Convert SQL Server Agent Jobs

Into

Elastic Database Jobs

# Introduction

The goal of the script is to migrate/copy existing SQL Server Agent Jobs to Elastic Database Jobs

# Prerequisites

The following resources are necessary before you can execute the migration script

* Install the SqlServer PoweShell module.
* Make sure you have created the [Elastic Job Agent](https://docs.microsoft.com/en-us/azure/sql-database/elastic-jobs-overview) and the Job database
* Create a [credential for job execution](https://docs.microsoft.com/en-us/azure/sql-database/elastic-jobs-tsql#create-a-credential-for-job-execution) in the Job database
* Make sure that the jobs\_resource\_manager has permission to read the database scoped credential. To do so, use the following command. Change <job credential> into the name of your job credential

GRANT CONTROL ON DATABASE SCOPED CREDENTIAL:: <job credential> TO jobs\_resource\_manager

# How does the script work?

When you launch the script, you will be asked to provide the following parameters

* The name of the resource group where the Elastic Agent is running
* The name of the logical server. Do not specify the FQDN but just the name. For example, MySQLServer and not MySQLServer.database.windows.net
* The name of the job credential
* The name of your SQL Server Instance that contain the jobs that you want to copy

Once all the parameters have been provided, you will see the list of the jobs. The script is going to create each job one by one in a disabled state if it doesn’t exist. Once the job is created, the job steps are added with the same step IDs, Command Text, Retry Attempts and Initial Retry Interval Seconds. The target group will be the database that is linked to the job step. If the target group doesn’t exist, it will automatically be created.

The job schedule is not transferred because the scheduling of Elastic Database Jobs is completely different compared to a traditional SQL Server. The schedule and enabling the job should be done manually. The output of each job migration is put on the screen and in a log file called output.log which is stored in the same folder of where your script is located.

# Limitations

* Disabled jobs will not be transferred
* Job Steps should have a subsystem T-SQL
* Job Steps should connect to a user database
* Schedules, Alerts and Notifications are not transferred