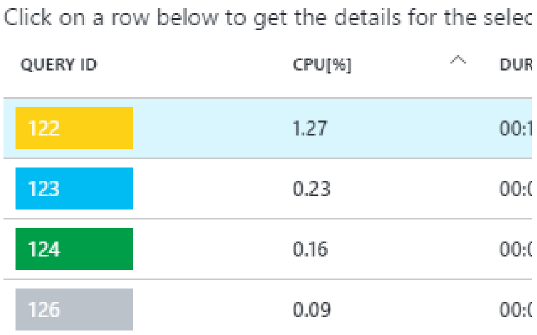
- Can drill through the queries to see Query text, CPU, data IO and Log IO utilization %, Duration and Execution count.

Viewing individual query details

Get details for the individual queries

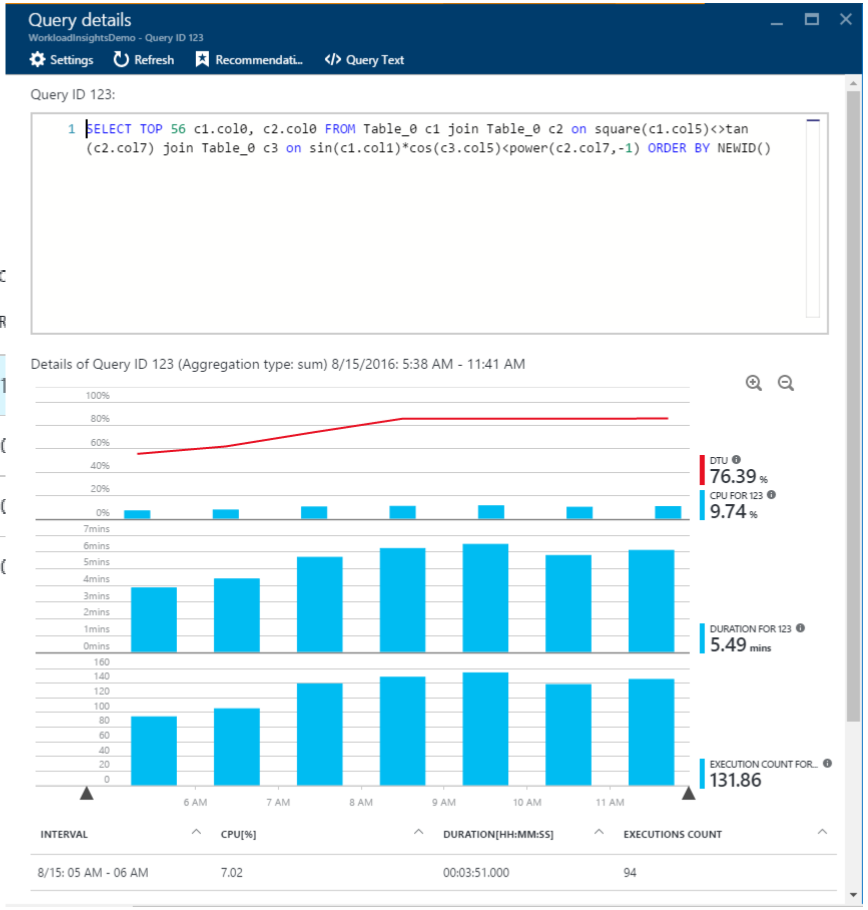
- CPU Consumption
- Duration
- Execution Count

It does not capture DDL queries



Click on a row below to get the details for the selec

QUERY ID	CPU[%]	DUR
122	1.27	00:1
123	0.23	00:(
124	0.16	00:(
126	0.09	00:(



Query ID 123:

```sql
SELECT TOP 56 c1.col0, c2.col0 FROM Table_0 c1 join Table_0 c2 on square(c1.col5)<>tan
(c2.col7) join Table_0 c3 on sin(c1.col1)*cos(c3.col5)<power(c2.col7,-1) ORDER BY NEWID()
```

Details of Query ID 123 (Aggregation type: sum) 8/15/2016: 5:38 AM - 11:41 AM

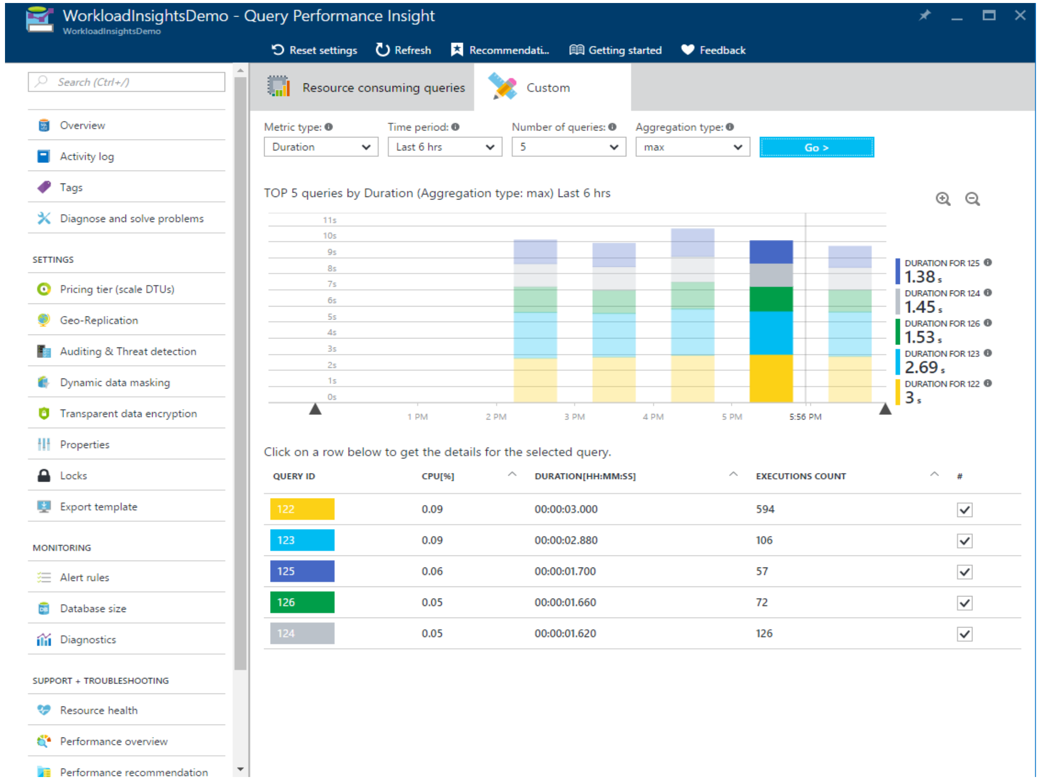
INTERVAL	CPU[%]	DURATION[HH:MM:SS]	EXECUTIONS COUNT
8/15: 05 AM - 06 AM	7.02	00:03:51.000	94

Review top queries per duration

Duration is one of the metrics showing potential bottleneck

Long-running queries has potential for:

- Longer locks
- Blocking other users
- Limiting scalability



Metric type: Duration
Time period: Last 6 hrs
Number of queries: 5
Aggregation type: max

TOP 5 queries by Duration (Aggregation type: max) Last 6 hrs

Click on a row below to get the details for the selected query.

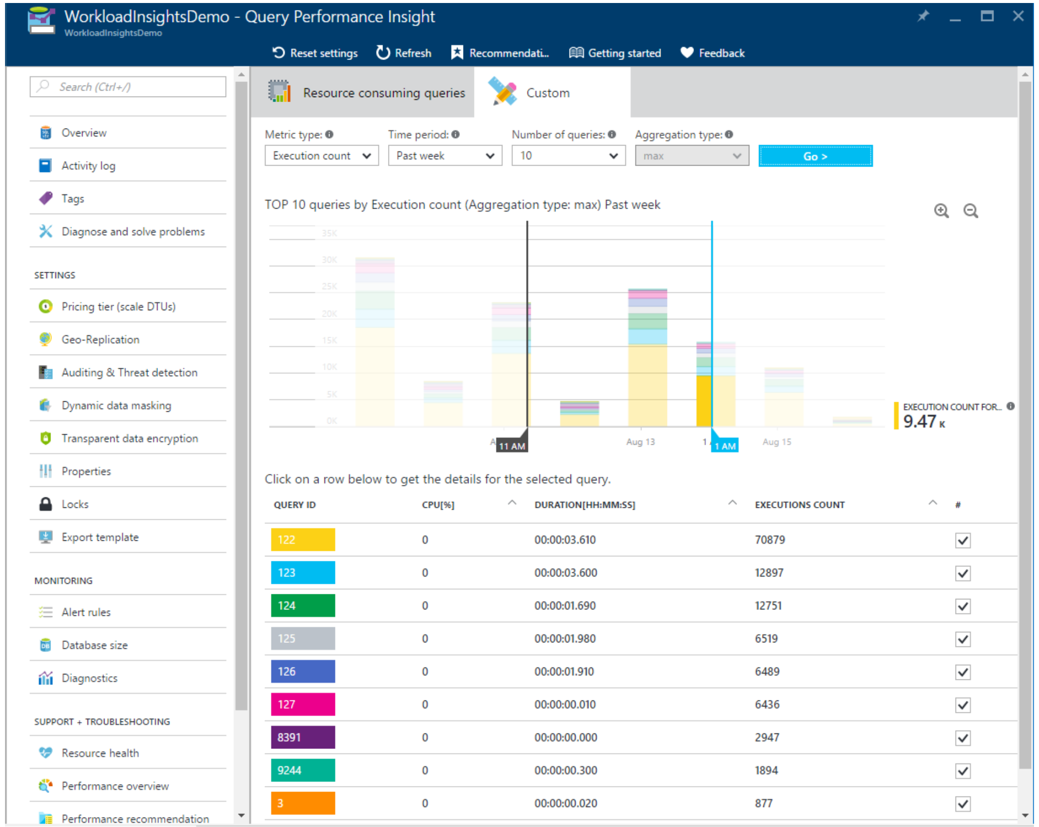
QUERY ID	CPU[%]	DURATION[HH:MM:SS]	EXECUTIONS COUNT	#
122	0.09	00:00:03.000	594	☑
123	0.09	00:00:02.880	106	☑
125	0.06	00:00:01.700	57	☑
126	0.05	00:00:01.660	72	☑
124	0.05	00:00:01.620	126	☑

Review top queries per execution count

Execution count is one of the metrics showing potential bottleneck

High number og executions has potential for:

- Database performance
- Network latency
- Downstream server latency



Metric type: Execution count

Time period: Past week

Number of queries: 10

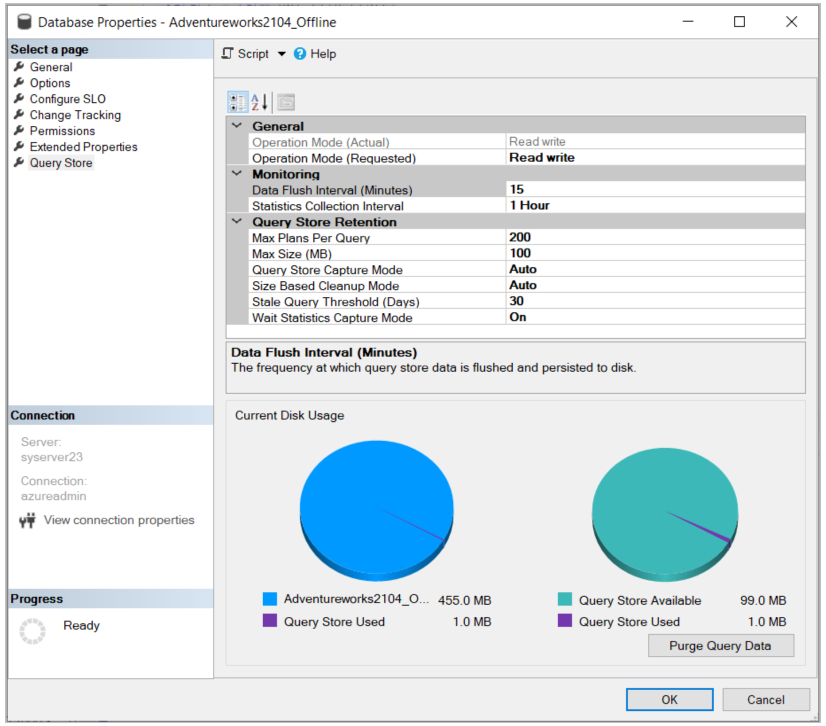
Aggregation type: max

TOP 10 queries by Execution count (Aggregation type: max) Past week

Click on a row below to get the details for the selected query.

QUERY ID	CPU[%]	DURATION[HH:MM:SS]	EXECUTIONS COUNT	#
122	0	00:00:03.610	70879	☑
123	0	00:00:03.600	12897	☑
124	0	00:00:01.690	12751	☑
125	0	00:00:01.980	6519	☑
126	0	00:00:01.910	6489	☑
127	0	00:00:00.010	6436	☑
8391	0	00:00:00.000	2947	☑
9244	0	00:00:00.300	1894	☑
3	0	00:00:00.020	877	☑

Query Store



Database Properties - Adventureworks2104_Offline

Server: syserver23

Connection: azureadmin

Ready

General	
Operation Mode (Actual)	Read write
Operation Mode (Requested)	**Read write**
Monitoring	
Data Flush Interval (Minutes)	**15**
Statistics Collection Interval	**1 Hour**
Query Store Retention	
Max Plans Per Query	**200**
Max Size (MB)	**100**
Query Store Capture Mode	**Auto**
Size Based Cleanup Mode	**Auto**
Stale Query Threshold (Days)	**30**
Wait Statistics Capture Mode	**On**

Data Flush Interval (Minutes)
The frequency at which query store data is flushed and persisted to disk.

Adventureworks2104_O...	455.0 MB
Query Store Used	1.0 MB

Query Store Available	99.0 MB
Query Store Used	1.0 MB

Retention Policy

- Size based – Auto cleanup when near max size.
- Time based – Default 30 days.
- Max Plans Per Query – Default 200.
- Wait Statistics Capture Mode – Default On.

Capture Policy

- All – Captures all queries.
- Auto – Infrequent queries are ignored.
- None – No queries are captured.
- Custom – Advanced Options

Demonstration

Query Performance Insight

- Analyze the Query Performance Insight output.

Monitoring Query Performance using Query Performance Insights

- Configure the Query Store.
- Analyze the Query Performance Insight.

LAB

Questions?

Knowledge Check

What feature should be enabled on your Azure SQL Database before you can use Query Performance Insight?

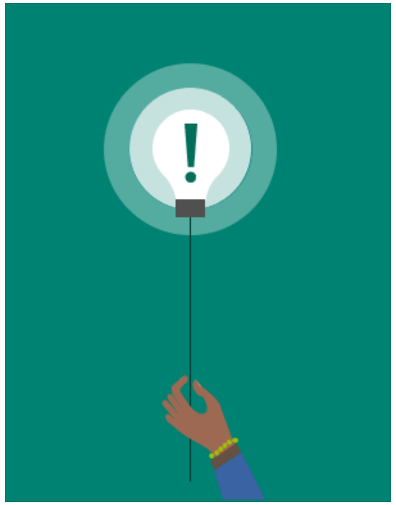
How can you view individual query details?

Lesson 4: Azure SQL Database Tuning using Automatic Tuning

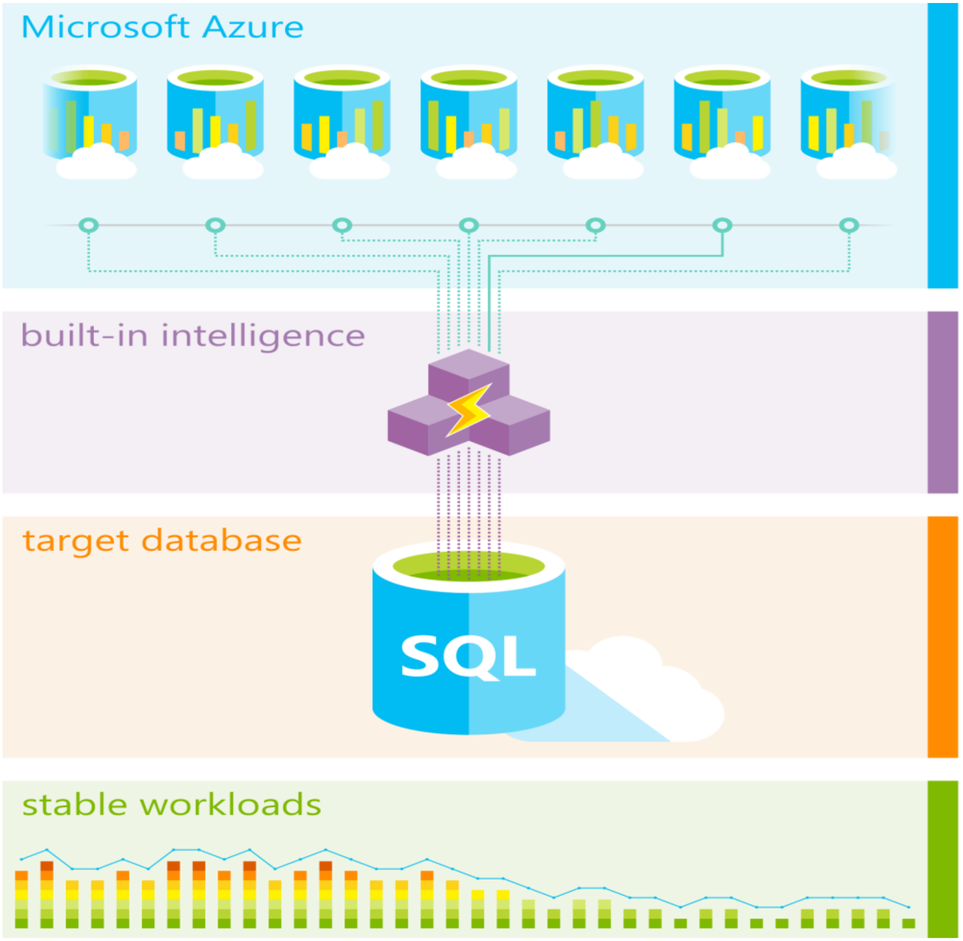
Objectives

After completing this learning, you will be able to:

- Know how Performance Recommendations can help to improve database performance.



Automatic Tuning



horizontally learning
from all databases

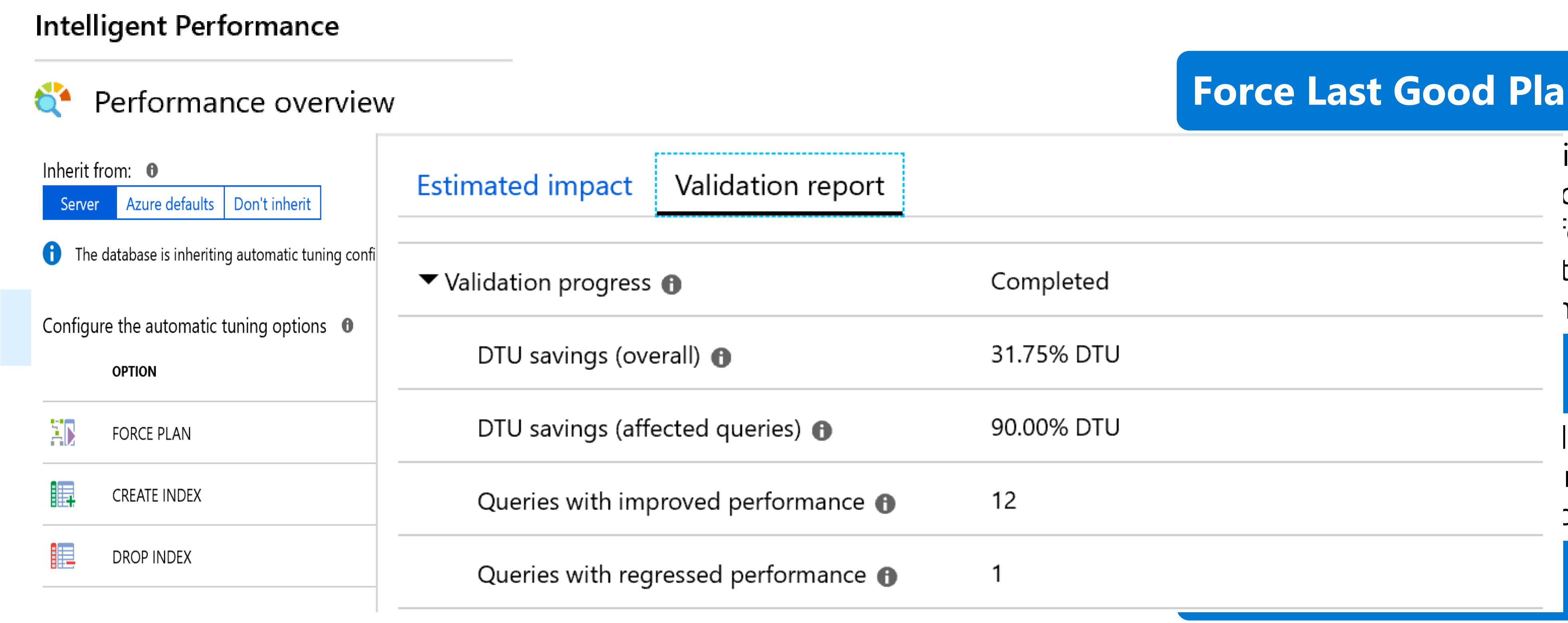
finding optimization
opportunities

applying during
low-usage periods

verifying
performance gains
& reverting if required

[Performance recommendations for SQL Database]

Intelligent Performance – Automatic Tuning



Intelligent Performance

Performance overview

Inherit from: Server | Azure defaults | Don't inherit

The database is inheriting automatic tuning confi

Configure the automatic tuning options

OPTION
FORCE PLAN
CREATE INDEX
DROP INDEX

Validation progress	Completed
DTU savings (overall)	31.75% DTU
DTU savings (affected queries)	90.00% DTU
Queries with improved performance	12
Queries with regressed performance	1

Force Last Good Plan:

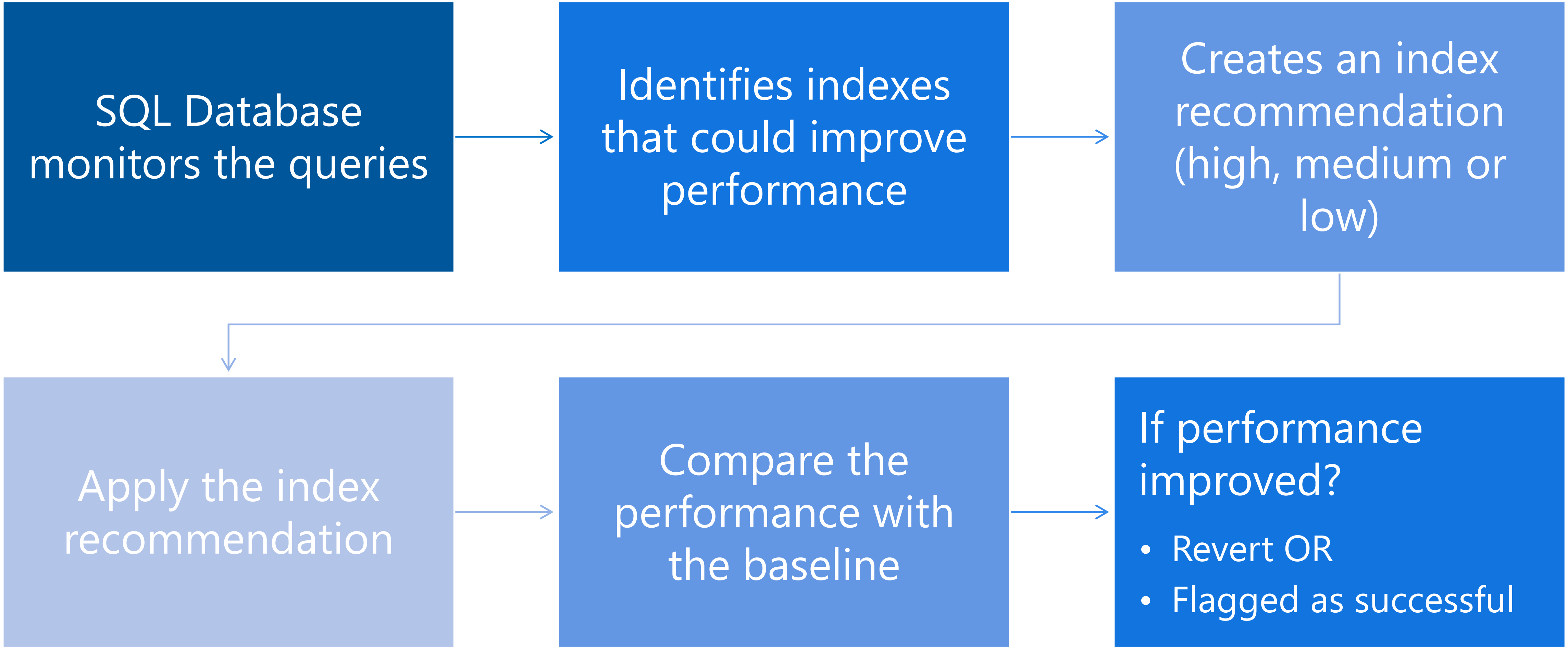
ies due to bad
d plan with last
ormance
ts the change if
prove.

lexes, validates
nts and reverts
e degrades.

- Identifies and drops unused Indexes, validates performance improvements and reverts the change if performance degrades.

http://automaticplancorrectiondemo.azurewebsites.net/index.html

Automatic Tuning – Create Index



SQL Database monitors the queries

Identifies indexes that could improve performance

Creates an index recommendation (high, medium or low)

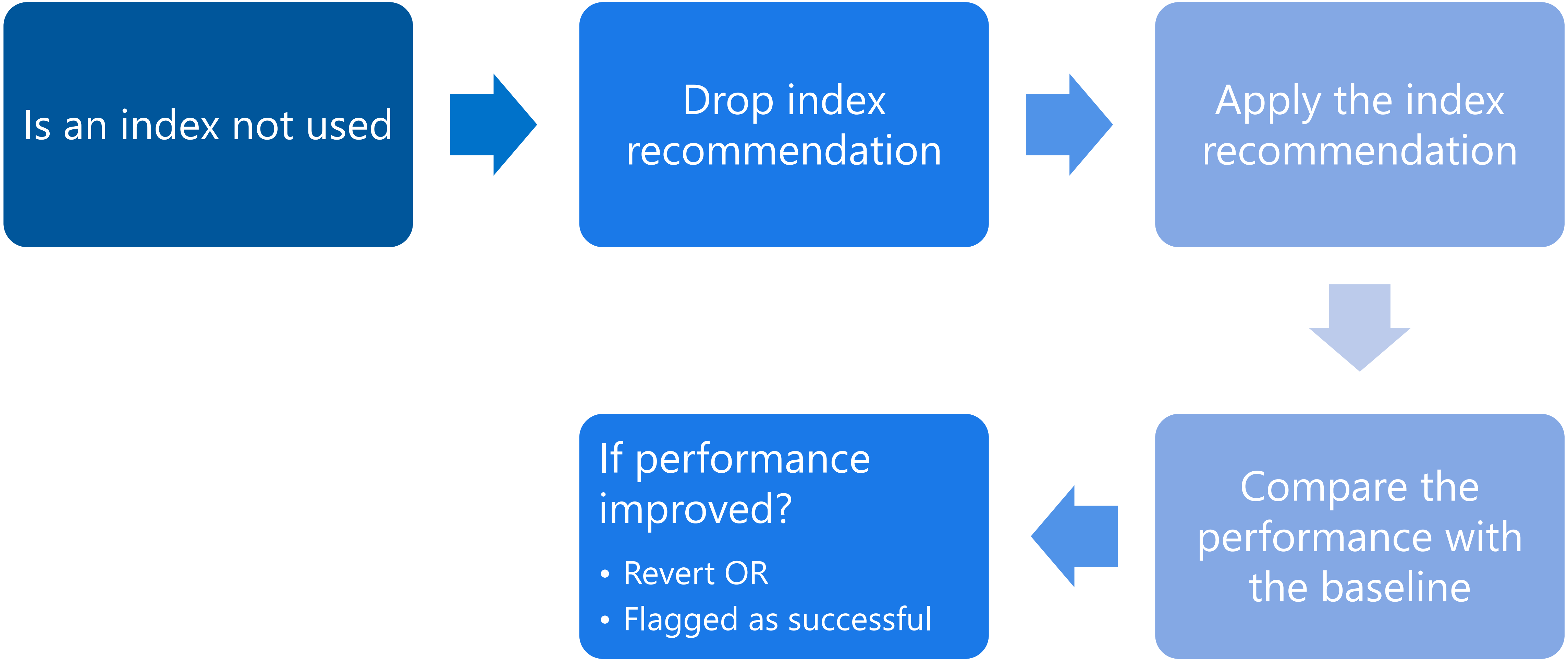
Apply the index recommendation

Compare the performance with the baseline

If performance improved?

- Revert OR
- Flagged as successful

Automatic Tuning – Drop Index



Is an index not used

Drop index recommendation

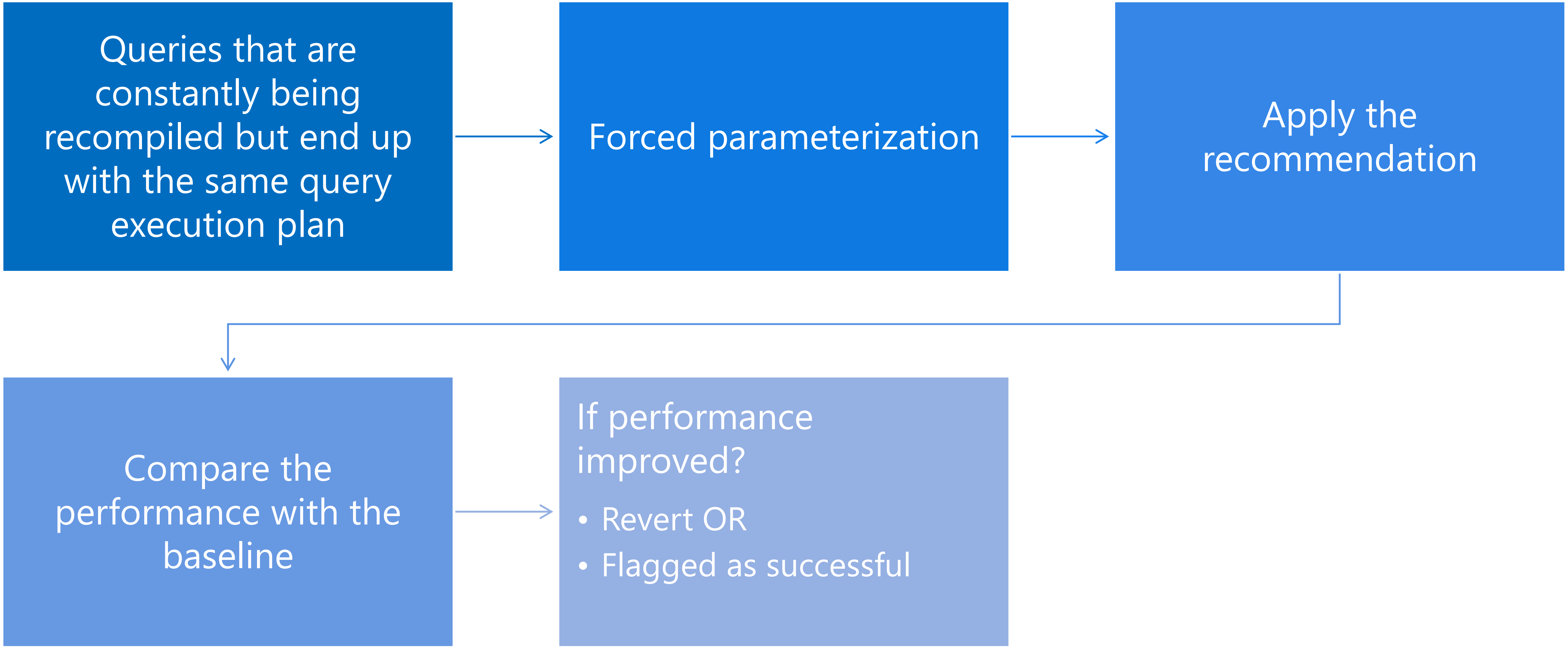
Apply the index recommendation

Compare the performance with the baseline

If performance improved?

- Revert OR
- Flagged as successful

Automatic Tuning – Parameterize Queries



Queries that are constantly being recompiled but end up with the same query execution plan

Forced parameterization

Apply the recommendation

Compare the performance with the baseline

If performance improved?

- Revert OR
- Flagged as successful

Questions?

Knowledge Check

List three types of recommendations from Automatic Tuning.

What could be a reason to disable the automatic tuning option?

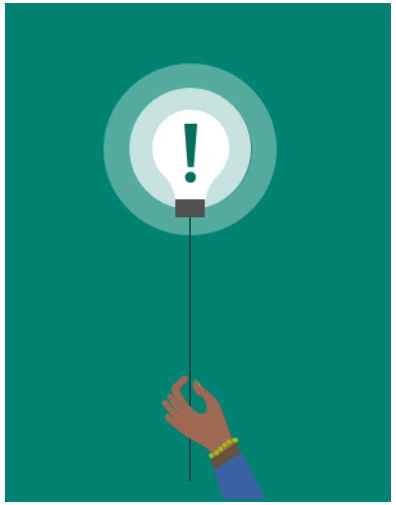
What technology is used for Automatic Tuning?

Lesson 5: Monitoring Performance using Database Watcher

Objectives

After completing this learning, you will be able to:

- Use Database Watcher to monitor database performance.
- Use Database Watcher to monitor any product in the Azure SQL family.



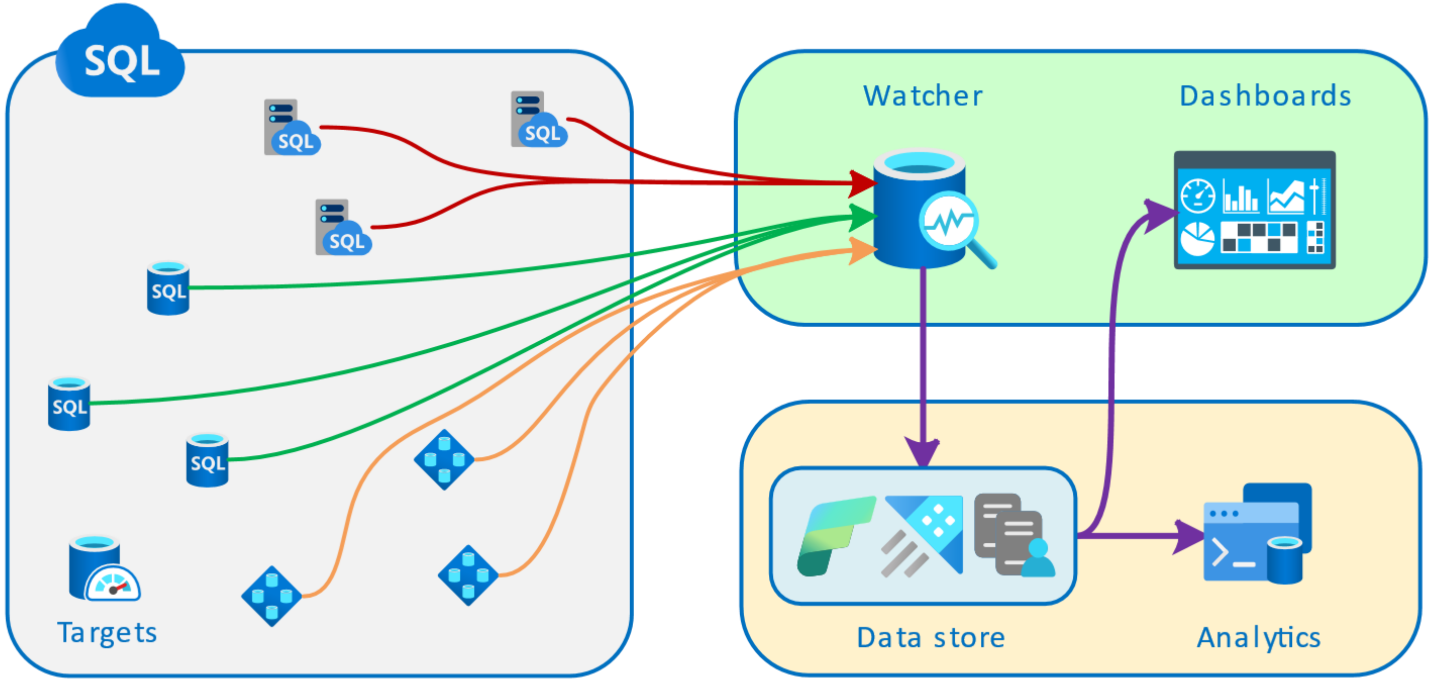
Introduction to Database Watcher (Preview)

Database watcher is a managed monitoring solution for database services in the Azure SQL family.

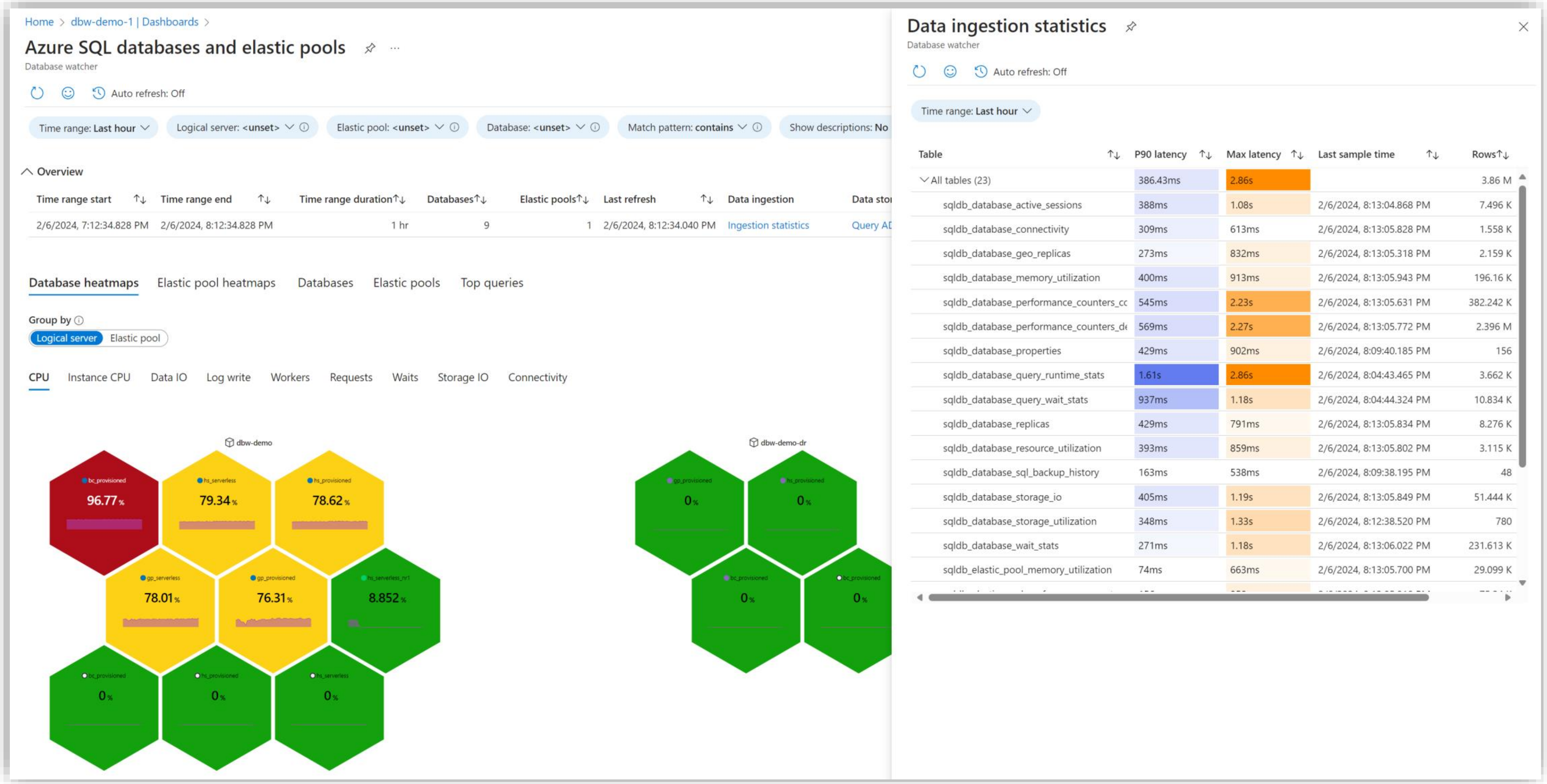
Supports Azure SQL Database, Elastic Pools, and Managed Instances

Collects workload monitoring data to give you a detailed view of database performance, configuration, and health.

Currently in Preview and only available in limited Azure regions.



Database Watcher Dashboards



Home > dbw-demo-1 | Dashboards >

Azure SQL databases and elastic pools

Database watcher

Auto refresh: Off

Time range: Last hour
Logical server: <unset>
Elastic pool: <unset>
Database: <unset>
Match pattern: contains
Show descriptions: No

Overview

Time range start	Time range end	Time range duration	Databases	Elastic pools	Last refresh	Data ingestion	Data stor
2/6/2024, 7:12:34.828 PM	2/6/2024, 8:12:34.828 PM	1 hr	9	1	2/6/2024, 8:12:34.040 PM	Ingestion statistics	Query AD

Database heatmaps
Elastic pool heatmaps
Databases
Elastic pools
Top queries

Group by

Logical server
Elastic pool

CPU
Instance CPU
Data IO
Log write
Workers
Requests
Waits
Storage IO
Connectivity

Data ingestion statistics

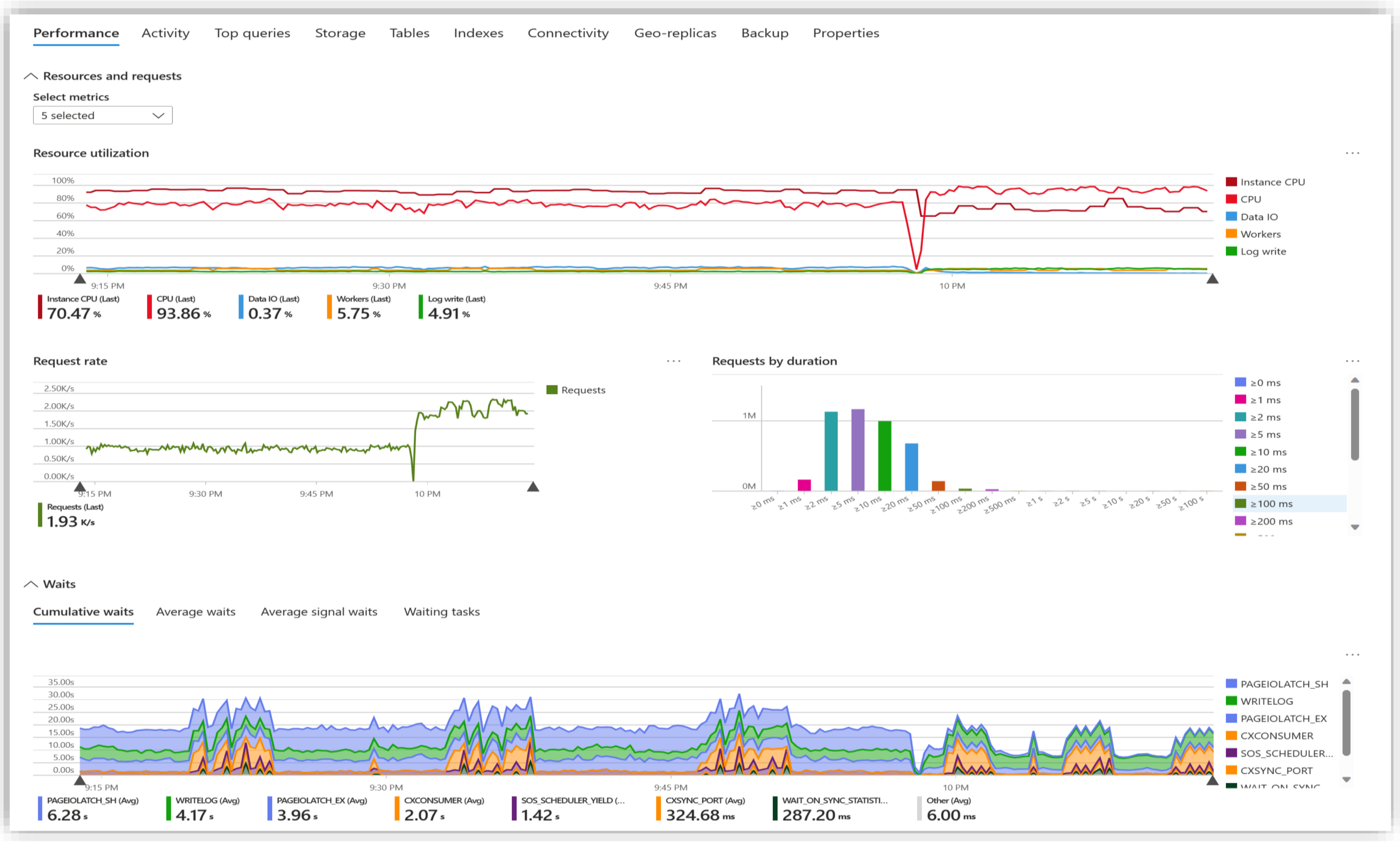
Database watcher

Auto refresh: Off

Time range: Last hour

Table	P90 latency	Max latency	Last sample time	Rows
All tables (23)	386.43ms	2.86s		3.86 M
sqldb_database_active_sessions	388ms	1.08s	2/6/2024, 8:13:04.868 PM	7.496 K
sqldb_database_connectivity	309ms	613ms	2/6/2024, 8:13:05.828 PM	1.558 K
sqldb_database_geo_replicas	273ms	832ms	2/6/2024, 8:13:05.318 PM	2.159 K
sqldb_database_memory_utilization	400ms	913ms	2/6/2024, 8:13:05.943 PM	196.16 K
sqldb_database_performance_counters_cc	545ms	2.23s	2/6/2024, 8:13:05.631 PM	382.242 K
sqldb_database_performance_counters_de	569ms	2.27s	2/6/2024, 8:13:05.772 PM	2.396 M
sqldb_database_properties	429ms	902ms	2/6/2024, 8:09:40.185 PM	156
sqldb_database_query_runtime_stats	1.61s	2.86s	2/6/2024, 8:04:43.465 PM	3.662 K
sqldb_database_query_wait_stats	937ms	1.18s	2/6/2024, 8:04:44.324 PM	10.834 K
sqldb_database_replicas	429ms	791ms	2/6/2024, 8:13:05.834 PM	8.276 K
sqldb_database_resource_utilization	393ms	859ms	2/6/2024, 8:13:05.802 PM	3.115 K
sqldb_database_sql_backup_history	163ms	538ms	2/6/2024, 8:09:38.195 PM	48
sqldb_database_storage_io	405ms	1.19s	2/6/2024, 8:13:05.849 PM	51.444 K
sqldb_database_storage_utilization	348ms	1.33s	2/6/2024, 8:12:38.520 PM	780
sqldb_database_wait_stats	271ms	1.18s	2/6/2024, 8:13:06.022 PM	231.613 K
sqldb_elastic_pool_memory_utilization	74ms	663ms	2/6/2024, 8:13:05.700 PM	29.099 K

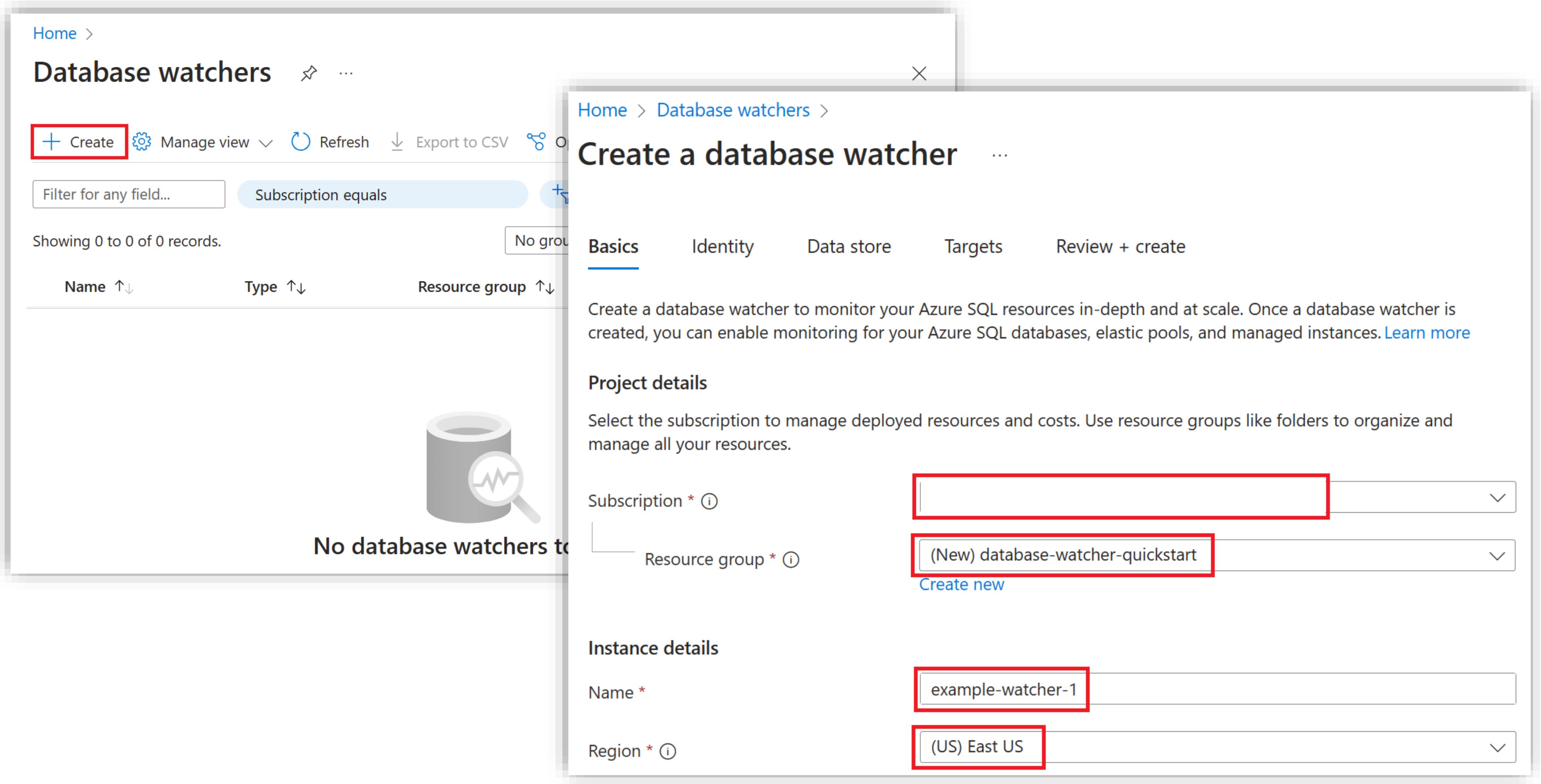
Database Watcher Dashboards



Select metrics

5 selected

Create a Database Watcher Service



Database watchers

Showing 0 to 0 of 0 records.

No database watchers t

Create a database watcher

Create a database watcher to monitor your Azure SQL resources in-depth and at scale. Once a database watcher is created, you can enable monitoring for your Azure SQL databases, elastic pools, and managed instances. Learn more

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * ⓘ

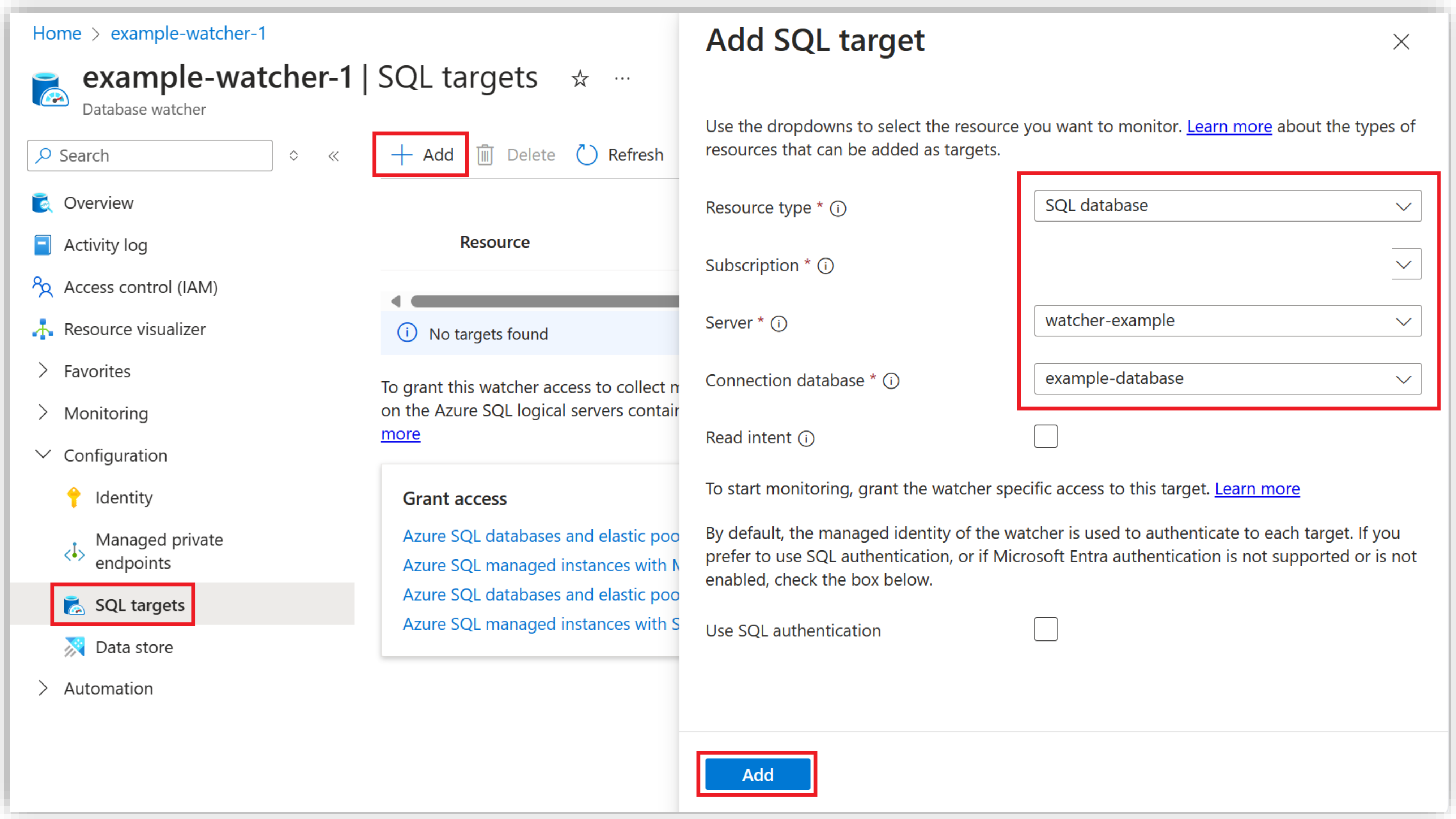
Resource group * ⓘ (New) database-watcher-quickstart

Instance details

Name * example-watcher-1

Region * ⓘ (US) East US

Add Targets to Database Watcher



No targets found

To grant this watcher access to collect m
on the Azure SQL logical servers contair
more

Grant access

Azure SQL databases and elastic poo
Azure SQL managed instances with M
Azure SQL databases and elastic poo
Azure SQL managed instances with S

Add SQL target

Use the dropdowns to select the resource you want to monitor. Learn more about the types of resources that can be added as targets.

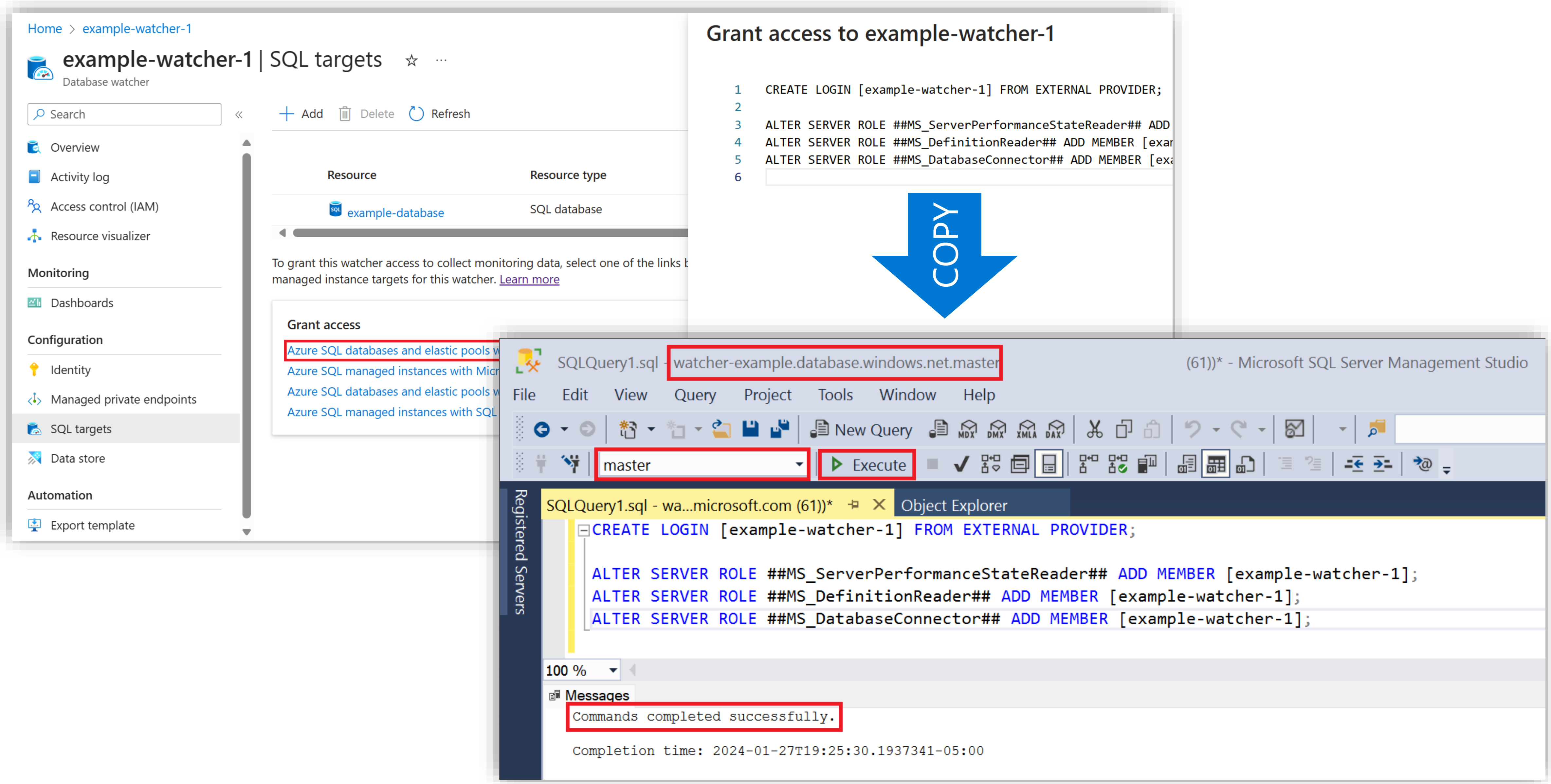
Resource type * ⓘ	SQL database
Subscription * ⓘ	
Server * ⓘ	watcher-example
Connection database * ⓘ	example-database
Read intent ⓘ	□

To start monitoring, grant the watcher specific access to this target. Learn more

By default, the managed identity of the watcher is used to authenticate to each target. If you prefer to use SQL authentication, or if Microsoft Entra authentication is not supported or is not enabled, check the box below.

Use SQL authentication □

Grant Access for Database Watcher



Resource	Resource type
example-database	SQL database

To grant this watcher access to collect monitoring data, select one of the links b managed instance targets for this watcher. Learn more

Grant access

- Azure SQL databases and elastic pools w
- Azure SQL managed instances with Micr
- Azure SQL databases and elastic pools w
- Azure SQL managed instances with SQL

Grant access to example-watcher-1

```
CREATE LOGIN [example-watcher-1] FROM EXTERNAL PROVIDER;

ALTER SERVER ROLE ##MS_ServerPerformanceStateReader## ADD
ALTER SERVER ROLE ##MS_DefinitionReader## ADD MEMBER [exar
ALTER SERVER ROLE ##MS_DatabaseConnector## ADD MEMBER [exa
```

SQLQuery1.sql - watcher-example.database.windows.net.master (61))* - Microsoft SQL Server Management Studio
File
Edit
View
Query
Project
Tools
Window
Help
New Query
master
Execute
SQLQuery1.sql - wa...microsoft.com (61))*
Object Explorer
Registered Servers
100 %
Messages

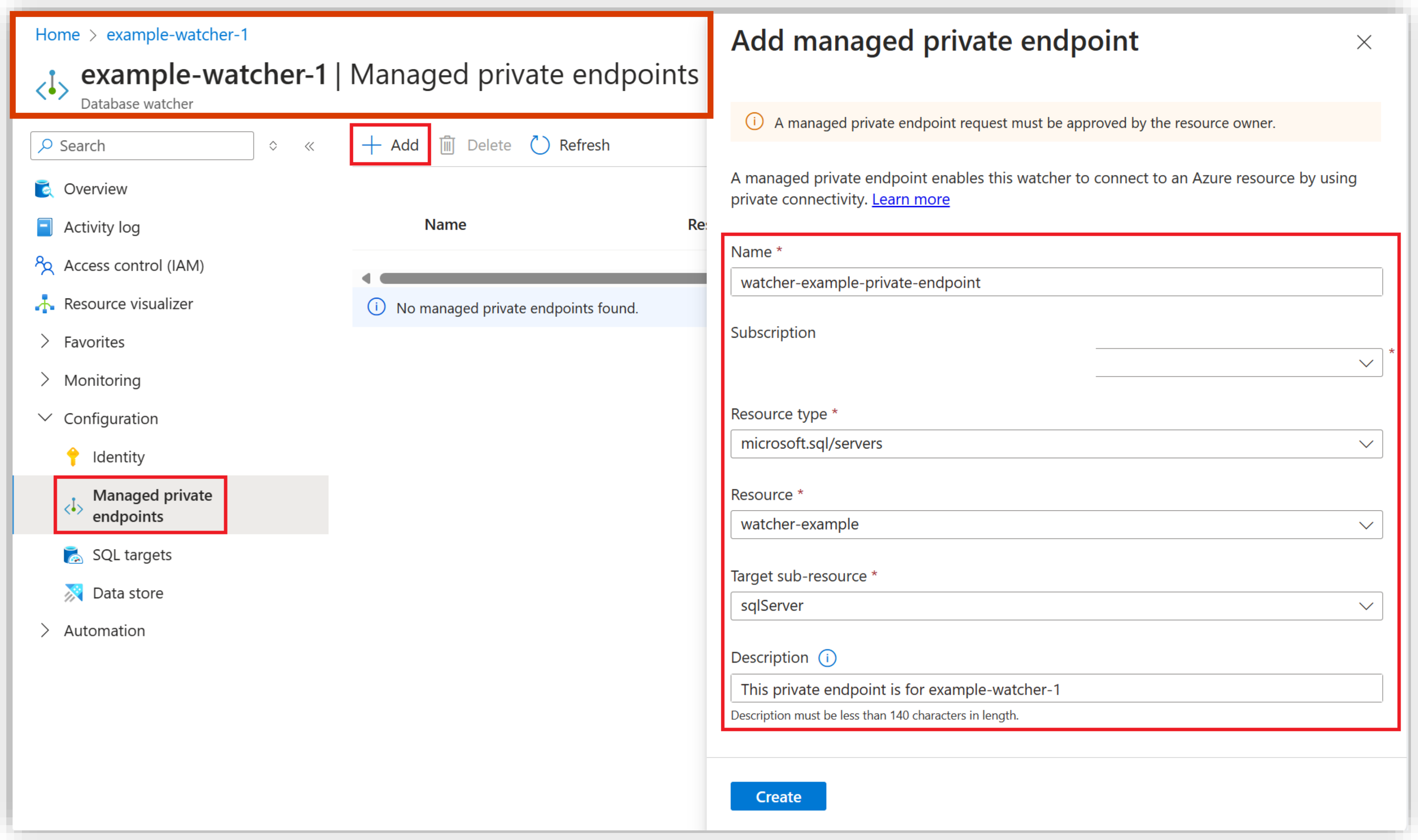
```
CREATE LOGIN [example-watcher-1] FROM EXTERNAL PROVIDER;

ALTER SERVER ROLE ##MS_ServerPerformanceStateReader## ADD MEMBER [example-watcher-1];
ALTER SERVER ROLE ##MS_DefinitionReader## ADD MEMBER [example-watcher-1];
ALTER SERVER ROLE ##MS_DatabaseConnector## ADD MEMBER [example-watcher-1];
```

```
Commands completed successfully.

Completion time: 2024-01-27T19:25:30.1937341-05:00
```

Create Managed Private Endpoint



Home > example-watcher-1

example-watcher-1 | Managed private endpoints

Database watcher

No managed private endpoints found.

Add managed private endpoint

A managed private endpoint request must be approved by the resource owner.

A managed private endpoint enables this watcher to connect to an Azure resource by using private connectivity. Learn more

Name *: watcher-example-private-endpoint

Subscription *:

Resource type *: microsoft.sql/servers

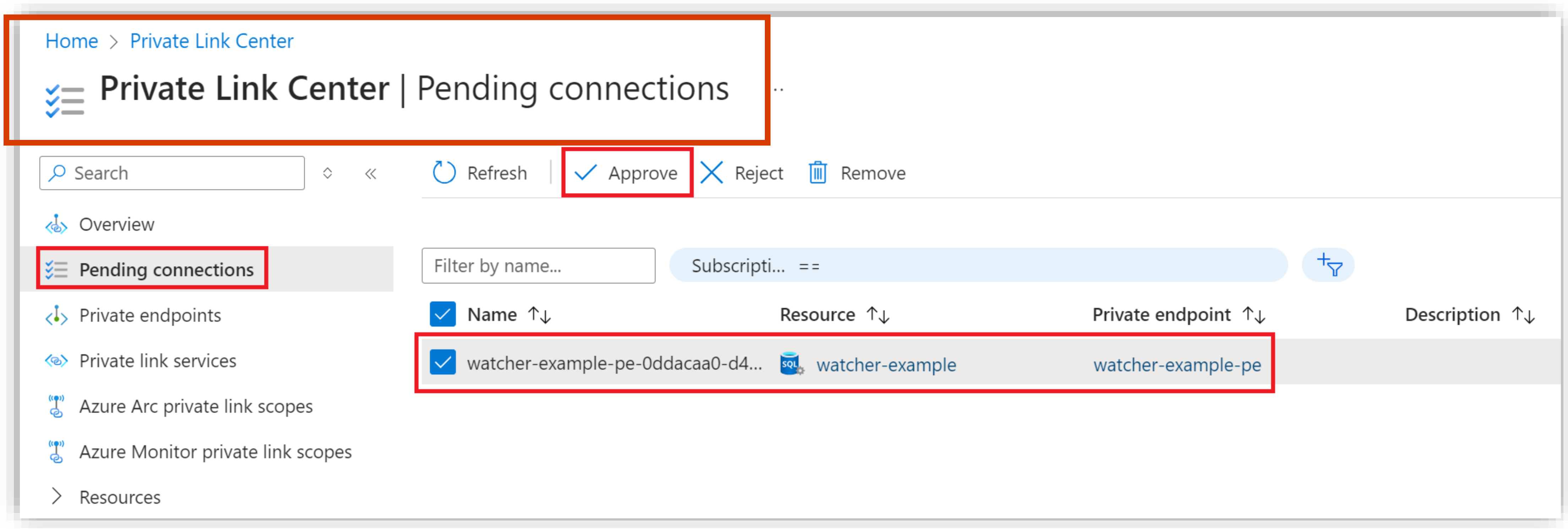
Resource *: watcher-example

Target sub-resource *: sqlServer

Description: This private endpoint is for example-watcher-1

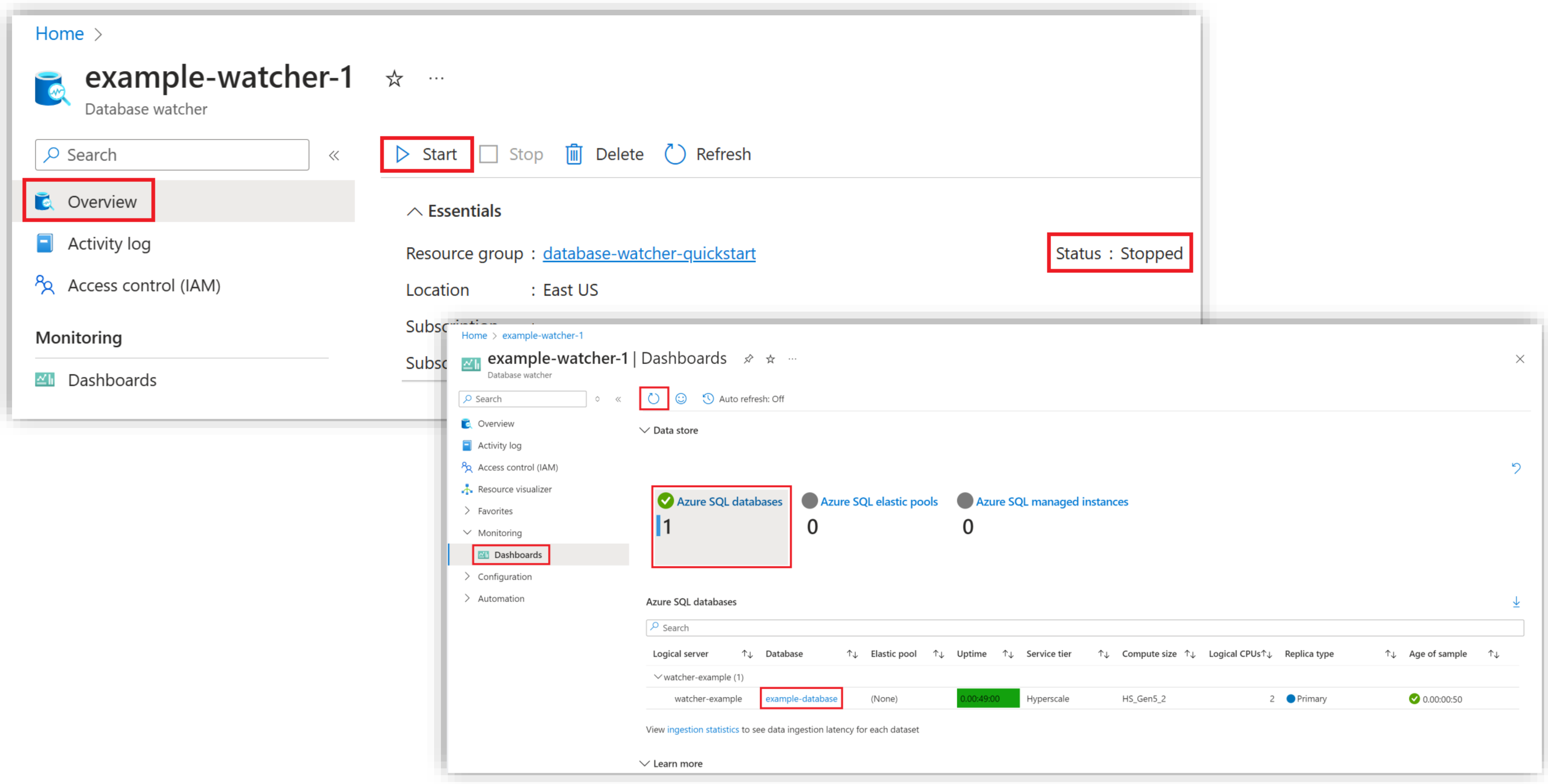
Description must be less than 140 characters in length.

Approve Managed Private Endpoint



☑	Name ↑↓	Resource ↑↓	Private endpoint ↑↓	Description ↑↓
☑	watcher-example-pe-0ddacaa0-d4...	watcher-example	watcher-example-pe	

Start Database Watcher



Resource group : database-watcher-quickstart

Status : Stopped

Location : East US

Subscription :

Subsc

Azure SQL databases	Azure SQL elastic pools	Azure SQL managed instances
1	0	0

Azure SQL databases

Logical server	Database	Elastic pool	Uptime	Service tier	Compute size	Logical CPUs	Replica type	Age of sample
watcher-example (1)								
watcher-example	example-database	(None)	0.00:49:00	Hyperscale	HS_Gen5_2	2	● Primary	✓ 0.00:00:50

View ingestion statistics to see data ingestion latency for each dataset

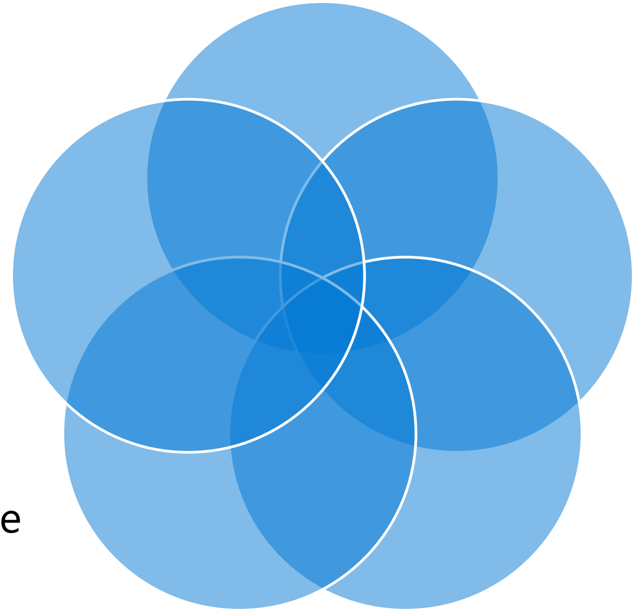
Demonstration

Monitor Performance with Database Watcher

- Azure SQL Database monitoring with Database Watcher

Questions?

Module Summary



Monitoring and Troubleshooting Azure SQL Database

Monitoring Query Performance using Query Performance Insight

Azure SQL Database Tuning using Automatic Tuning

Monitoring Azure SQL Database Performance using Database Watcher

Configure Alerts through Azure Portal

Microsoft