Querying many SQL Azure database to get information - Elastic Database Jobs

Last updated by | Vitor Tomaz | Aug 5, 2020 at 12:42 PM PDT

Problem

Querying many SQL Azure databases and collecting the result in one central location is difficult without the overhead of setting up large number of external objects and elastic query. This is where elastic database jobs is quite good, as you can pull all the data in one central location.

Solution:

I am assuming that all the targeting databases have appropriate logins and users have the permissions required to perform the activities in the job. You can refer the following link for more information on setting up the logins and credentials.

https://docs.microsoft.com/en-us/azure/sql-database/elastic-jobs-powershell I am using databases in elastic pool, but you can add/exclude individual databases according to your needs. Here's the script: This is an example of getting counts of a particular table from all the databases. Please create a blank database called ResultDB. The result will be store in a table called ResultsTable which will be automatically created. 1. Create a Target group which will target your elastic pool:

-- Add a target group containing pool(s)

EXEC jobs.sp_add_target_group 'PoolGroup'

-- Add an elastic pool(s) target member

EXEC jobs.sp_add_target_group_member 'PoolGroup', @target_type = 'SqlElasticPool', @refresh_credential_name='mymastercred', @server_name='sarrveshtest.database.windows.net', @elastic_pool_name='SarveshElasticPool'

2. Create an Elastic Job.

EXEC jobs.sp_add_job @job_name='GetCounts', @description='GetCountofCustomers'

3. Create job step to query all databases. You can check the query in @Command variable according to your requirement. In this case I am just getting the counts for customer table from all databases in my pool

DECLARE @JobName NVARCHAR(128) = N'GetCounts'

DECLARE @JobStepName NVARCHAR(128) = N'CustomerCount'

DECLARE @Command NVARCHAR(MAX)=N'select t.name ,s.row_count from sys.tables t join sys.dm_db_partition_stats s ON t.object_id = s.object_id and t.type_desc = ''USER_TABLE'' and t.name ='Customer'' and s.index_id = 1 '

```
declare @CredentialName NVARCHAR(128)='myjobcred'

declare @TargetGroupName NVARCHAR(128)='PoolGroup'

declare @output_type NVARCHAR(128) ='SqlDatabase'

declare @output_credential_name NVARCHAR(128) ='myjobcred'

declare @output_server_name NVARCHAR(128) ='sarrveshtest.database.windows.net '

declare @output_database_name NVARCHAR(128) ='ResultDB'

declare @output_table_name NVARCHAR(128) ='resultstable'

-- Add job step

EXEC jobs.sp_add_jobstep @job_name = @JobName,

                @step_name = @JobStepName,

                @command = @Command,

@credential_name = @CredentialName,

@target_group_name = @TargetGroupName,

@output_type = @output_type,

@output_credential_name = @output_credential_name,

@output_server_name = @output_server_name,

            @output_database_name = @output_database_name,

            @output_table_name = @output_table_name
```

4. Execute Job

```
EXEC jobs.sp_start_job @job_name = 'GetCounts'
```

5. Check the job execution log

```
select * from jobs.job_executions where job_name='GetCounts'
```

6. Check the resultstable in ResultDB

```
select * from resultstable
```

Resources:

https://docs.microsoft.com/en-us/azure/sql-database/elastic-jobs-powershell

https://docs.microsoft.com/en-us/azure/sql-database/elastic-jobs-tsql

Thanks.
Sarvesh

How good have you found this content?