



Exploring Resource Governor IO Enhancements



Contents

Resource Governor enhancements for IO 3

Terms of use18

Resource Governor enhancements for IO

Estimated time to complete lab is 30 minutes

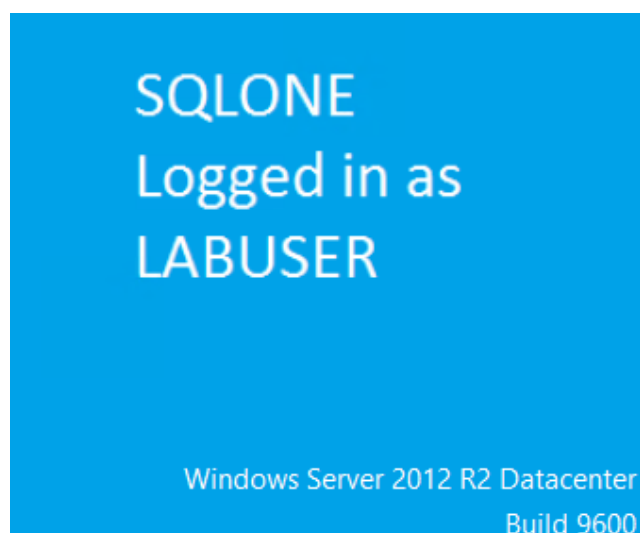
With many users running applications, there are many requests competing with each other for the SQL Server resources. You can put in place resource governance rules meaning queries run by the DBA team have limited I/O resources so their maintenance queries will not slow down others. SQL Server 2014 allows physical limits on the CPU, physical IO and memory that incoming applications can use within a resource pool to stop runaway queries blocking resources for all others

NOTE: The SQL Server 2014 feature of controlling I/O operations is currently only available through T-SQL, so this scenario cannot be completed using the SQL Server 2014 Management Studio wizards.

In this lab, you will first enable the Resource Governor. The Resource Governor is turned off by default. Once you enable it, new connections will be classified so their workloads will be assigned to workload groups, resource limits configured will be honored, and any existing requests will be affected by configuration changes made when the Resource Governor was disabled.

Connect to SQLONE computer

1. Click on **SQLONE** button on right side of the screen to connect to the **SQLONE** computer. If you see the following in the lower right corner of the screen, you can jump to step 5 below to set your screen resolution.



2. Click **Send Ctrl-Alt-Del** for **SQLONE** computer and then click **Switch user**.
3. Click **Send Ctrl-Alt-Del** for **SQLONE** computer again and then click **Other user**.

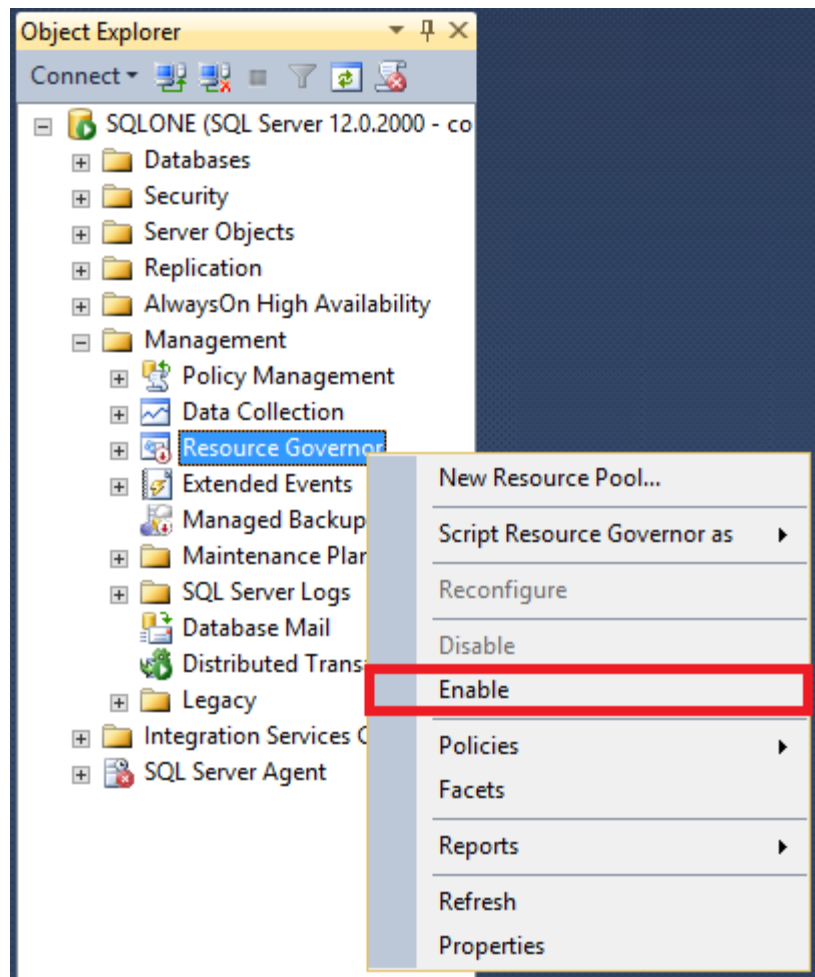
4. Log on to **SQLONE** computer as **labuser** with password **pass@word1**

Note, if you have a monitor that supports a larger screen resolution than 1024 x 768, you can change the screen resolution for the lab to go as high as 1920 x 1080. By going to a higher screen resolution, it will be easier to use SQL Server Management Studio.

5. Right click on the desktop and click on **Screen resolution**.
6. Select **1366 x 786** (a good minimum screen size for using SSMS) and click **OK**.
7. Click **Keep Changes**.
8. Resize the client **holLaunchPad Online** window for the lab to fit your screen resolution.

Enable Resource Governor

9. Open **SQL Server 2014 Management Studio** and enter the following details. Server Type: **Database Engine** Server Name: **SQLONE** Authentication Type: **Windows Authentication**. Click **Connect**
10. In the **Object Explorer**, expand the **Management** and then **Resource Governor** nodes
11. Right-click **Resource Governor** and click **Enable**



Create a Resource Pool

You will now create a Resource Pool. Resource Pools are sections of resource (CPU and memory) which can be used by one or more Workload groups (groups of applications.) SQL Server has two resource pools by default – Internal for the server itself and default for all other unassigned workloads. You will be creating a resource pool with all the default values here, and will alter it later.

1. In the **File** menu, click on **Open then File...**
2. Browse to **C:\SQLSCRIPTS\E2** and open **E2B-2#1-Create Resource Pool.sql**
3. Click **Execute** in the command bar

```
E2B-2#1-Create Res...ntoso\labuser (52) X
1  -- Create a new resource pool explicitly stating all values
2  -- As all values being given are defaults, this could alternatively be done
3  -- as CREATE RESOURCE POOL dbapool;
4  CREATE RESOURCE POOL dbapool
5  WITH
6  ( MIN_CPU_PERCENT = 0, -- how much must be assigned to this pool
7    MAX_CPU_PERCENT = 100, -- how much would be assigned if possible (note, can be exceeded if no
8    CAP_CPU_PERCENT = 100, -- cannot-be-exceeded maximum, useful for predictable billing
9    AFFINITY_SCHEDULER = AUTO,
10   MIN_MEMORY_PERCENT = 0, -- memory allocated to this pool that cannot be shared
11   MAX_MEMORY_PERCENT = 100, -- percentage total server memory which is allowed to be used by thi
12   MIN_IOPS_PER_VOLUME = 0, -- minimum number of I/O operations per second per disk volume to res
13   MAX_IOPS_PER_VOLUME = 2147483647 -- maximum. Note, this is the max allowed value for this prop
14 );
15
16
17
```

Create a Workload Group

You will now create a Workload Group. Sessions are allocated into workload groups by the Classifier function

1. In the **SQL Server 2014 Management Studio Object Explorer**, open the **File** menu, select **Open** then **File...**
2. Navigate to **C:\SQLSCRIPTS\E2** then select **E2B-2#2-Create Workgroup.sql**
3. Click **Open** to open the file

```
E2B-2#2-Create Wor...ntoso\labuser (56) X E2B-2#1-Create Res...ntoso\labuser (52)
1  -- Create a workgroup using the resource pool we just created
2  CREATE WORKLOAD GROUP dbagroup
3  WITH (
4    IMPORTANCE = MEDIUM, -- relative importance compared to other workgroups
5    REQUEST_MAX_MEMORY_GRANT_PERCENT = 50, -- how much memory a single process can request from th
6    REQUEST_MAX_CPU_TIME_SEC = 0, -- how long a single request can take without generating a CPU
7    MAX_DOP = 0, -- max degree of parallelism allowed
8    GROUP_MAX_REQUESTS = 0 -- num simultaneous events allowed, 0 means unlimited
9  )
10 USING dbapool
11 ;
12 GO
13 -- update resource governor to changes
14 ALTER RESOURCE GOVERNOR RECONFIGURE;
15 GO
16
17
18
```

4. Click **Execute** in the menu bar to run the query

Create a Classifier Function

You will now define the Classifier. The Classifier function allocates incoming queries into workgroups. You will now allocate all queries run by a DBA into the newly-created DBA workgroup

1. In the **SQL Server 2014 Management Studio** in **Object Explorer**, open the **File** menu, select **Open** then **File...**
2. Navigate to **C:\SQLSCRIPTS\E2** then select **E2B-2#3-Create Classifier.sql**
3. Click **Open** to open the file

```

1  -- make a function returning the workgroup name for the session to be allocated to
2  -- allocates new sessions to dbapool if dba in login name
3  USE Master
4  GO
5  CREATE FUNCTION fx_DBAClassifier()
6  RETURNS sysname
7  WITH SCHEMABINDING
8  AS
9  BEGIN
10
11      DECLARE @wg sysname
12  IF
13      SUSER_NAME() LIKE '%dba%'
14      SET @wg = 'dbagroup'
15      ELSE SET @wg = 'default'
16      RETURN @wg
17  END;
18  GO
19
20  -- tell the resource governor to use the function
21  ALTER RESOURCE GOVERNOR with (CLASSIFIER_FUNCTION = dbo.fx_DBAClassifier);
22  -- restart the resource governor
23  ALTER RESOURCE GOVERNOR RECONFIGURE;
24  GO
25

```

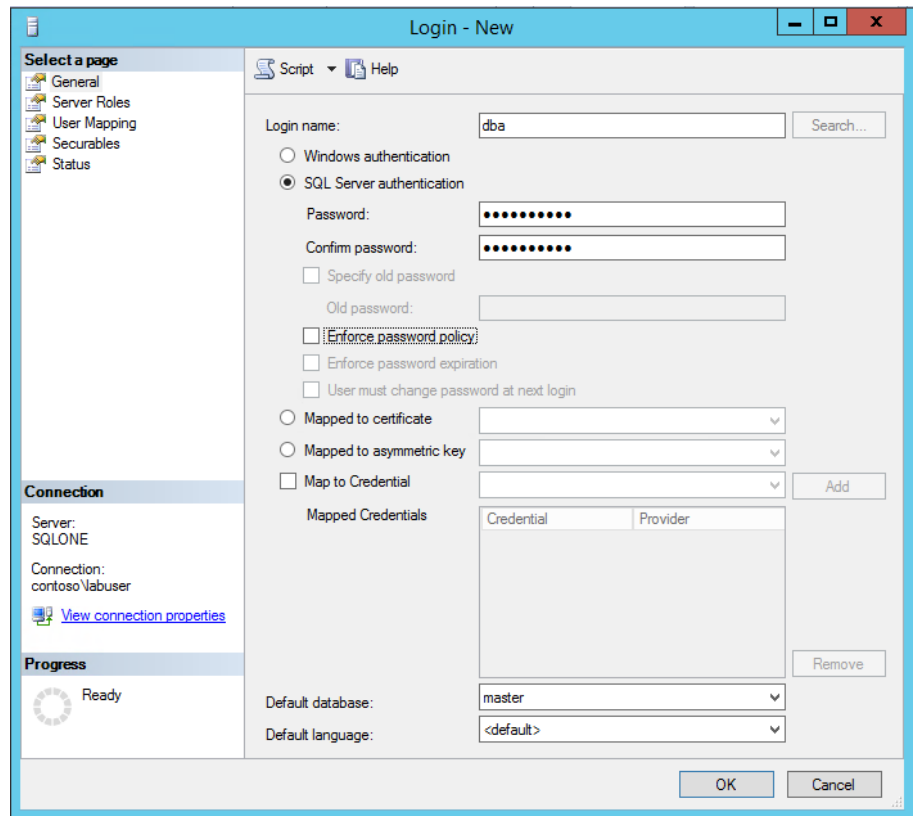
4. Click **Execute** in the menu bar to run the query

Setup a dba session for testing Resource Governor behavior

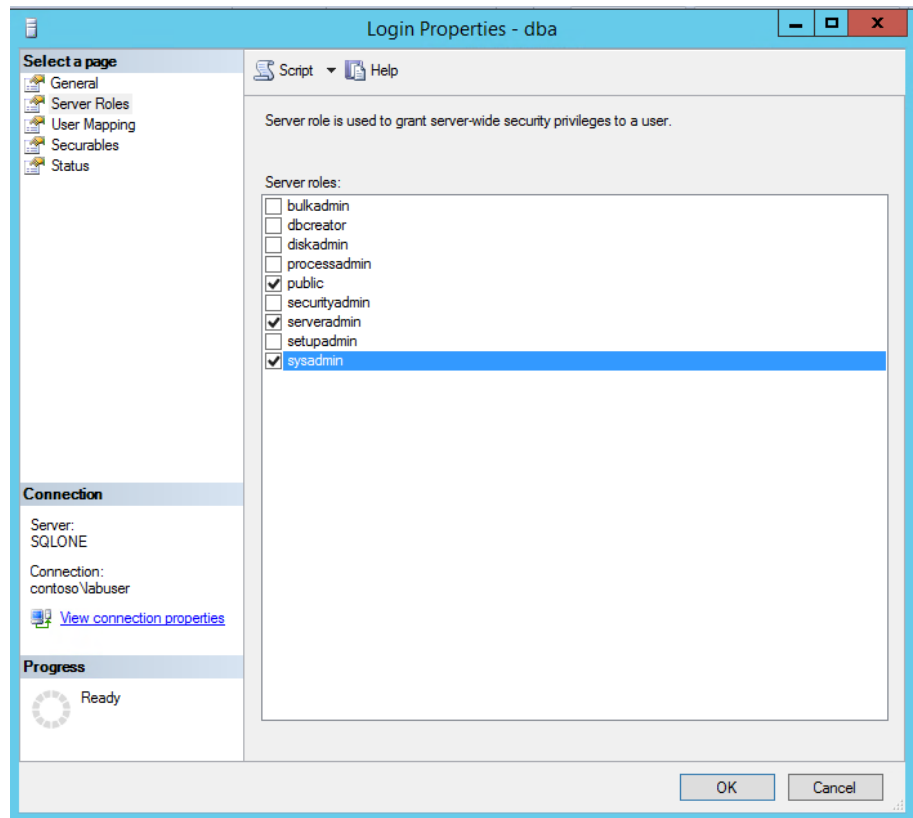
You will now start a New Session (Allocated by the Resource Governor Using Classifier Function)

1. In the **SQL Server 2014 Management Studio** in **Object Explorer**, expand the **Security**, right-click on **Logins** and select **New Login...**
2. Enter **dba** as the Login name
3. Select **SQL Server Authentication** and give the password **pass@word1** and confirm it
4. Un-tick **User must change password at next login**

*If desired, uncheck **Enforce password policy** to allow the password to be simple. This is not generally recommended policy, but the login you are creating does not have access to secure data, so does not need strong security.*



5. Click **OK** to create the login
6. Refresh the **Logins** node in the **Object Explorer** by right-clicking on it and selecting **Refresh**
7. Double-click on the **dba** node in **Logins** node
8. Click on the **Server Roles** tab and select **serveradmin** and **sysadmin**

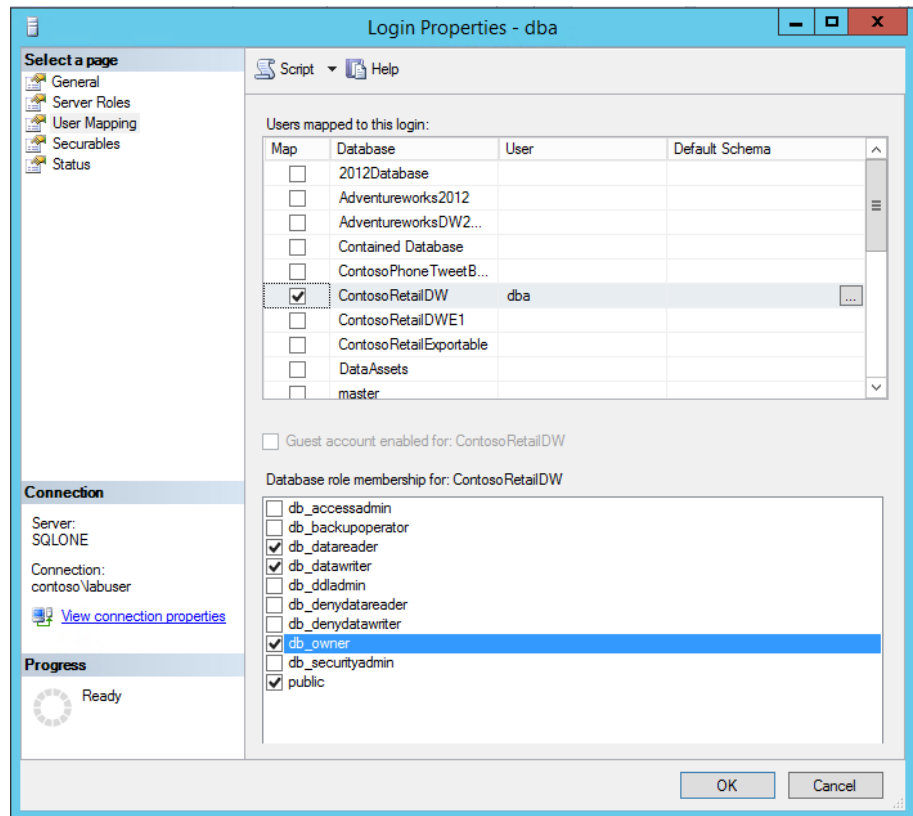


Serveradmin role allows the user to clean the server buffers and so accurately see performance, sysadmin is necessary to see what workgroup the user's session is running in.

9. Click on User Mapping

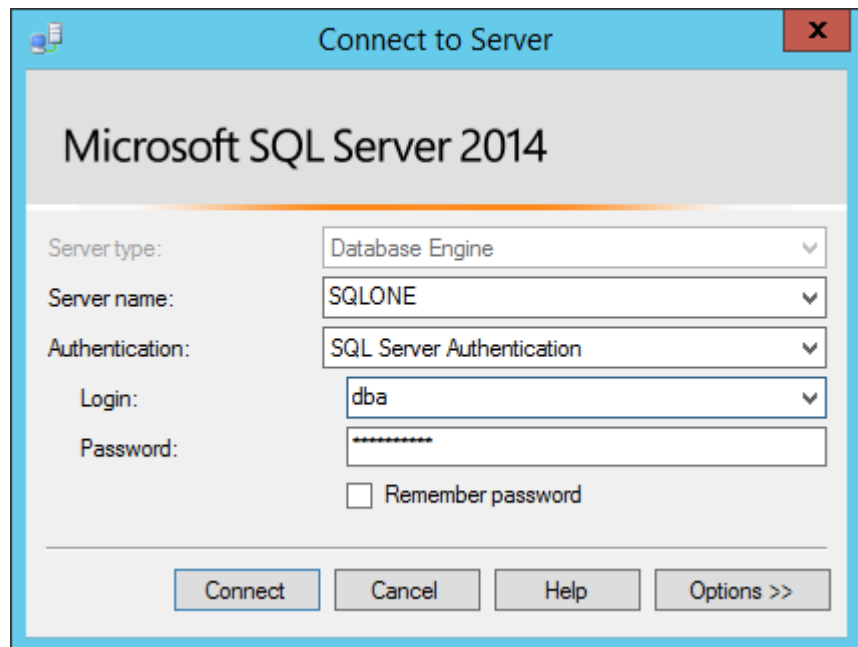
If told One or more databases are inaccessible and will not be displayed in list. click OK. This is not a problem

10. Select **ContosoRetailDW** database in the first table then check **db_datareader**, **db_datawriter**, and **db_owner**



These are the necessary permissions for the dba user to be able to do the IO-intensive read-write queries required to show the effect of the resource governance

11. Click **OK**
12. In the **Object Explorer** click on **Connect** and then **Database Engine...**
13. Select **SQL Server Authentication** and enter **dba** as the username and the password you defined earlier

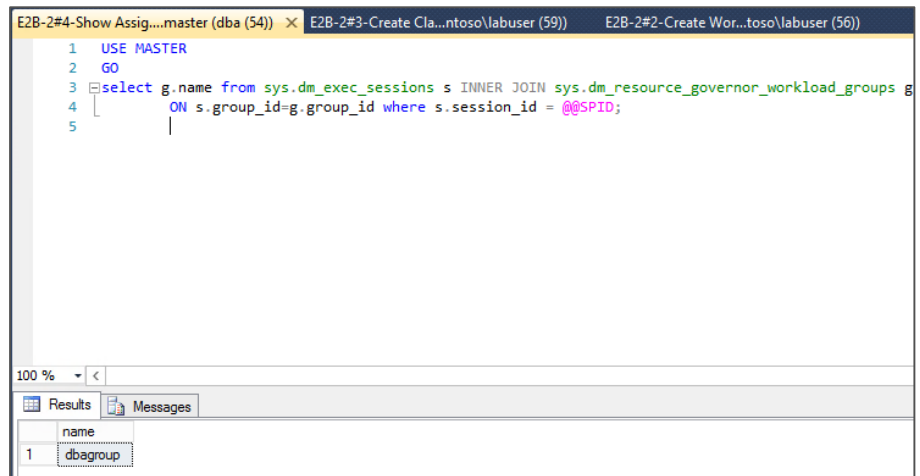


14. Click **Connect**

Display Resource Governor Workload Group assignment for session

The following will show how the Resource Governor allocated the session to a workgroup

1. Ensure the new connection you just created is selected (highlighted in blue) in the **Object Explorer**
2. From the **File** menu, select **Open** and then **File...**
3. Navigate to **C:\SQLSCRIPTS\E2** then open **E2B-2#4-Show Assigned Workgroup.sql**
4. Click **Execute** in the command bar to run the query

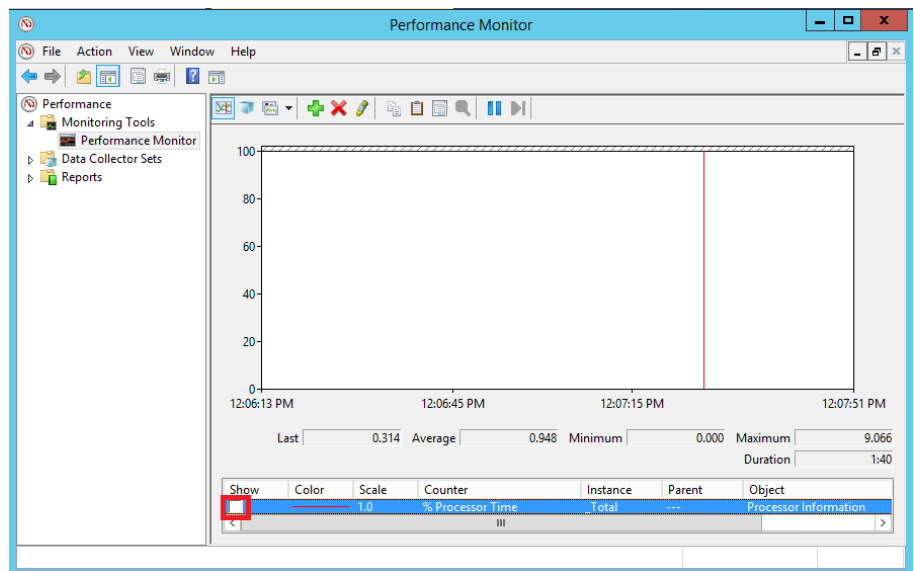


The result is the name of the workload group the current session is assigned to, and should say **dbagroup**. Do not close Management Studio or the connection.


Testing the performance impact of the workload group assignment

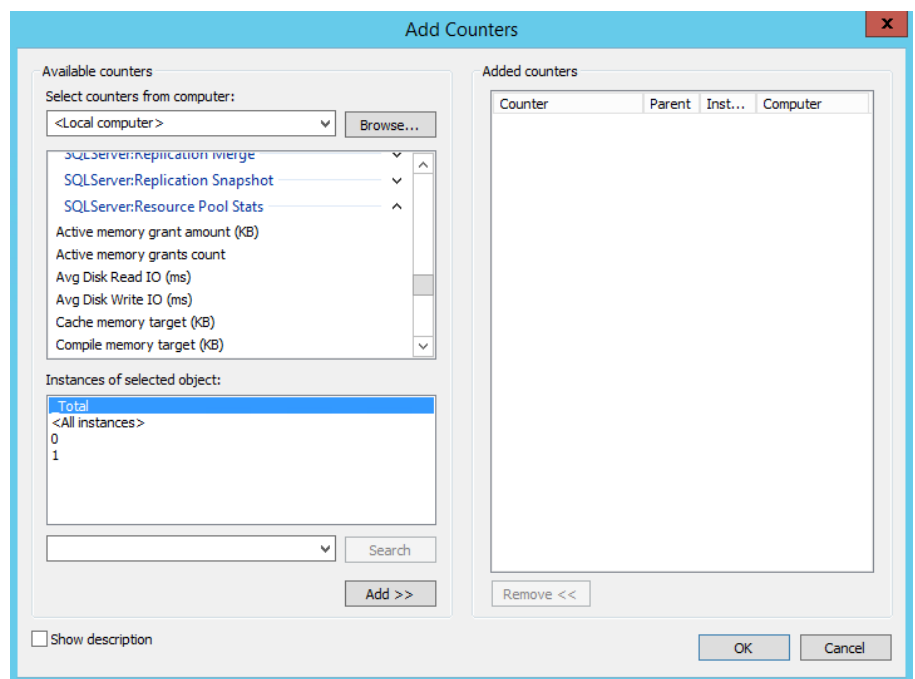
Currently, the dbapool has default parameters, so performance will effectively be as if the session was in the default group

1. Open the Windows Start screen and then type **Performance Monitor**
2. Click on **Performance Monitor** in the list of search results this produces
3. In the left-hand pane, expand **Monitoring Tools** and click on **Performance Monitor**
4. In the table at the bottom, uncheck the **Show** checkbox for the **%Processor Time** counter

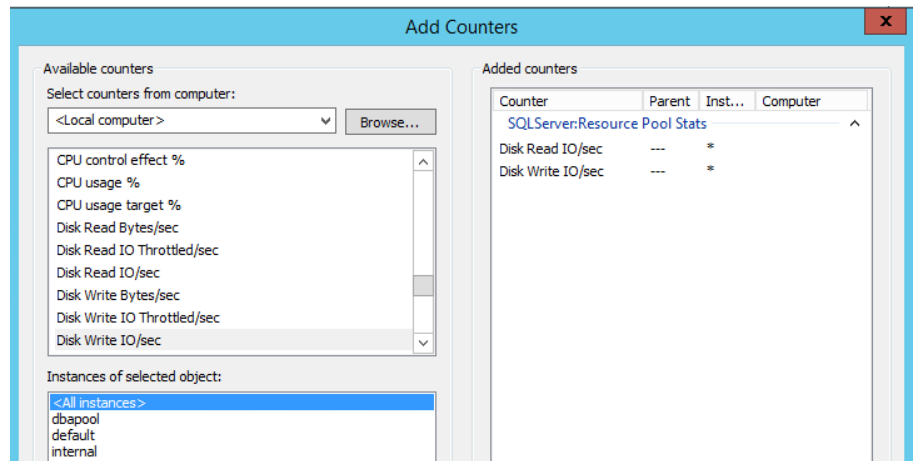


This will stop the processor time counter obscuring the IO counters we are about to add

5. In the **Performance Monitor** command bar, click 
6. Scroll down to **SQL Server:Resource Pool Stats** and expand it by clicking the down arrow



7. Select **Disk Read IO/sec** and click the **Add>>**
8. Select **Disk Write IO/sec** and click the **Add>>**



9. Click **OK**

Do not close the Performance Monitor

Execute the test query

1. Return to **SQL Server 2014 Management Studio**
2. Ensure the **SQLONE** connection you opened using the **dba** connection is selected in the **Object Explorer**
3. From the **File** menu, select **Open** then **File...**
4. Navigate to **C:\SQLSCRIPTS\E2** then select **E2B-2#5-Test Query.sql**
5. Click **Open**

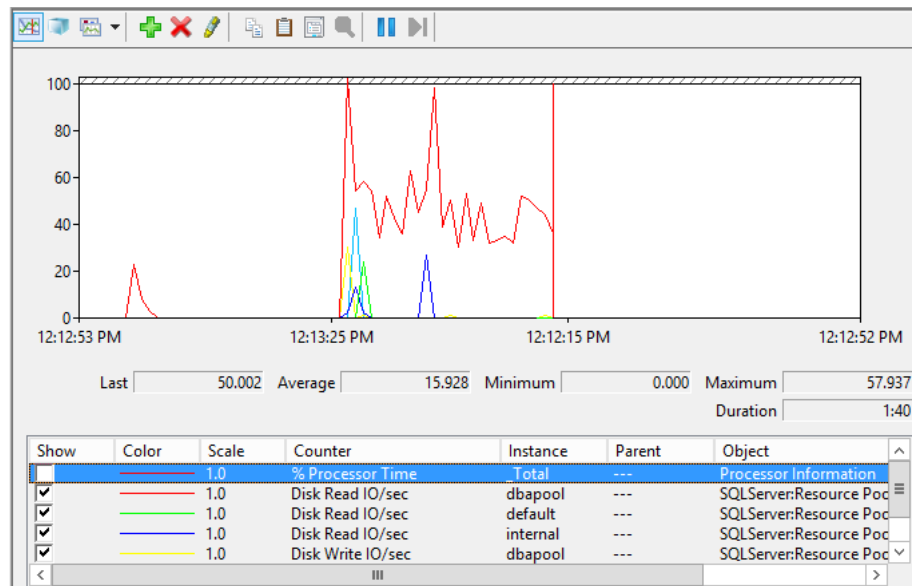
```

1 -- NOTE: reporting service performance is not affected by resource governance
2 -- so we must show improvements here by a query
3 -- clear the buffers
4 CHECKPOINT
5 GO
6 DBCC DROPCLEANBUFFERS
7 DBCC FREEPROCCACHE
8 GO
9
10 USE [ContosoRetailDW]
11 GO
12
13 -- make a temporary table to do IO-intensive write operations on
14 CREATE TABLE Temp (ID int);
15 GO
16
17 -- do the IO-intensive write operation
18 INSERT INTO Temp
19 select top 10000000 ProductKey from FactOnlineSales;
20 GO
21
22 -- clean up
23 DROP TABLE Temp

```

6. Click **Execute**

- Switch to the **Performance Monitor** window again by clicking  in the taskbar



See how the disk write and read counters spiked when you ran the query. Do not close the Performance Monitor

- Close editor for select **E2B-2#5-Test Query.sql**

Alter the Resource Governor Resource Pool dbapool to limit MAX_IOPS_PER_VOLUME

- Return to **SQL Server 2014 Management Studio** and ensure your **dba** connection is selected in the **Object Explorer**
- In the File menu, click on **Open** then **File...**
- Browse to **C:\SQLSCRIPTS\E2** and select **E2B-2#6-Alter Resource Pool.sql**

This query lowers the allowed IO transactions per second and so makes any queries run in this resource pool run with fewer resources, meaning other resource pools will have more whenever there is competition

- Click **Open**

```

E2B-2#6-Alter Reso...RetailDW (dba (60))  E2B-2#5-Test Query...etailDW (dba (55))  E2B-2#4-Show Assign...master (dba (54))
1  USE [ContosoRetailDW]
2  GO
3  ALTER RESOURCE POOL dbapool
4  WITH (
5      MAX_IOPS_PER_VOLUME = 50 -- This is a very low IOPS, so should have a noticeable effect
6  );
7  GO
8  ALTER RESOURCE GOVERNOR RECONFIGURE;
9

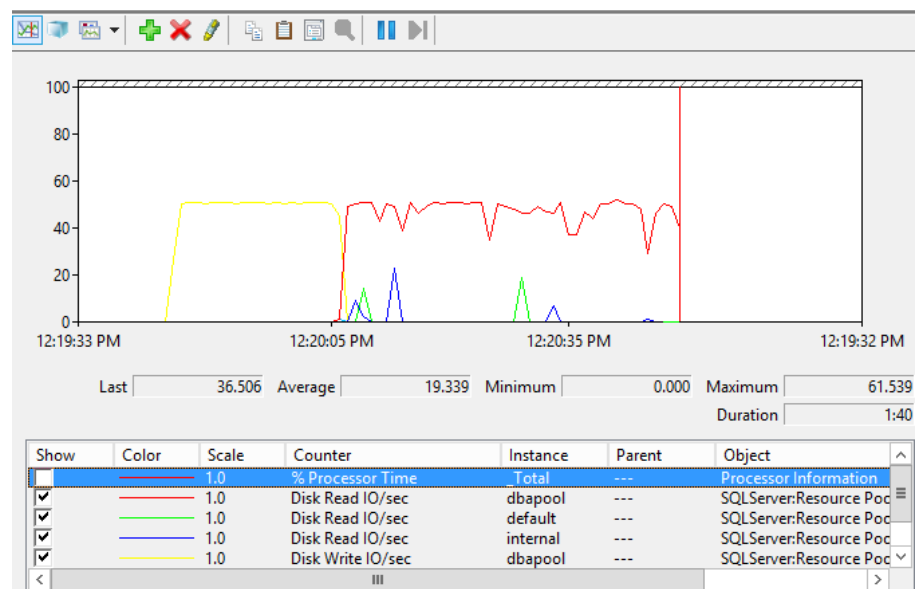
```

- Click **Execute**

Check the effect of the of Altered Rules

1. From the **File** menu, click **Open** then **File...**
2. Navigate to **C:\SQLSCRIPTS\E2** then select **E2B-2#5-Test Query.sql**
3. Click **Open**
4. Click **Execute** in the command bar to run the query

Return to the Performance Monitor window again and note the spike is wider and shorter than when you previously ran the query. Notice that the maximum value for the read and write IO is limited to approximately 50 as well.



5. Close everything without saving

NOTE: Resource management using Resource Governor does have some limitations

- Resource management is limited to the SQL Server Database Engine. Resource Governor cannot be used for Analysis Services, Integration Services, and Reporting Services
- There is no workload monitoring or workload management between SQL Server instances
- Resource Governor can manage OLTP workloads but these types of queries, which are typically very short in duration, are not always on the CPU long enough to apply bandwidth controls. This may skew in the statistics returned for CPU usage %

- The ability to govern physical IO only applies to user operations and not system tasks. System tasks include write operations to the transaction log and Lazy Writer IO operations. The Resource Governor applies primarily to user read operations because most write operations are typically performed by system tasks
- You cannot set IO thresholds on the internal resource pool

You have learned how to set up resource governance to ensure queries run by the dba user will always have resources to use.

Terms of use

© 2014 Microsoft Corporation. All rights reserved.

By using this Hands-on Lab, you agree to the following terms:

The technology/functionality described in this Hands-on Lab is provided by Microsoft Corporation in a “sandbox” testing environment for purposes of obtaining your feedback and to provide you with a learning experience. You may only use the Hands-on Lab to evaluate such technology features and functionality and provide feedback to Microsoft. You may not use it for any other purpose. You may not modify, copy, distribute, transmit, display, perform, reproduce, publish, license, create derivative works from, transfer, or sell this Hands-on Lab or any portion thereof.

COPYING OR REPRODUCTION OF THE HANDS-ON LAB (OR ANY PORTION OF IT) TO ANY OTHER SERVER OR LOCATION FOR FURTHER REPRODUCTION OR REDISTRIBUTION IS EXPRESSLY PROHIBITED.

THIS HANDS-ON LAB PROVIDES CERTAIN SOFTWARE TECHNOLOGY/PRODUCT FEATURES AND FUNCTIONALITY, INCLUDING POTENTIAL NEW FEATURES AND CONCEPTS, IN A SIMULATED ENVIRONMENT WITHOUT COMPLEX SET-UP OR INSTALLATION FOR THE PURPOSE DESCRIBED ABOVE. THE TECHNOLOGY/CONCEPTS REPRESENTED IN THIS HANDS-ON LAB MAY NOT REPRESENT FULL FEATURE FUNCTIONALITY AND MAY NOT WORK THE WAY A FINAL VERSION MAY WORK. WE ALSO MAY NOT RELEASE A FINAL VERSION OF SUCH FEATURES OR CONCEPTS. YOUR EXPERIENCE WITH USING SUCH FEATURES AND FUNCTIONALITY IN A PHYSICAL ENVIRONMENT MAY ALSO BE DIFFERENT.

FEEDBACK. If you give feedback about the technology features, functionality and/or concepts described in this Hands-on Lab to Microsoft, you give to Microsoft, without charge, the right to use, share and commercialize your feedback in any way and for any purpose. You also give to third parties, without charge, any patent rights needed for their products, technologies and services to use or interface with any specific parts of a Microsoft software or service that includes the feedback. You will not give feedback that is subject to a license that requires Microsoft to license its software or documentation to third parties because we include your feedback in them. These rights survive this agreement.

MICROSOFT CORPORATION HEREBY DISCLAIMS ALL WARRANTIES AND CONDITIONS WITH REGARD TO THE HANDS-ON LAB, INCLUDING ALL WARRANTIES AND CONDITIONS OF MERCHANTABILITY, WHETHER EXPRESS, IMPLIED OR STATUTORY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. MICROSOFT DOES NOT MAKE ANY ASSURANCES OR

REPRESENTATIONS WITH REGARD TO THE ACCURACY OF THE RESULTS, OUTPUT THAT DERIVES FROM USE OF THE VIRTUAL LAB, OR SUITABILITY OF THE INFORMATION CONTAINED IN THE VIRTUAL LAB FOR ANY PURPOSE.

DISCLAIMER

This lab contains only a portion of new features and enhancements in Microsoft SQL Server 2014. Some of the features might change in future releases of the product. In this lab, you will learn about some, but not all, new features.