Automating system administration tasks – Part3

 michaelfirsov.wordpress.com/automating-system-administration-tasks-part3
September 15, 2017

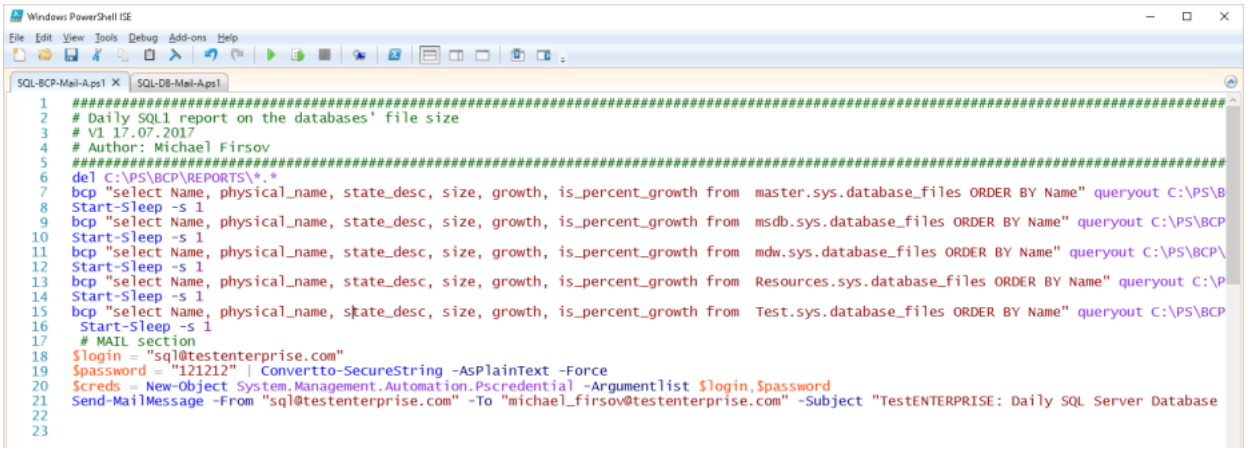
Part2

III SQL Server

What I want to know about my SQL server:

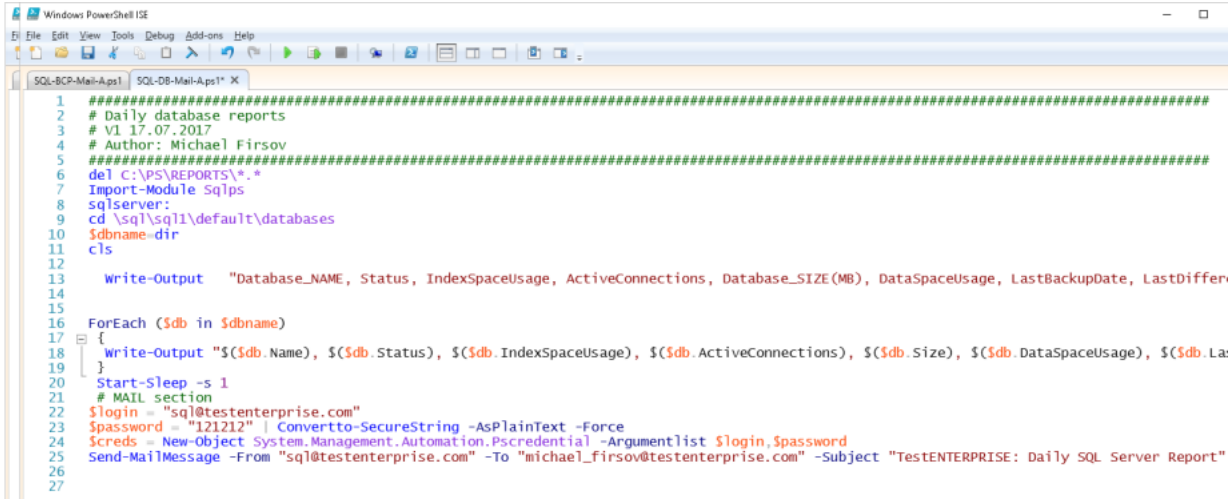
1. **The size of database and log files**
2. ***Status, IndexSpaceUsage, ActiveConnections, Database_SIZE(MB), DataSpaceUsage, LastBackupDate, LastDifferentialBackupDate, LastLogBackupDate, LogReuseWaitStatus*** of each database
3. **Did the backup of my databases complete successfully?**
4. **What is the overall health of my SQL Server infrastructure?**

1-2) **The scripts:**
Report on the databases' file size – DBFileSizeReport-ps1
(please rename the *.docx* file into *.ps1* after downloading and make any changes you need before using the scripts)



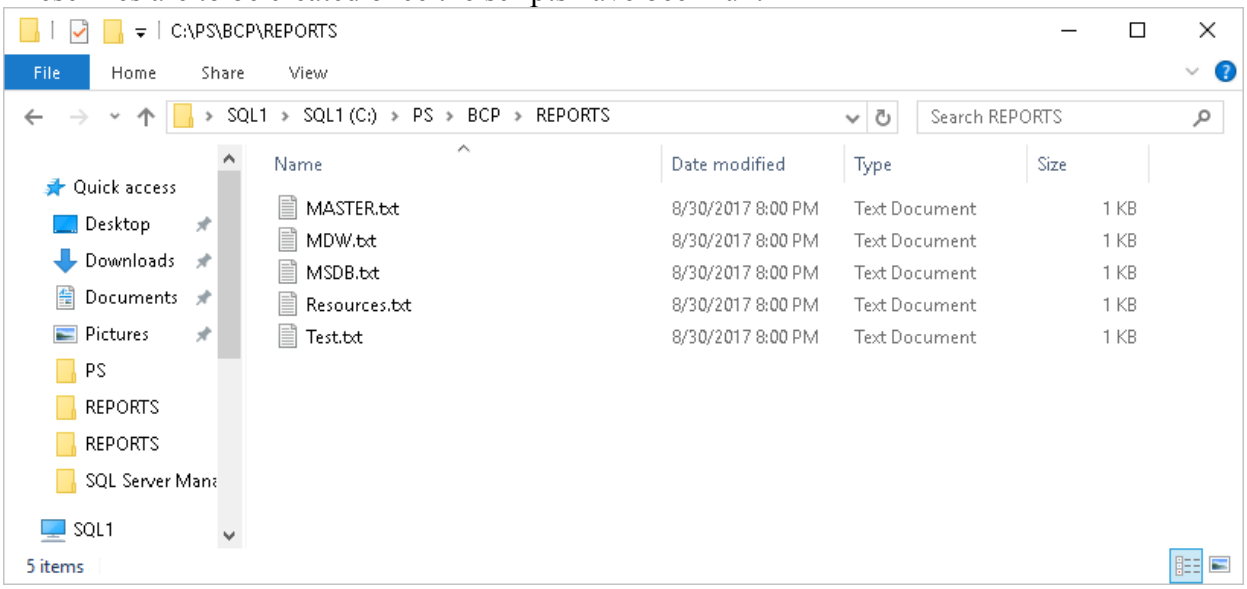
The DBreports.ps1 scripts generate reports only for non-system databases. To create report on all databases please download dbfilesizereportv2.

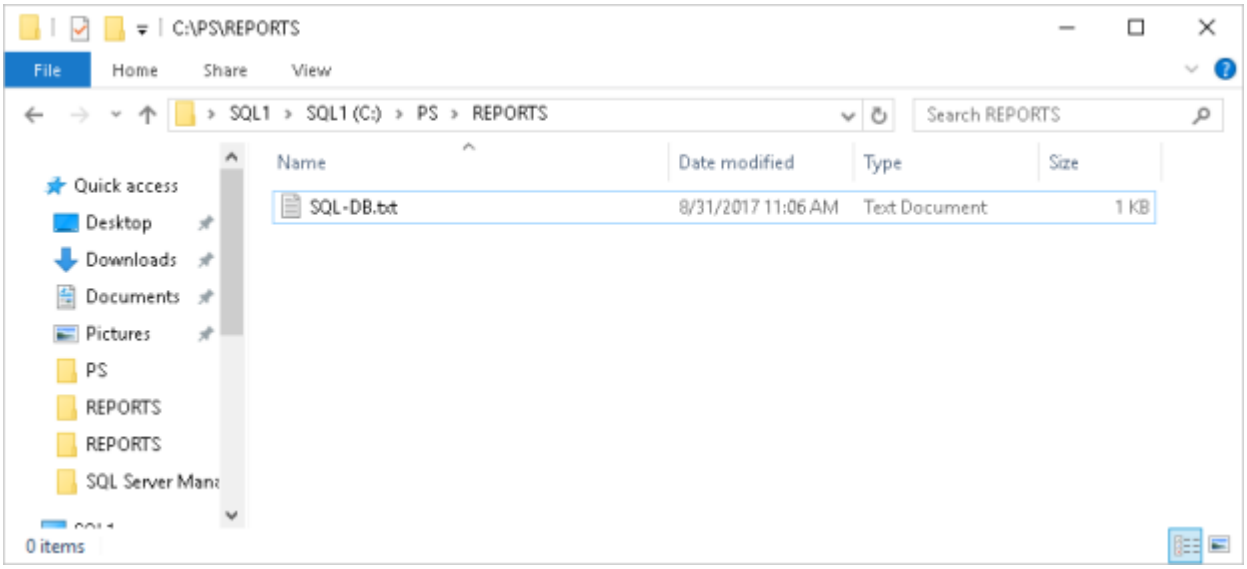
Database reports – DBreportV2



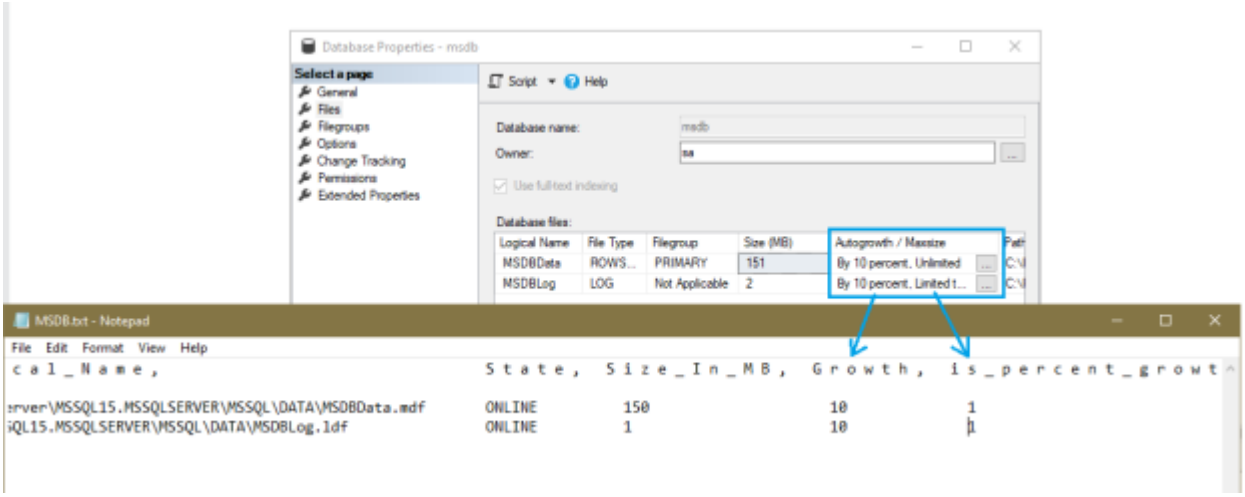
```
##########################################################################################################################################
# Daily database reports
# V1 17.07.2017
# Author: Michael Firsov
##########################################################################################################################################
del C:\PS\REPORTS\*.*
Import-Module Sqlps
sqlserver:
cd \sql\sql1\default\databases
$dbname=dir
cls

  Write-Output   "Database_NAME, Status, IndexSpaceUsage, ActiveConnections, Database_SIZE(MB), DataSpaceUsage, LastBackupDate, LastDiffer

ForEach ($db in $dbname)
 {
  Write-Output "$($db.Name), $($db.Status), $($db.IndexSpaceUsage), $($db.ActiveConnections), $($db.Size), $($db.DataSpaceUsage), $($db.La
 }
 Start-Sleep -s 1
 # MAIL section
$login = "sql@testenterprise.com"
$password = "121212" | Convertto-SecureString -AsPlainText -Force
$creds = New-Object System.Management.Automation.Pscredential -Argumentlist $login,$password
Send-MailMessage -From "sql@testenterprise.com" -To "michael_firsov@testenterprise.com" -Subject "TestENTERPRISE: Daily SQL Server Report"
```

These files are to be created once the scripts have been run:

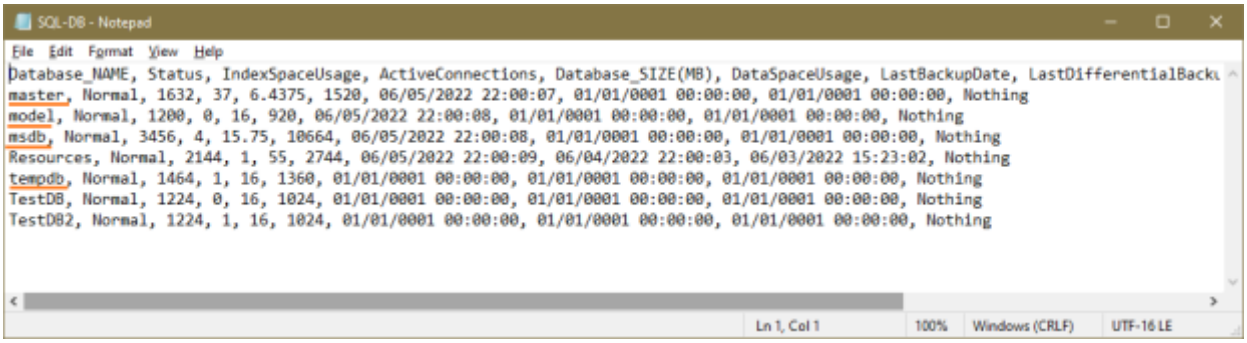




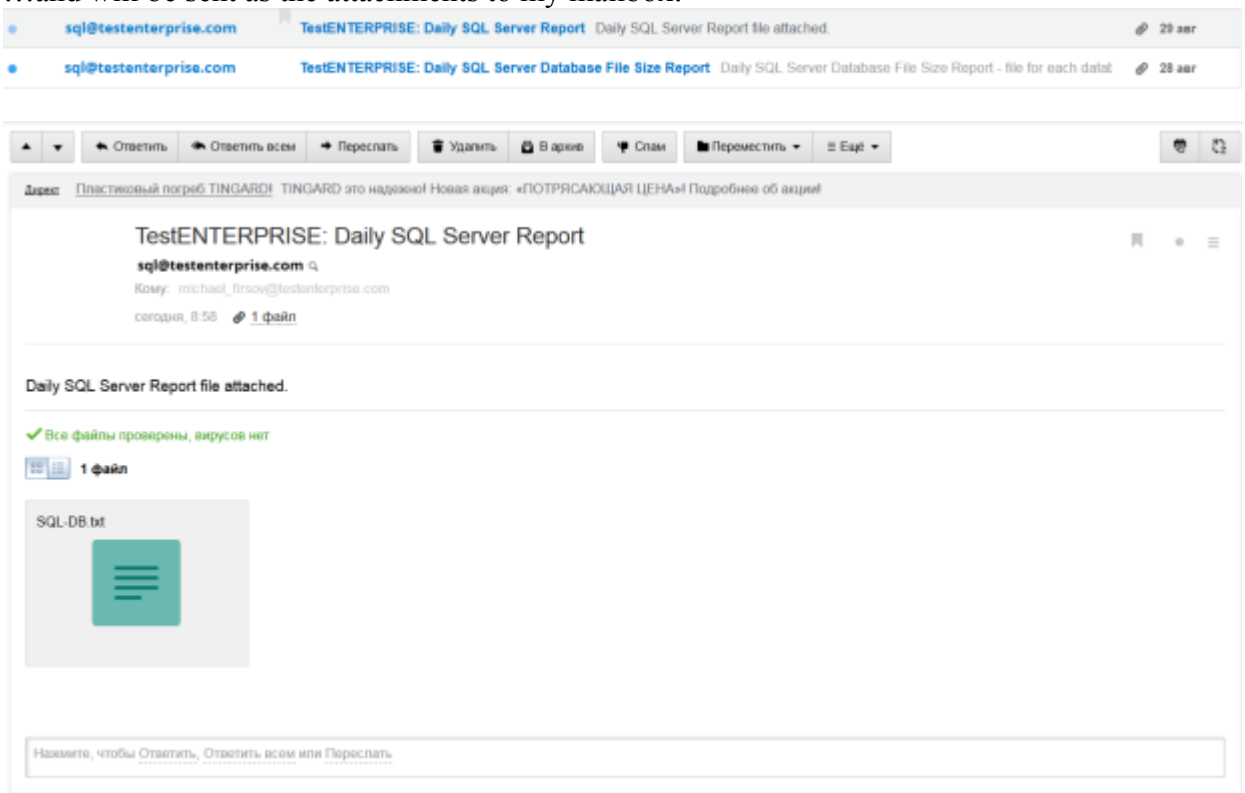
For dbfilesizereportV2 reports (I placed the header line in the beginnig of the file manually by cutting/pasting it for readability – the script inserts it at the bottom of the file):

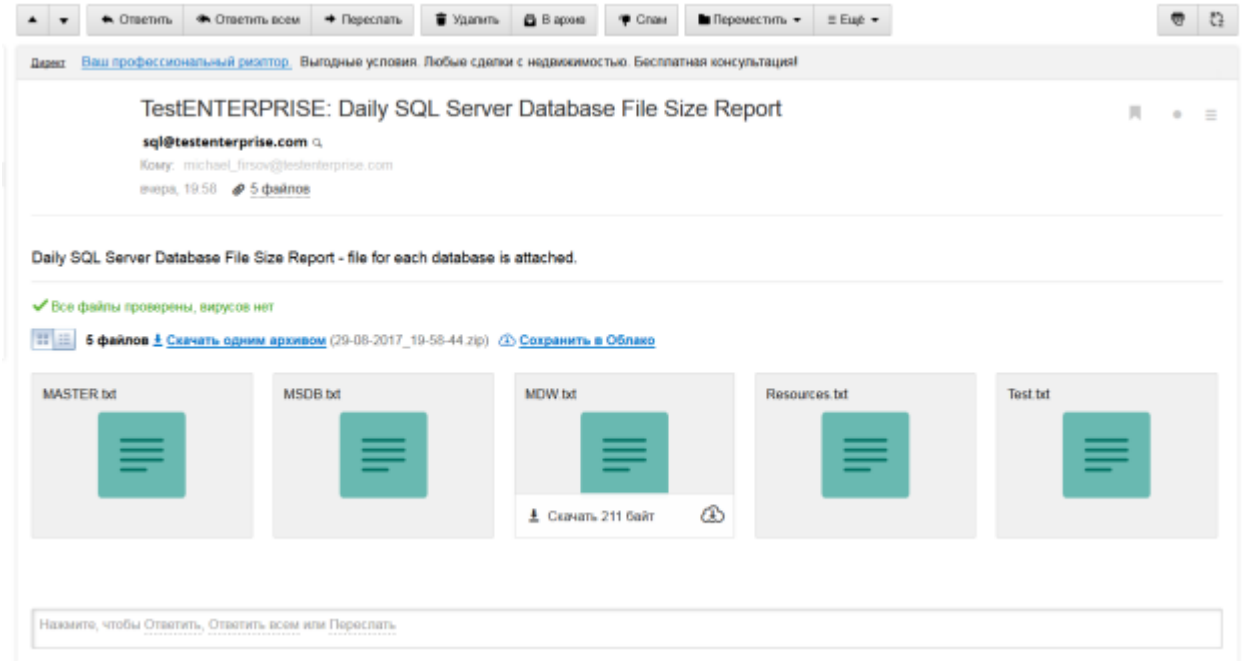


For DBreportsV2.ps1:



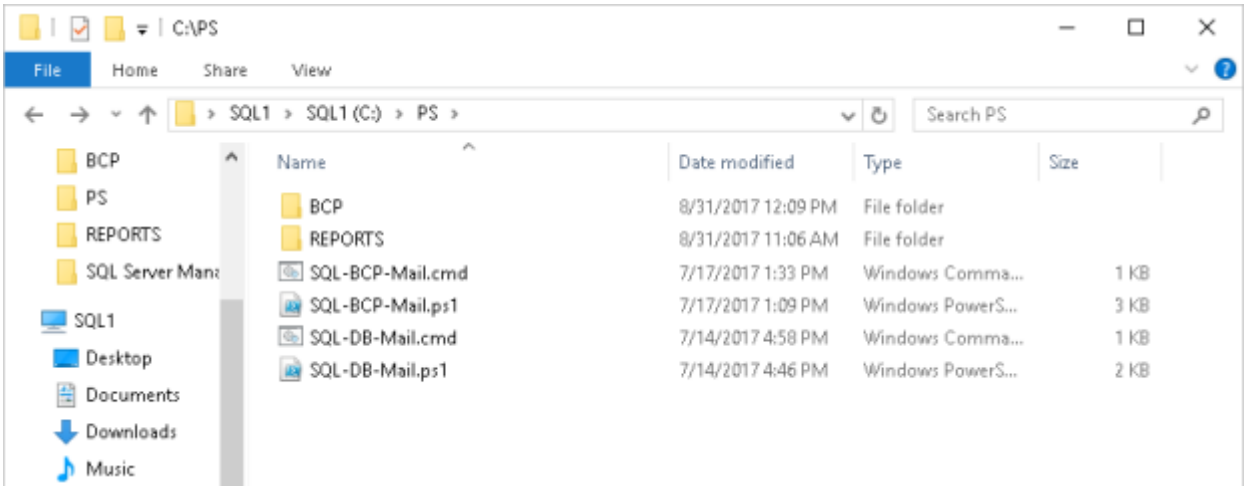
…and will be sent as the attachments to my mailbox:

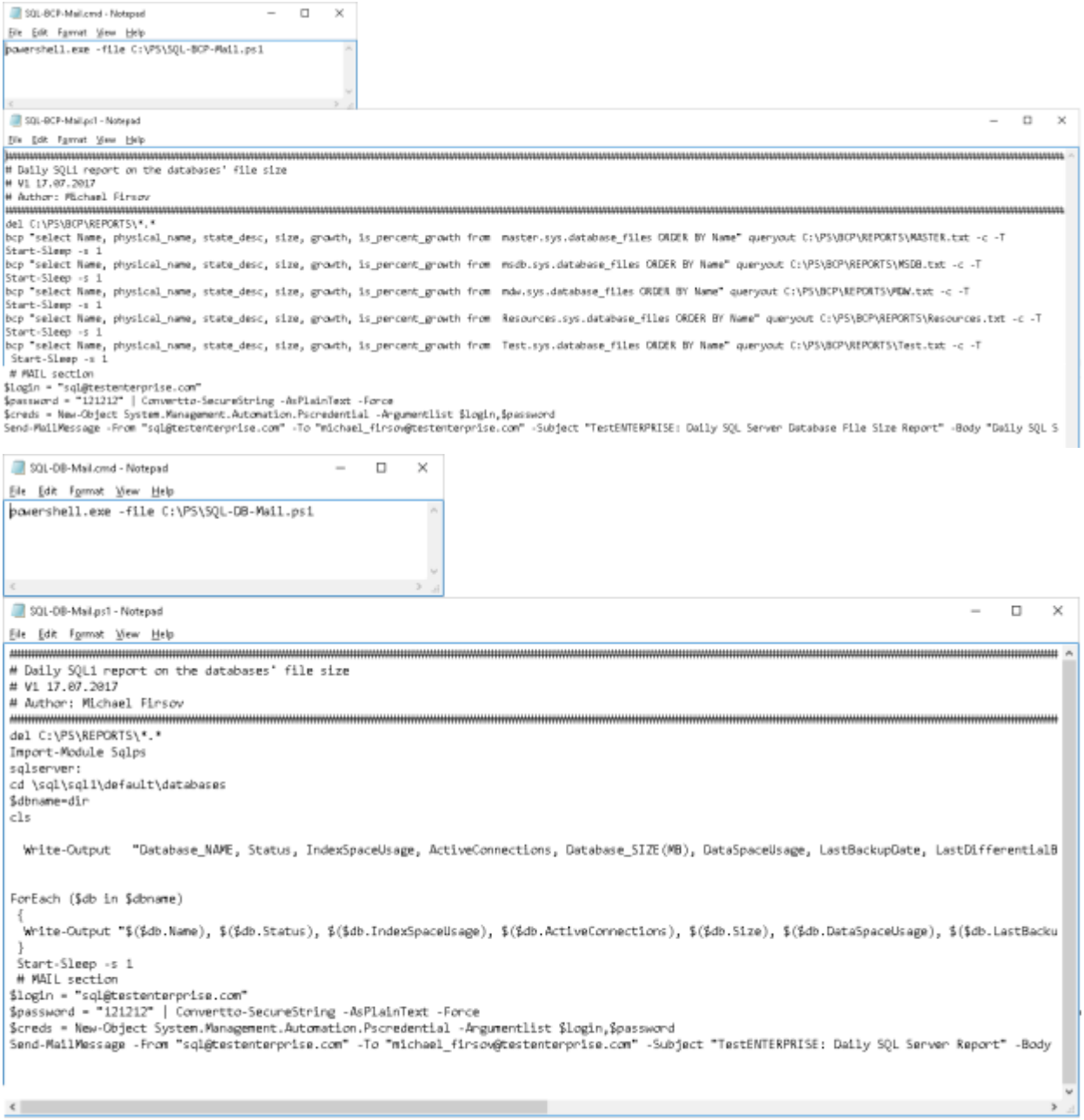




After reading through these reports an administrator will be aware of the most common parameters of SQL Server databases.

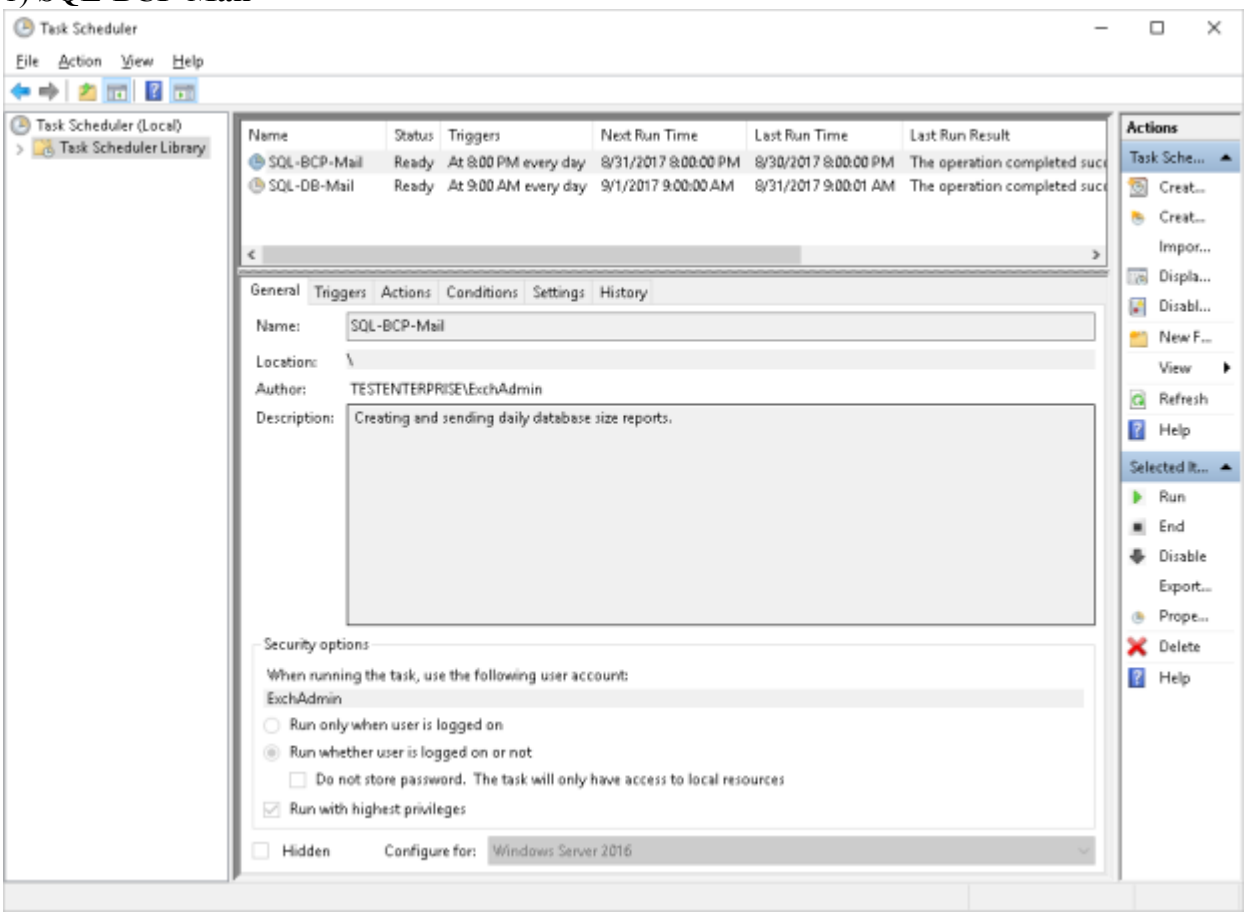
The last step is to create a couple of scheduled tasks to get the reports via email.
Here are the files that will be used in the **Task Schedular** (*.cmd* files will run corresponding *.ps1* files – SQL-BCP-Mail-cmd and SQL-DB-Mail-cmd):

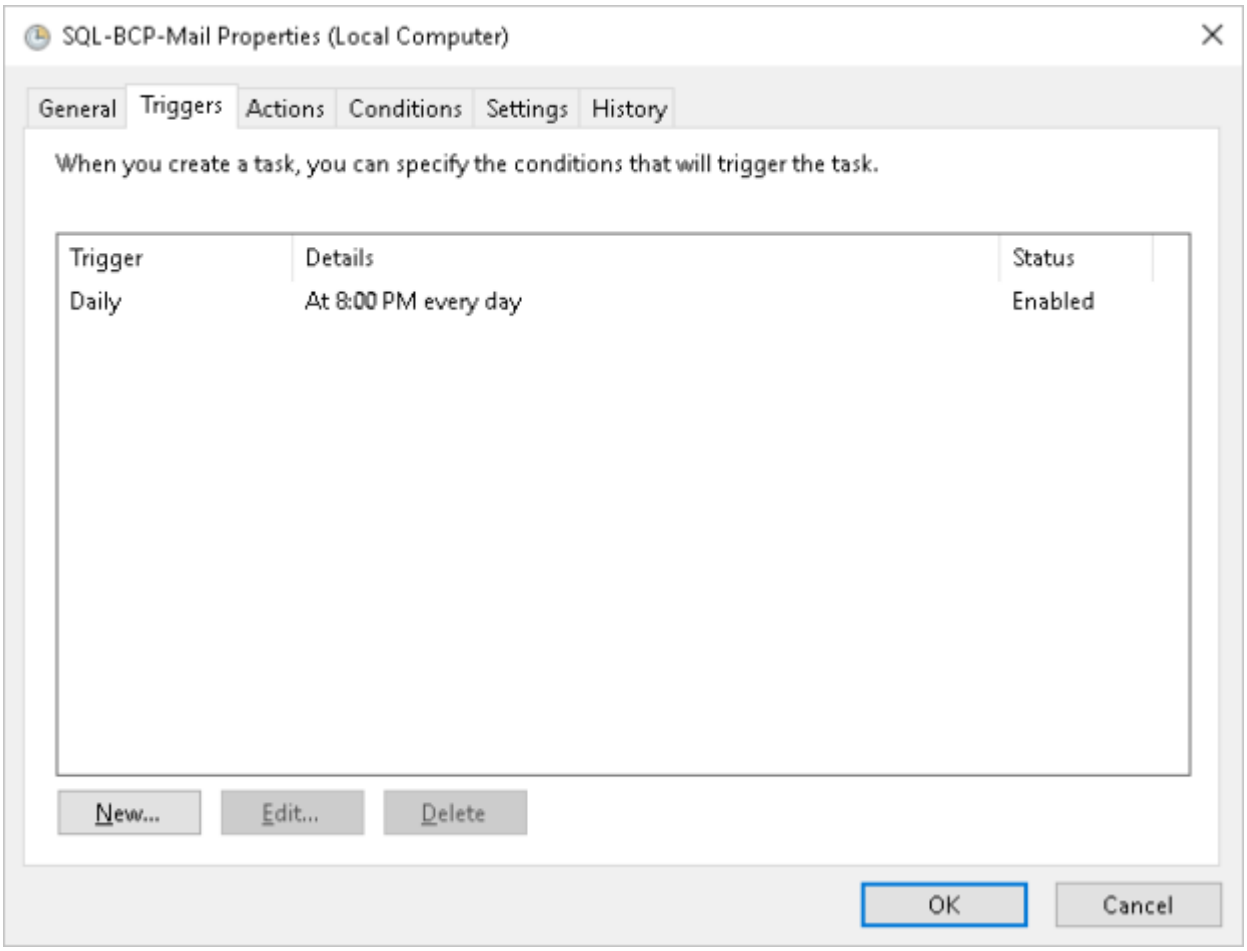


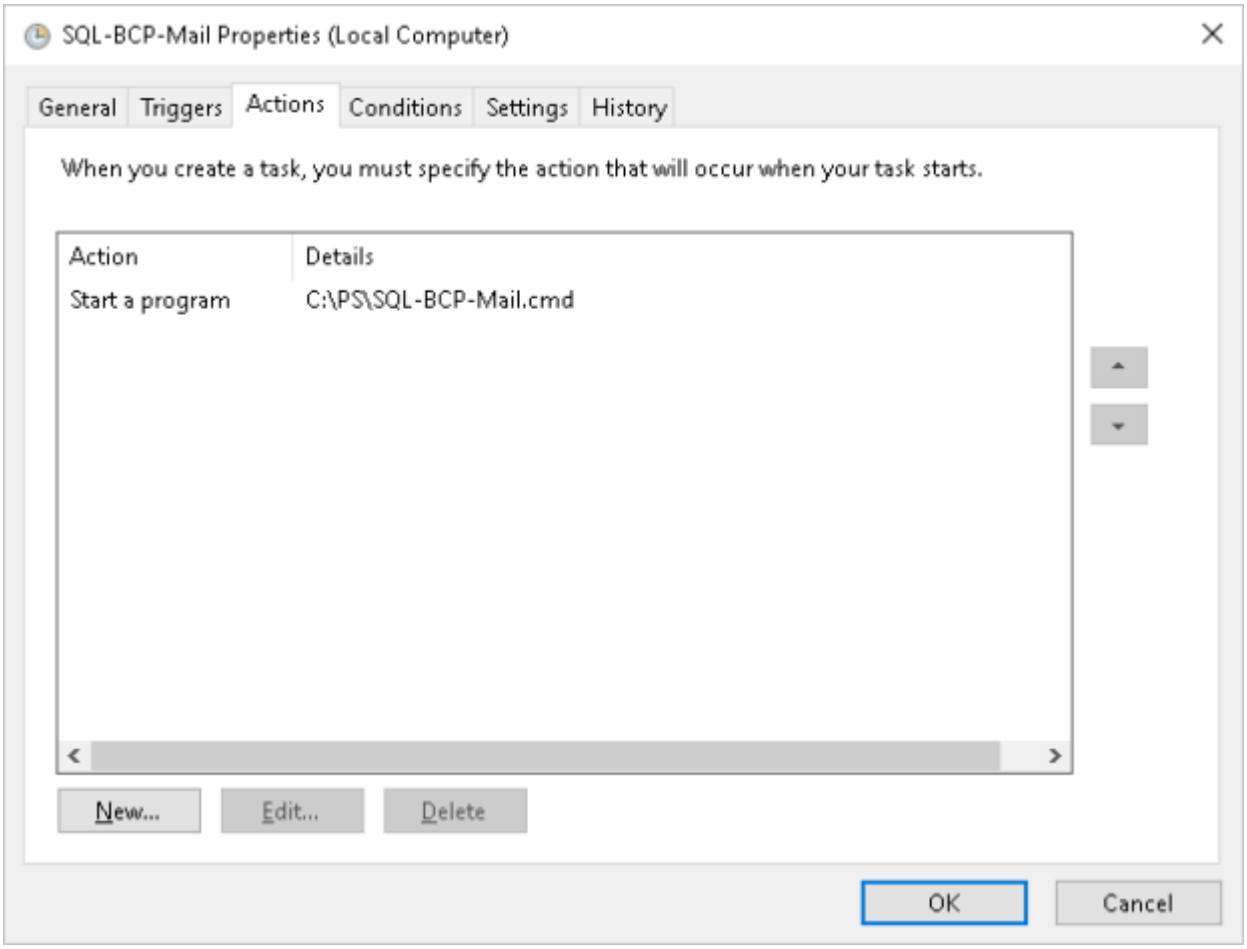


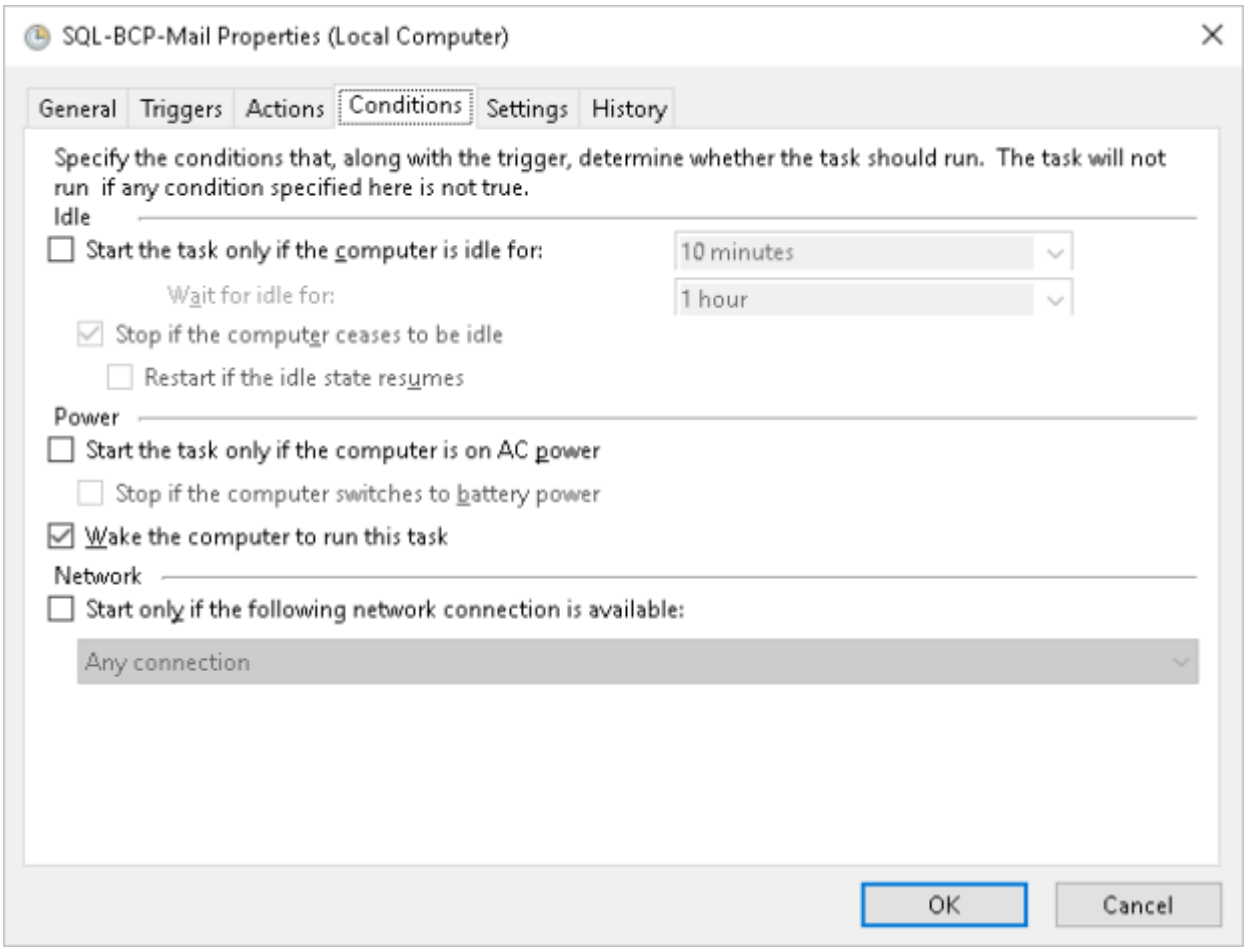
I’ve created the following two tasks:

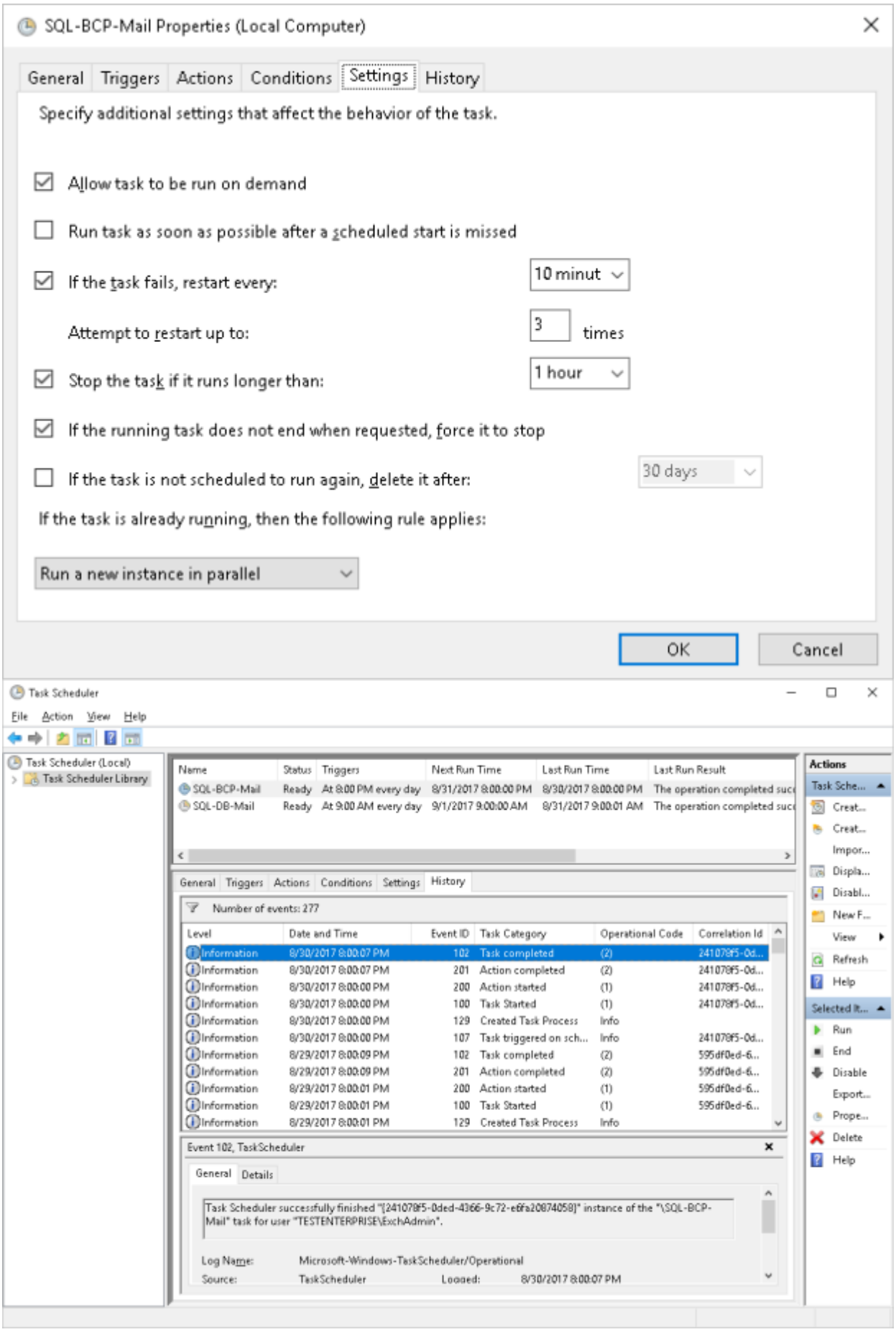
1) **SQL-BCP-Mail**



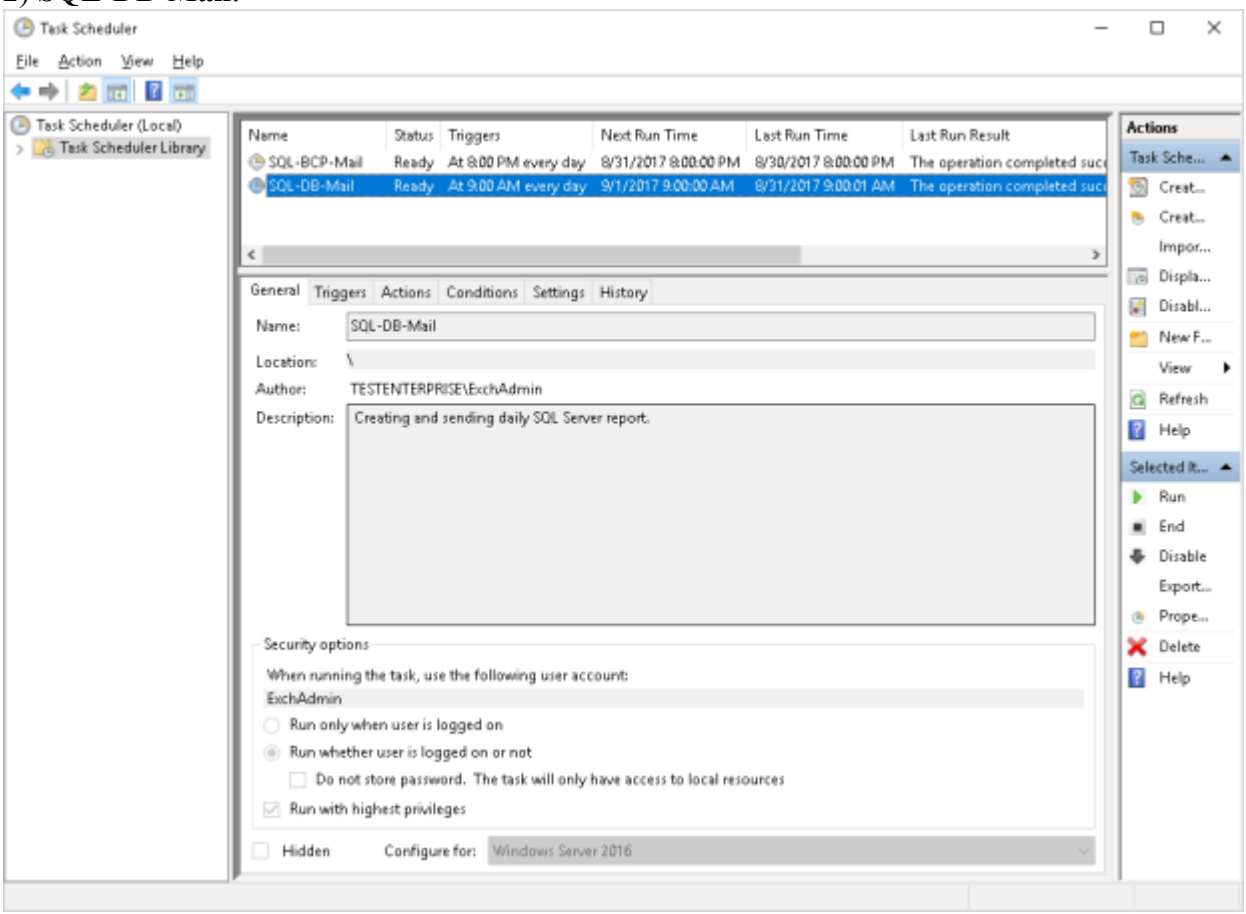


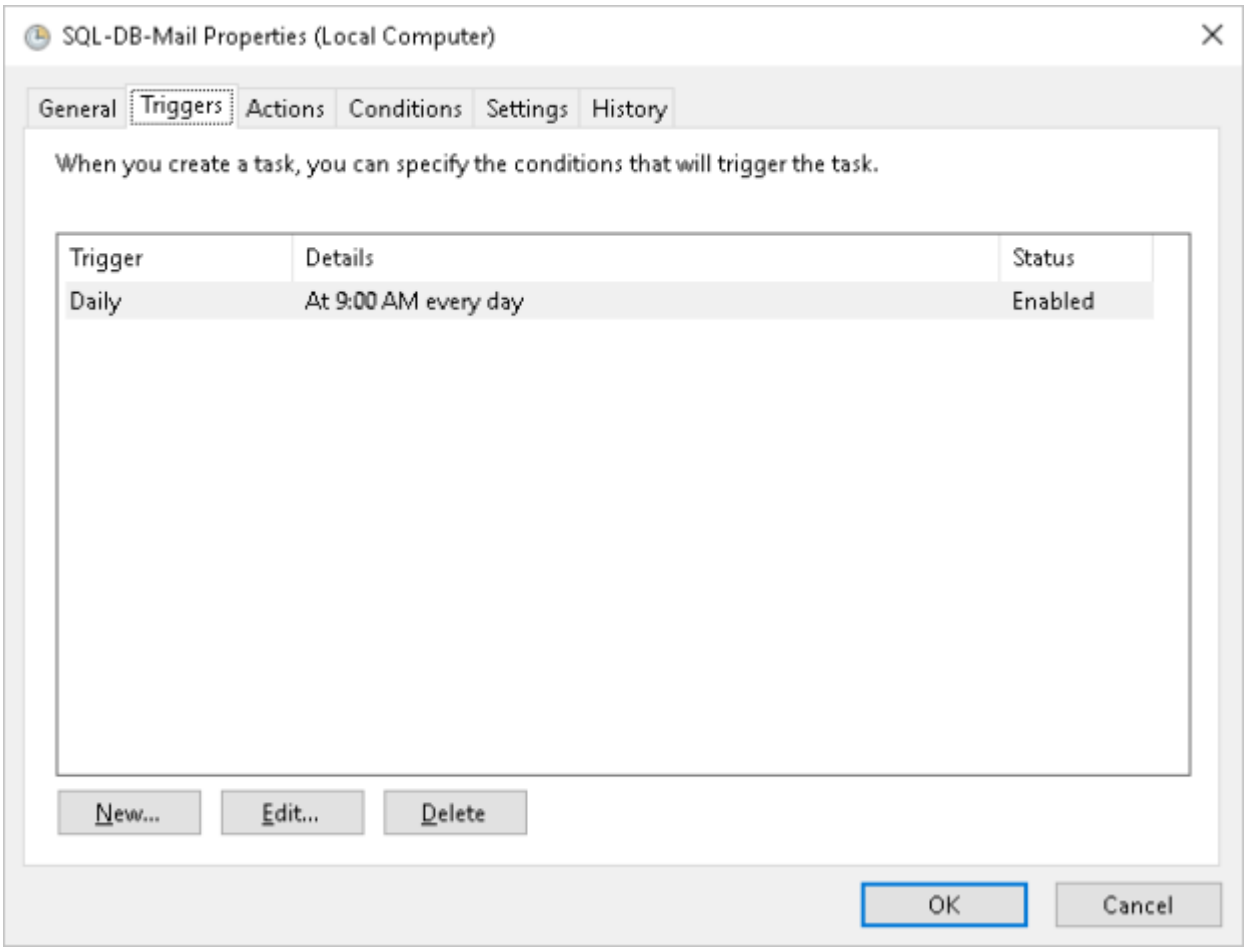

SQL-BCP-Mail Properties (Local Computer)
General
Triggers
Actions
Conditions
Settings
History
When you create a task, you must specify the action that will occur when your task starts.
Action
Details
Start a program
C:\PS\SQL-BCP-Mail.cmd
New...
Edit...
Delete
OK
Cancel

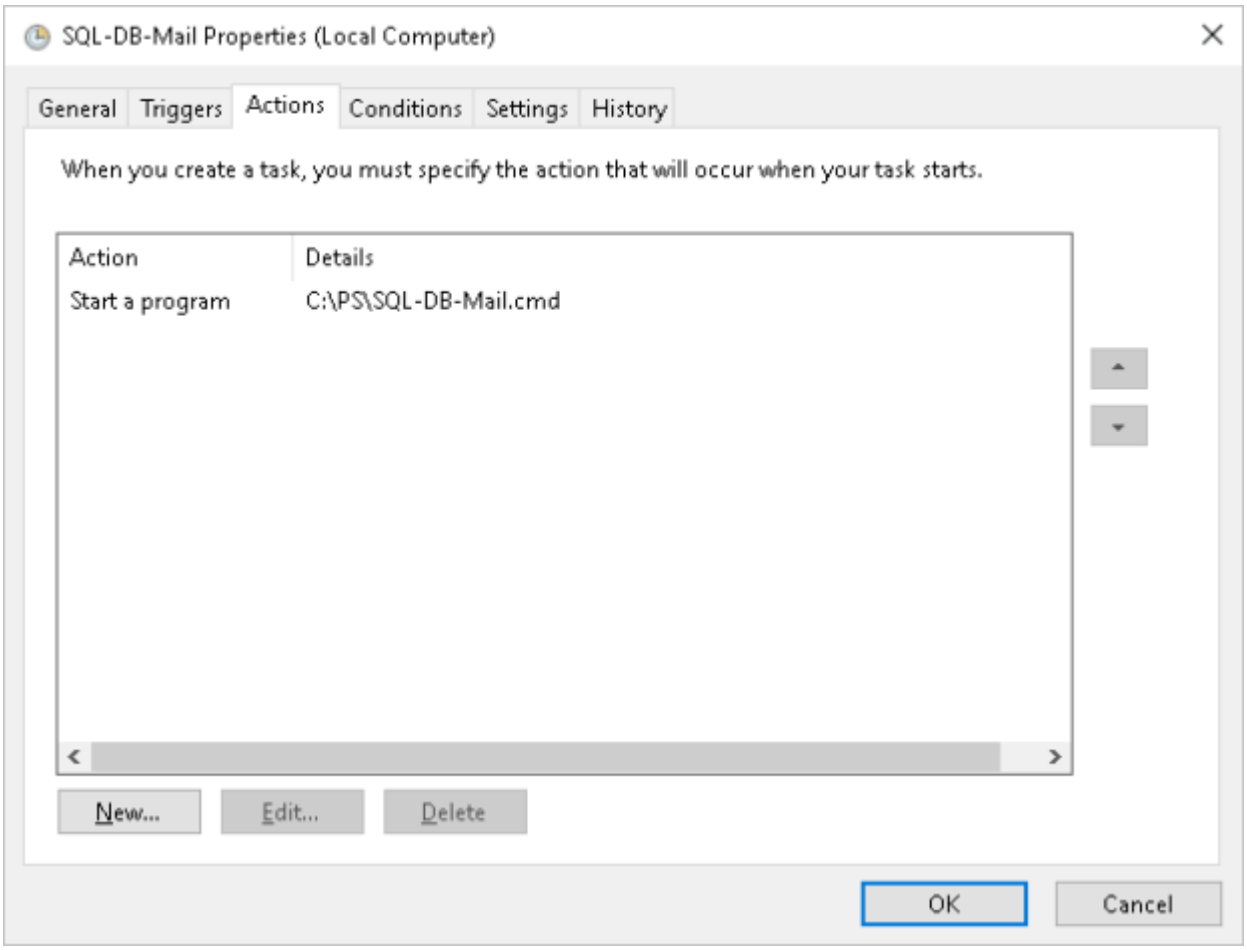

SQL-BCP-Mail Properties (Local Computer)
General
Triggers
Actions
Conditions
Settings
History
Specify the conditions that, along with the trigger, determine whether the task should run. The task will not run if any condition specified here is not true.
Idle
Start the task only if the computer is idle for:
10 minutes
Wait for idle for:
1 hour
Stop if the computer ceases to be idle
Restart if the idle state resumes
Power
Start the task only if the computer is on AC power
Stop if the computer switches to battery power
Wake the computer to run this task
Network
Start only if the following network connection is available:
Any connection
OK
Cancel

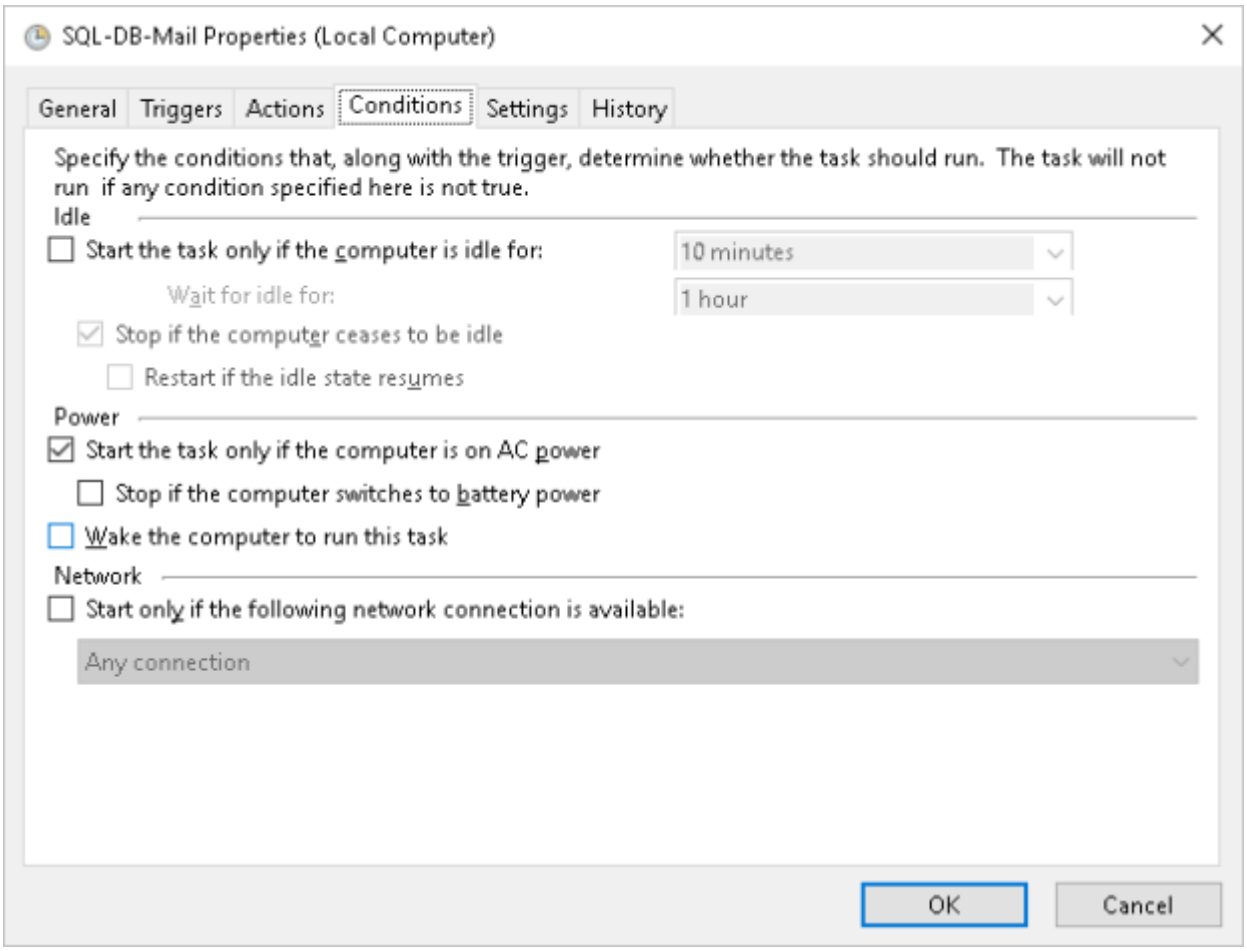

SQL-BCP-Mail Properties (Local Computer)
General
Triggers
Actions
Conditions
Settings
History
Specify additional settings that affect the behavior of the task.
Allow task to be run on demand
Run task as soon as possible after a scheduled start is missed
If the task fails, restart every:
10 minut
Attempt to restart up to:
3
times
Stop the task if it runs longer than:
1 hour
If the running task does not end when requested, force it to stop
If the task is not scheduled to run again, delete it after:
30 days
If the task is already running, then the following rule applies:
Run a new instance in parallel
OK
Cancel
Task Scheduler
File
Action
View
Help
Task Scheduler (Local)
Task Scheduler Library
Name
Status
Triggers
Next Run Time
Last Run Time
Last Run Result
SQL-BCP-Mail
Ready
At 8:00 PM every day
8/31/2017 8:00:00 PM
8/30/2017 8:00:00 PM
The operation completed succ
SQL-DB-Mail
Ready
At 9:00 AM every day
9/1/2017 9:00:00 AM
8/31/2017 9:00:01 AM
The operation completed succ
General
Triggers
Actions
Conditions
Settings
History
Number of events: 277
Level
Date and Time
Event ID
Task Category
Operational Code
Correlation Id
Information
8/30/2017 8:00:07 PM
102
Task completed
(2)
241078f5-0d...
Information
8/30/2017 8:00:07 PM
201
Action completed
(2)
241078f5-0d...
Information
8/30/2017 8:00:00 PM
200
Action started
(1)
241078f5-0d...
Information
8/30/2017 8:00:00 PM
100
Task Started
(1)
241078f5-0d...
Information
8/30/2017 8:00:00 PM
129
Created Task Process
Info
Information
8/30/2017 8:00:00 PM
107
Task triggered on sch...
Info
241078f5-0d...
Information
8/29/2017 8:00:09 PM
102
Task completed
(2)
595df0ed-6...
Information
8/29/2017 8:00:09 PM
201
Action completed
(2)
595df0ed-6...
Information
8/29/2017 8:00:01 PM
200
Action started
(1)
595df0ed-6...
Information
8/29/2017 8:00:01 PM
100
Task Started
(1)
595df0ed-6...
Information
8/29/2017 8:00:01 PM
129
Created Task Process
Info
Event 102, TaskScheduler
General
Details
Task Scheduler successfully finished "{241078f5-0ded-4366-9c72-e6fa20874058}" instance of the "\SQL-BCP-Mail" task for user "TESTENTERPRISE\ExchAdmin".
Log Name:
Microsoft-Windows-TaskScheduler/Operational
Source:
TaskScheduler
Logged:
8/30/2017 8:00:07 PM
Actions
Task Sche...
Creat...
Creat...
Impor...
Displa...
Disabl...
New F...
View
Refresh
Help
Selected It...
Run
End
Disable
Export...
Prope...
Delete
Help

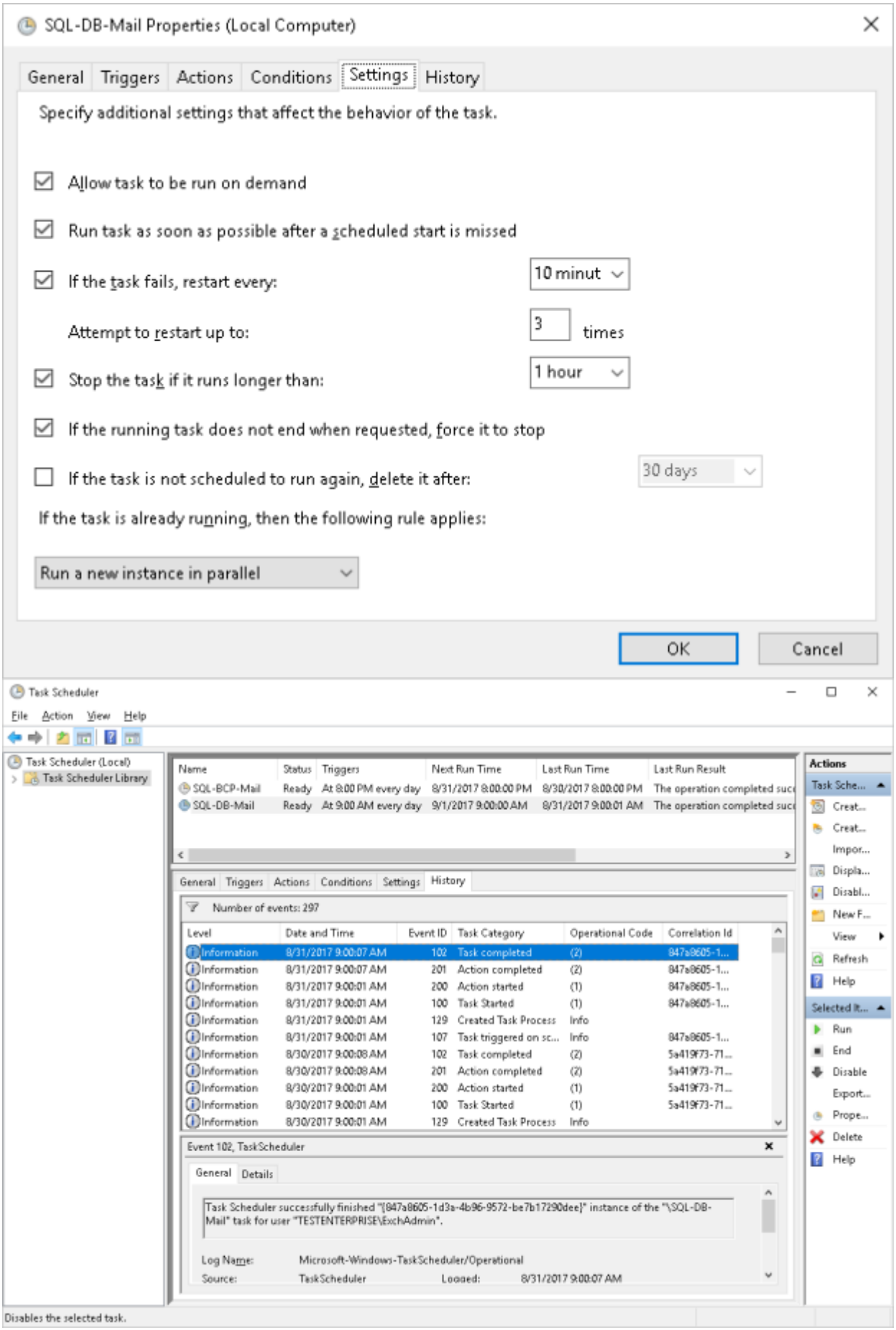
2) **SQL-DB-Mail**:


Task Scheduler
File Action View Help
Task Scheduler (Local)
Task Scheduler Library
Name Status Triggers Next Run Time Last Run Time Last Run Result
SQL-BCP-Mail Ready At 8:00 PM every day 8/31/2017 8:00:00 PM 8/30/2017 8:00:00 PM The operation completed succ
SQL-DB-Mail Ready At 9:00 AM every day 9/1/2017 9:00:00 AM 8/31/2017 9:00:01 AM The operation completed succ
General Triggers Actions Conditions Settings History
Name: SQL-DB-Mail
Location: \
Author: TESTENTERPRISE\ExchAdmin
Description: Creating and sending daily SQL Server report.
Security options
When running the task, use the following user account:
ExchAdmin
Run only when user is logged on
Run whether user is logged on or not
Do not store password. The task will only have access to local resources
Run with highest privileges
Hidden
Configure for: Windows Server 2016
Actions
Task Sche...
Creat...
Creat...
Impor...
Displa...
Disabl...
New F...
View
Refresh
Help
Selected It...
Run
End
Disable
Export...
Prope...
Delete
Help


SQL-DB-Mail Properties (Local Computer)
General Triggers Actions Conditions Settings History
When you create a task, you can specify the conditions that will trigger the task.
Trigger Details Status
Daily At 9:00 AM every day Enabled
New... Edit... Delete
OK Cancel


SQL-DB-Mail Properties (Local Computer)
General
Triggers
Actions
Conditions
Settings
History
When you create a task, you must specify the action that will occur when your task starts.
Action
Details
Start a program
C:\PS\SQL-DB-Mail.cmd
New...
Edit...
Delete
OK
Cancel

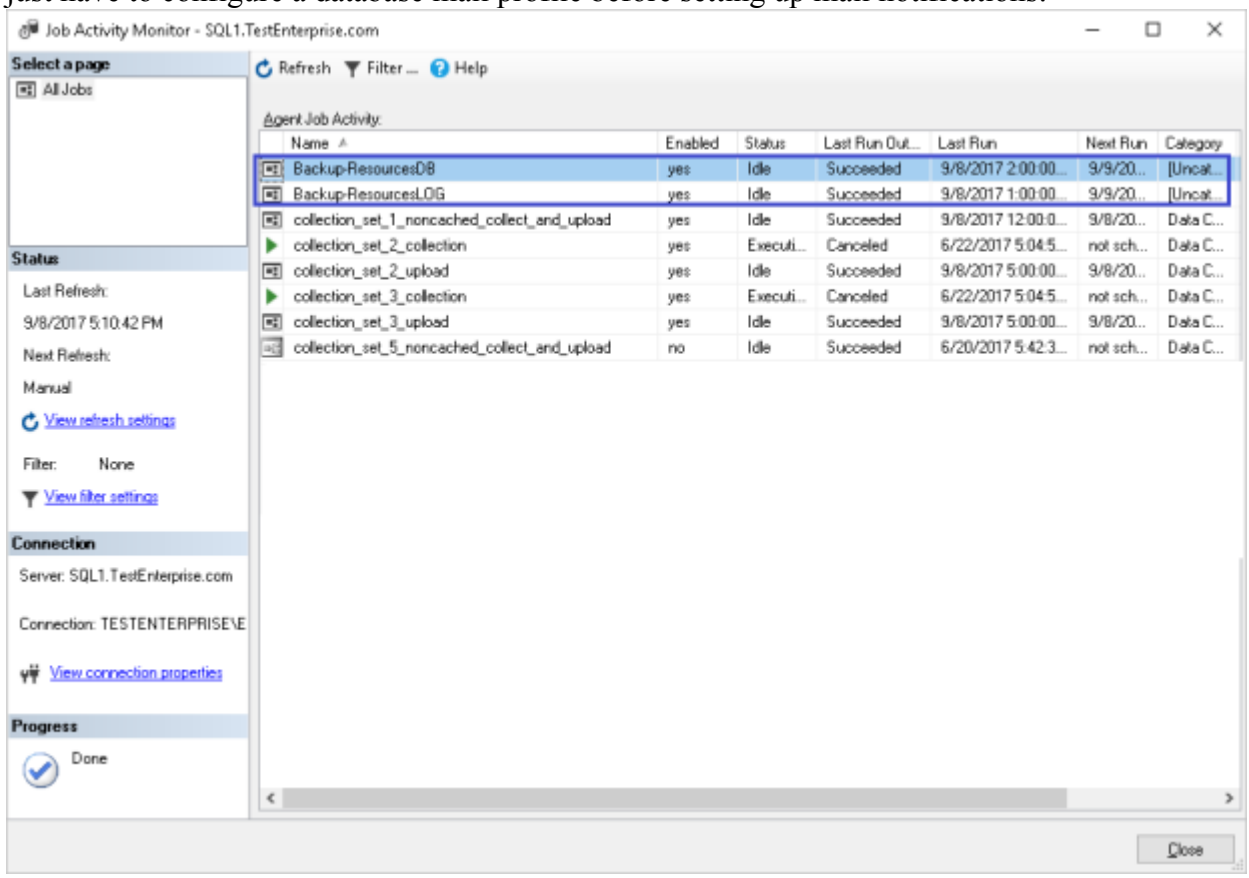

SQL-DB-Mail Properties (Local Computer)
General
Triggers
Actions
Conditions
Settings
History
Specify the conditions that, along with the trigger, determine whether the task should run. The task will not run if any condition specified here is not true.
Idle
Start the task only if the computer is idle for:
10 minutes
Wait for idle for:
1 hour
Stop if the computer ceases to be idle
Restart if the idle state resumes
Power
Start the task only if the computer is on AC power
Stop if the computer switches to battery power
Wake the computer to run this task
Network
Start only if the following network connection is available:
Any connection
OK
Cancel

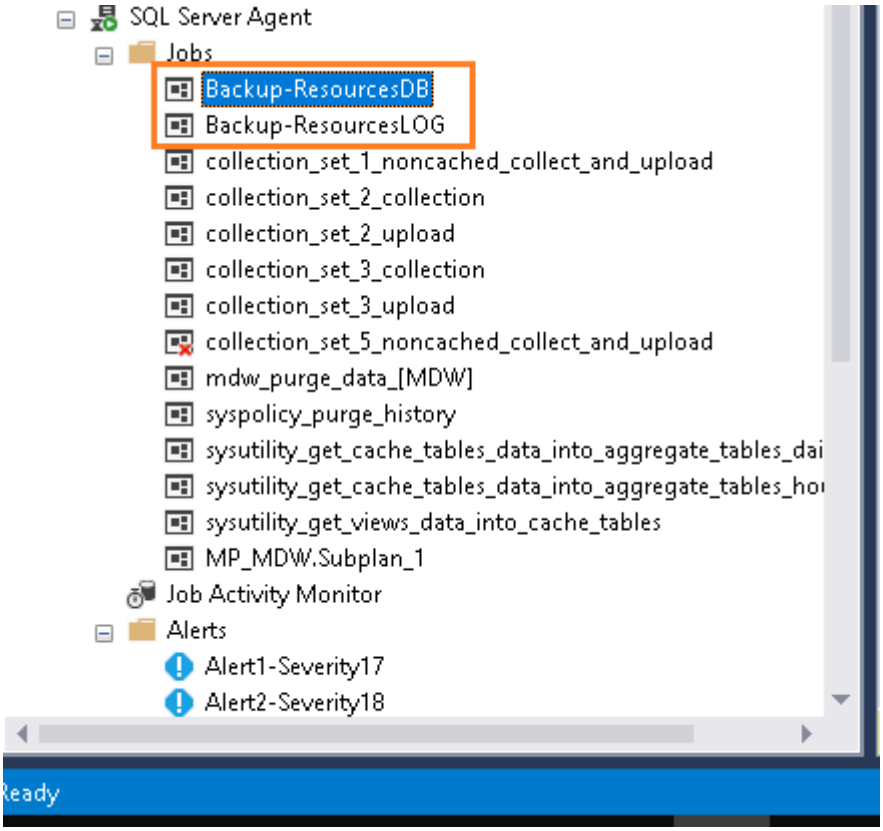


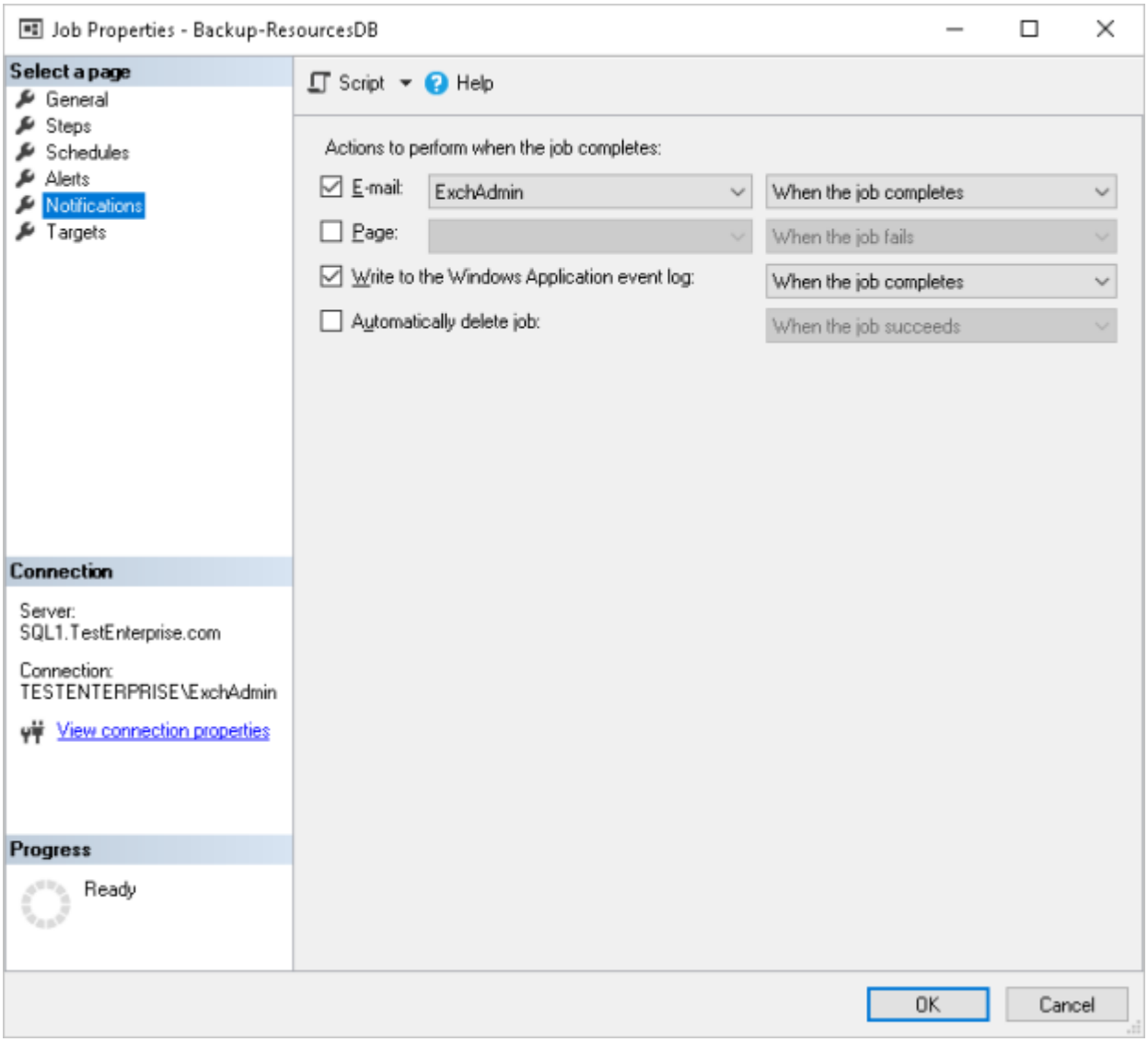
The information contained in the SQL reports will allow administrators to know the main current database parameters and take corrective actions if the necessity arises.

3) **Databases backups**

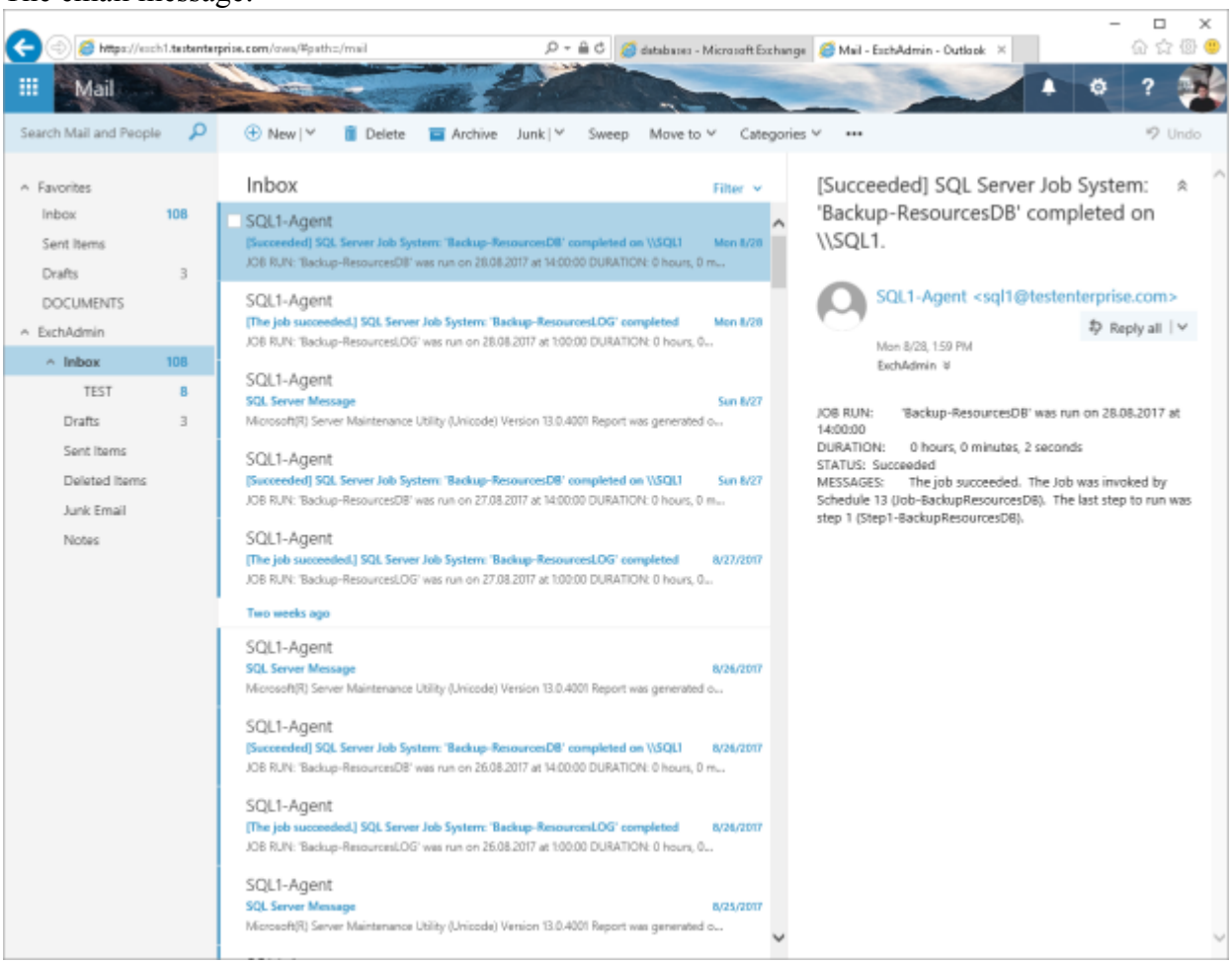
Each sql server backup job or maintenance plan can send email notifications upon completion – you just have to configure a database mail profile before setting up mail notifications:

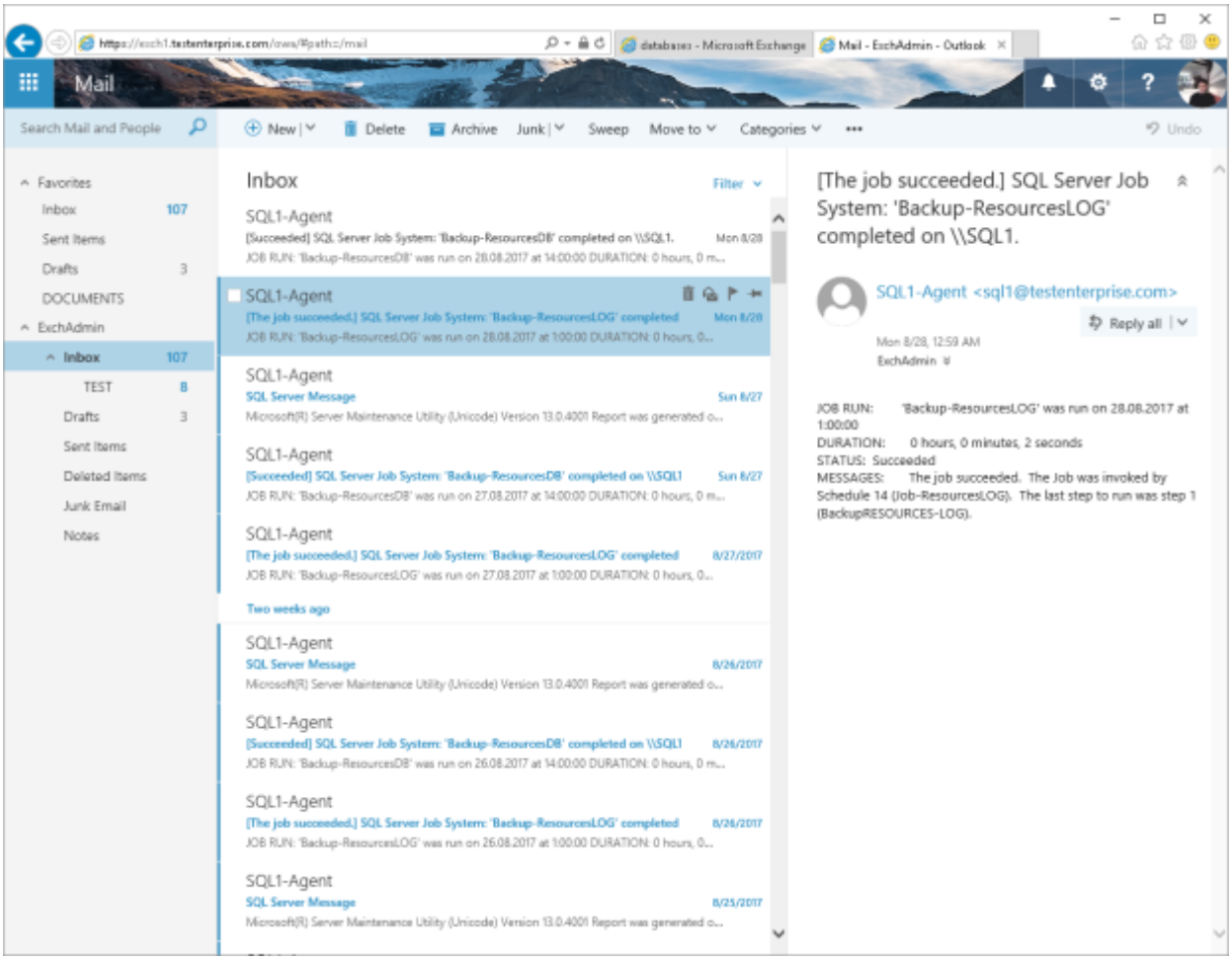





Job Properties - Backup-ResourcesDB
Select a page
General
Steps
Schedules
Alerts
Notifications
Targets
Script
Help
Actions to perform when the job completes:
E-mail:
ExchAdmin
When the job completes
Page:
When the job fails
Write to the Windows Application event log:
When the job completes
Automatically delete job:
When the job succeeds
Connection
Server:
SQL1.TestEnterprise.com
Connection:
TESTENTERPRISE\ExchAdmin
View connection properties
Progress
Ready
OK
Cancel

The email message:





4-1) Monitoring

For daily monitoring I'm using the following data collector sets:

SQL-MEMORY

SQLServer:Buffer Manager\Buffer cache hit ratio
SQLServer:Buffer Manager\Checkpoint pages/sec
SQLServer:Buffer Manager\Page life expectancy
SQLServer:Buffer Manager\Lazy writes/sec
SQLServer:Buffer Manager\Free list stalls/sec

SQLServer:Memory Manager\Memory Grants Pending
SQLServer:Memory Manager\Total Server Memory (KB)
SQLServer:Memory Manager\Target Server Memory (KB)
SQLServer:Memory Manager\Connection Memory (KB)

SQL1

SQLServer:Databases(*)\Data File(s) Size (KB)
SQLServer:Databases(*)\Log File(s) Size (KB)
SQLServer:Databases(*)\Log File(s) Used Size (KB)
SQLServer:Databases(*)\Log Truncations
SQLServer:Databases(*)\Log Growths
SQLServer:Databases(*)\Transactions/sec

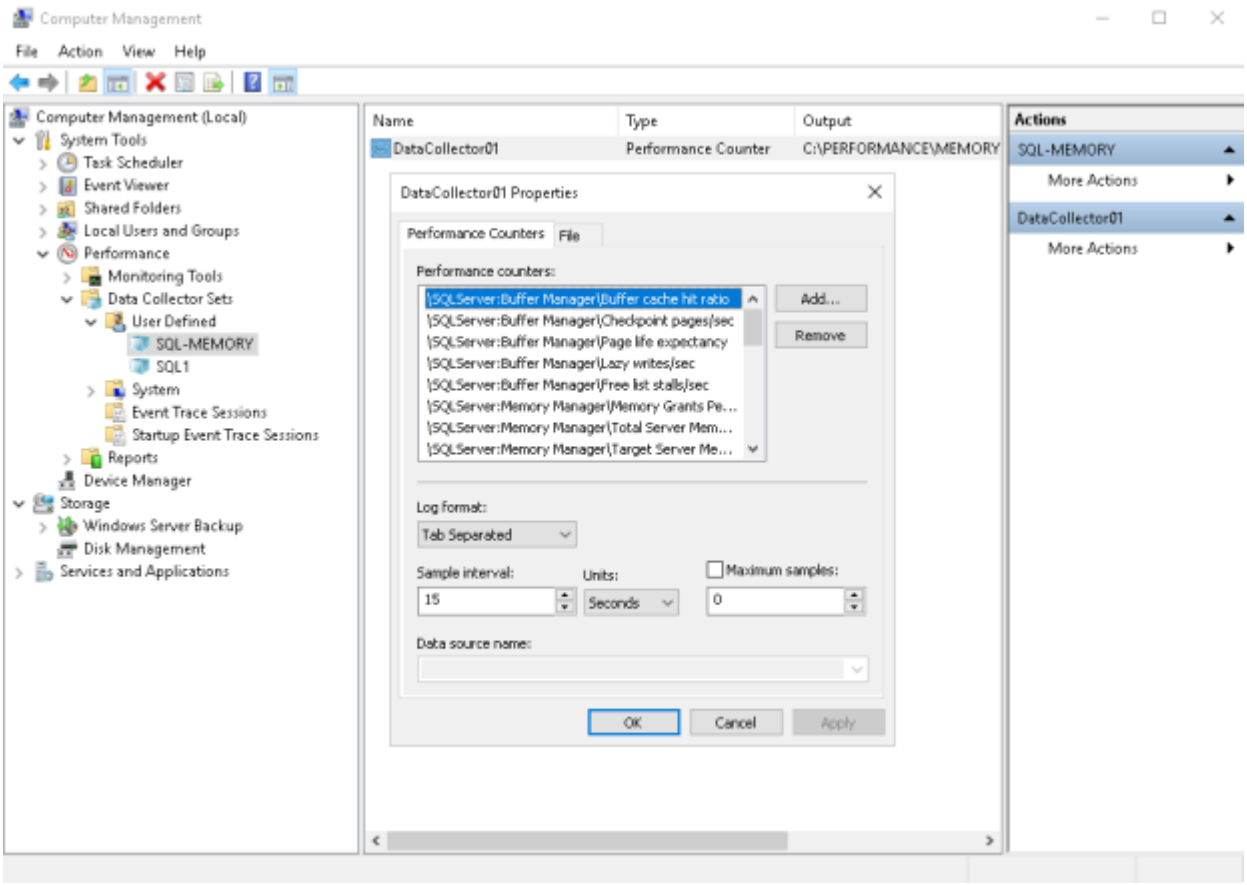
SQLServer:Latches\Latch Waits/sec
SQLServer:Latches\Total Latch Wait Time (ms)

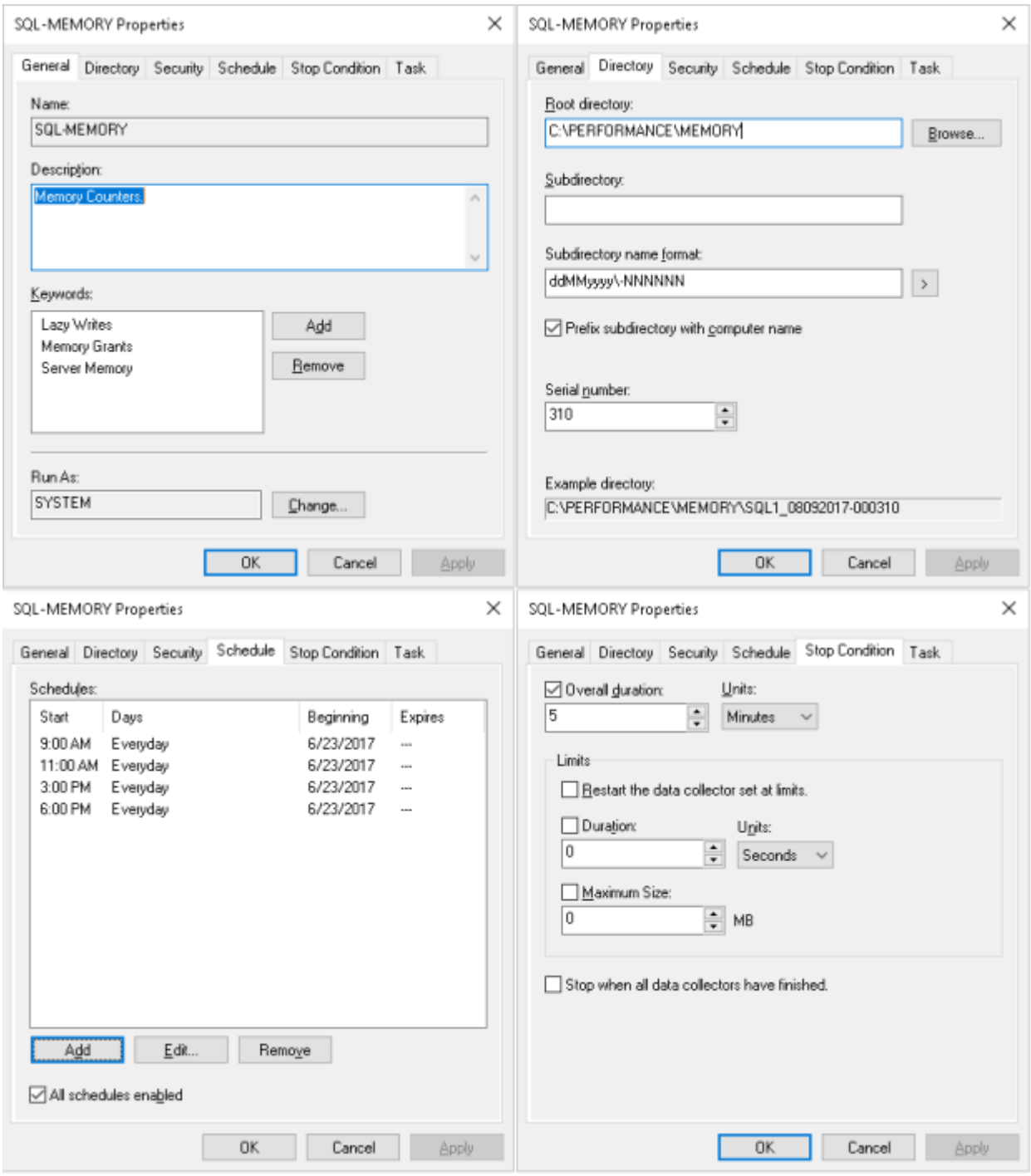
SQLServer:Locks(*)\Number of Deadlocks/sec
SQLServer:Locks(_Total)\Number of Deadlocks/sec
SQLServer:Locks(*)\Lock Timeouts/sec
SQLServer:Locks(_Total)\Lock Timeouts/sec

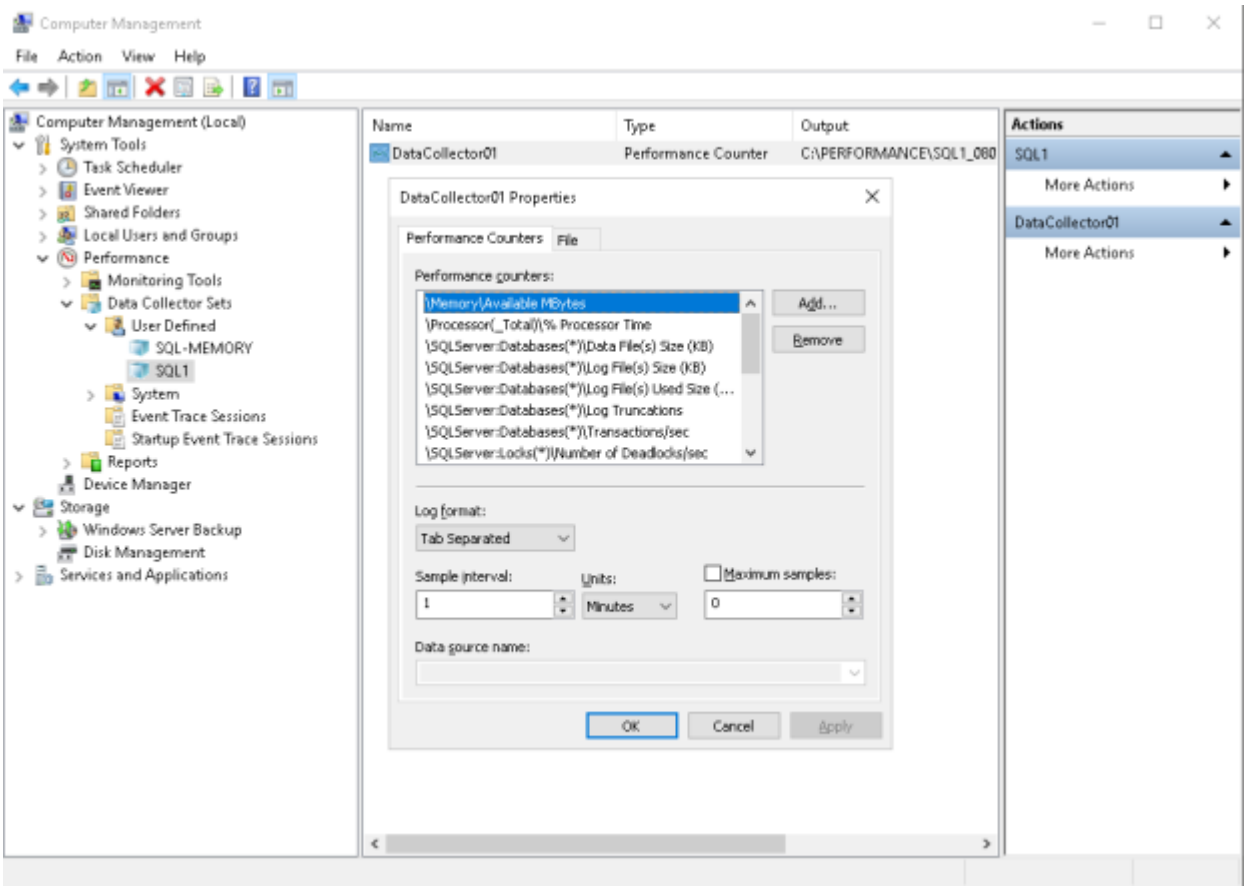
SQL-Server:SQL Errors\Errors\sec

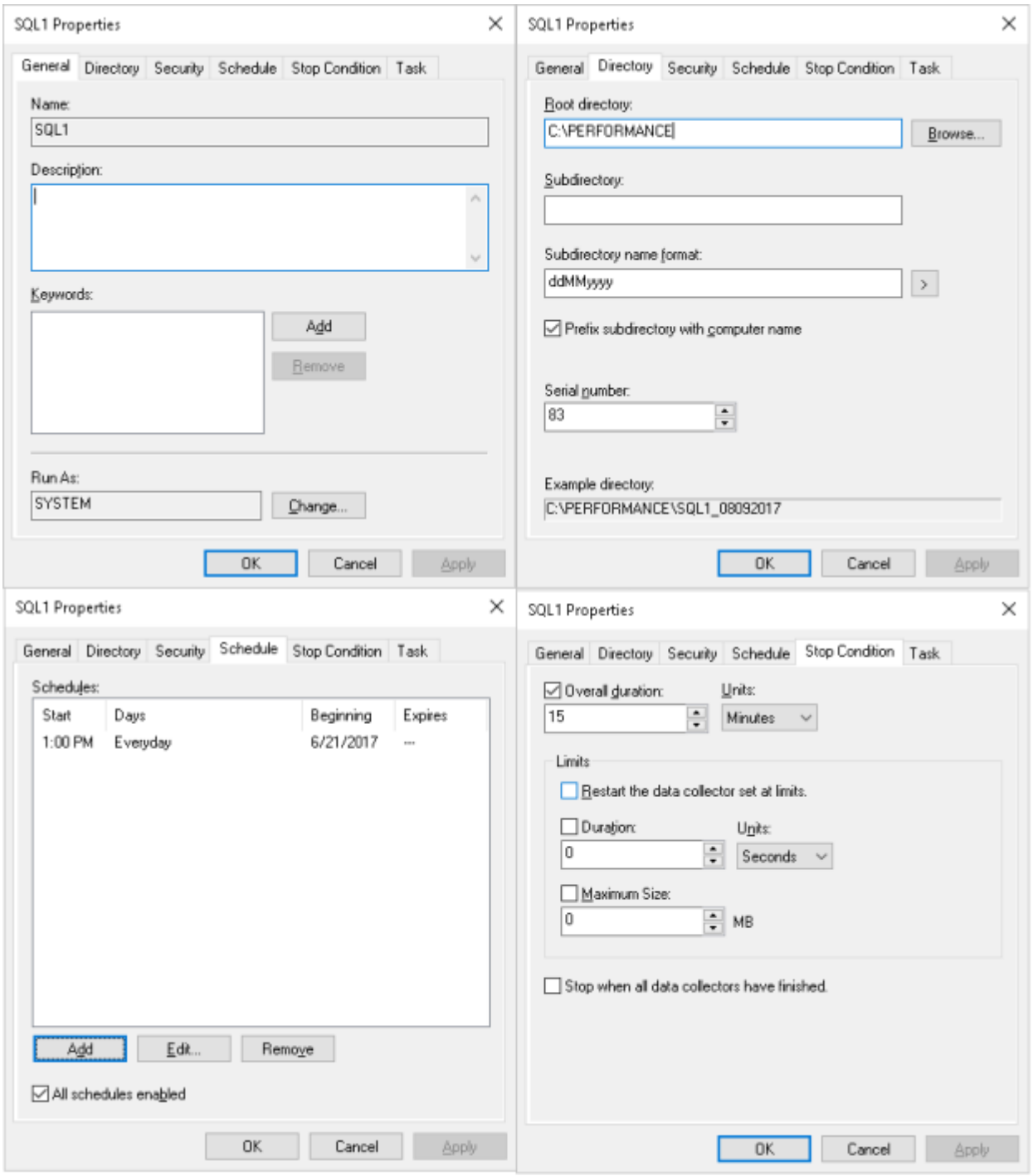
SQLServer:Transactions\FreeSpace in tempdb database
SQLServer:Transactions\LongestTranscationRunningTime

SQLServer:Wait Statistics(*)\Lock waits

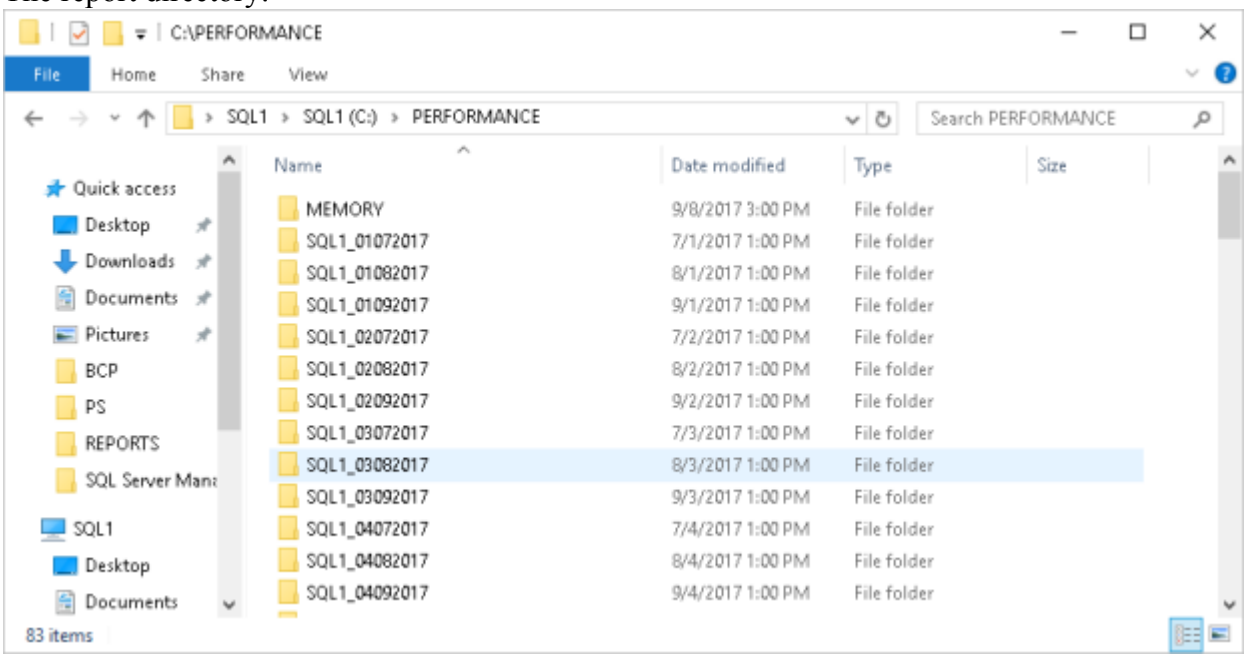

Computer Management
File Action View Help
Computer Management (Local)
System Tools
Task Scheduler
Event Viewer
Shared Folders
Local Users and Groups
Performance
Monitoring Tools
Data Collector Sets
User Defined
SQL-MEMORY
SQL1
System
Event Trace Sessions
Startup Event Trace Sessions
Reports
Device Manager
Storage
Windows Server Backup
Disk Management
Services and Applications
Name
Type
Output
DataCollector01
Performance Counter
C:\PERFORMANCE\MEMORY
DataCollector01 Properties
Performance Counters
File
Performance counters:
\SQLServer:Buffer Manager\Buffer cache hit ratio
\SQLServer:Buffer Manager\Checkpoint pages/sec
\SQLServer:Buffer Manager\Page life expectancy
\SQLServer:Buffer Manager\Lazy writes/sec
\SQLServer:Buffer Manager\Free list stalls/sec
\SQLServer:Memory Manager\Memory Grants Pe...
\SQLServer:Memory Manager\Total Server Mem...
\SQLServer:Memory Manager\Target Server Me...
Add...
Remove
Log format:
Tab Separated
Sample interval:
15
Units:
Seconds
Maximum samples:
0
Data source name:
OK
Cancel
Apply
Actions
SQL-MEMORY
More Actions
DataCollector01
More Actions


SQL-MEMORY Properties
General
Directory
Security
Schedule
Stop Condition
Task
Name:
SQL-MEMORY
Description:
Memory Counters
Keywords:
Lazy Writes
Memory Grants
Server Memory
Add
Remove
Run As:
SYSTEM
Change...
OK
Cancel
Apply
SQL-MEMORY Properties
General
Directory
Security
Schedule
Stop Condition
Task
Root directory:
C:\PERFORMANCE\MEMORY
Browse...
Subdirectory:
Subdirectory name format:
ddMMyyyy\-NNNNNN
Prefix subdirectory with computer name
Serial number:
310
Example directory:
C:\PERFORMANCE\MEMORY\SQL1_08092017-000310
OK
Cancel
Apply
SQL-MEMORY Properties
General
Directory
Security
Schedule
Stop Condition
Task
Schedules:
Start
Days
Beginning
Expires
9:00 AM
Everyday
6/23/2017
11:00 AM
Everyday
6/23/2017
3:00 PM
Everyday
6/23/2017
6:00 PM
Everyday
6/23/2017
Add
Edit...
Remove
All schedules enabled
OK
Cancel
Apply
SQL-MEMORY Properties
General
Directory
Security
Schedule
Stop Condition
Task
Overall duration:
5
Units:
Minutes
Limits
Restart the data collector set at limits.
Duration:
0
Units:
Seconds
Maximum Size:
0
MB
Stop when all data collectors have finished.
OK
Cancel
Apply


Computer Management
File Action View Help
Computer Management (Local)
System Tools
Task Scheduler
Event Viewer
Shared Folders
Local Users and Groups
Performance
Monitoring Tools
Data Collector Sets
User Defined
SQL-MEMORY
SQL1
System
Event Trace Sessions
Startup Event Trace Sessions
Reports
Device Manager
Storage
Windows Server Backup
Disk Management
Services and Applications
Name
Type
Output
DataCollector01
Performance Counter
C:\PERFORMANCE\SQL1_060
DataCollector01 Properties
Performance Counters
File
Performance counters:
\Memory\Available MBytes
\Processor(_Total)\% Processor Time
\SQLServer:Databases(*)\Data File(s) Size (KB)
\SQLServer:Databases(*)\Log File(s) Size (KB)
\SQLServer:Databases(*)\Log File(s) Used Size (...
\SQLServer:Databases(*)\Log Truncations
\SQLServer:Databases(*)\Transactions/sec
\SQLServer:Locks(*)\Number of Deadlocks/sec
Add...
Remove
Log format:
Tab Separated
Sample interval:
1
Units:
Minutes
Maximum samples:
0
Data source name:
OK
Cancel
Apply
Actions
SQL1
More Actions
DataCollector01
More Actions

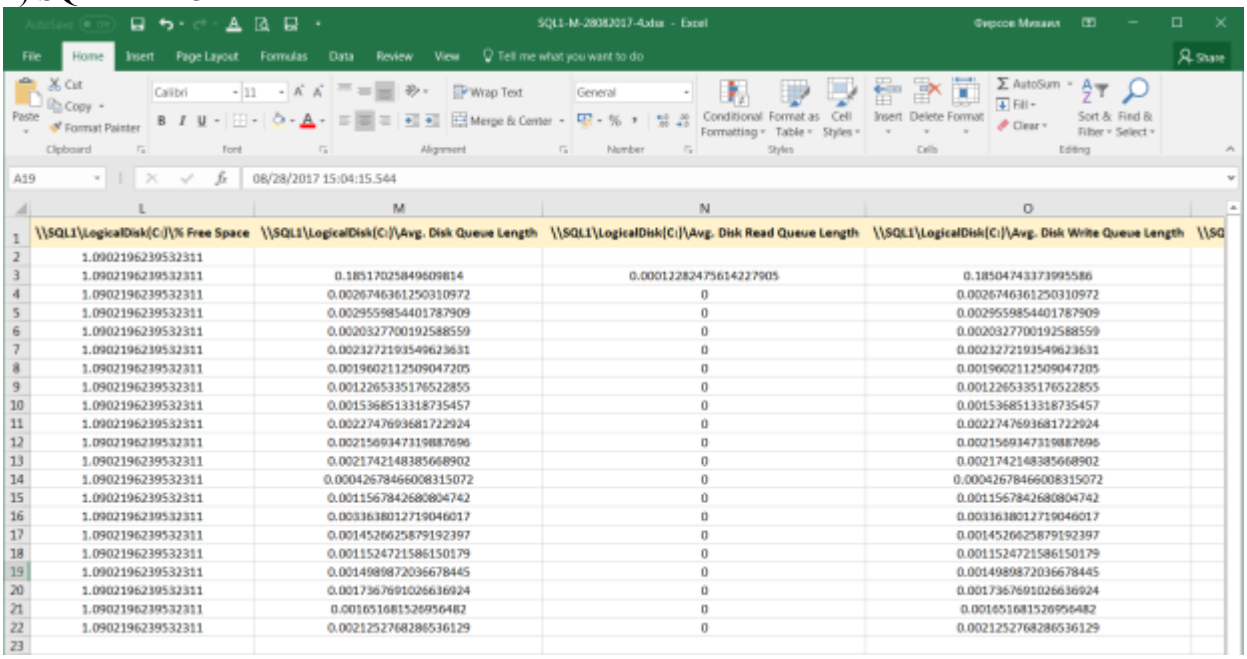

SQL1 Properties
General
Directory
Security
Schedule
Stop Condition
Task
Name:
SQL1
Description:
Keywords:
Add
Remove
Run As:
SYSTEM
Change...
OK
Cancel
Apply
SQL1 Properties
General
Directory
Security
Schedule
Stop Condition
Task
Root directory:
C:\PERFORMANCE
Browse...
Subdirectory:
Subdirectory name format:
ddMMyyyy
Prefix subdirectory with computer name
Serial number:
83
Example directory:
C:\PERFORMANCE\SQL1_08092017
OK
Cancel
Apply
SQL1 Properties
General
Directory
Security
Schedule
Stop Condition
Task
Schedules:
Start
Days
Beginning
Expires
1:00 PM
Everyday
6/21/2017
...
Add
Edit...
Remove
All schedules enabled
OK
Cancel
Apply
SQL1 Properties
General
Directory
Security
Schedule
Stop Condition
Task
Overall duration:
Units:
15
Minutes
Limits
Restart the data collector set at limits.
Duration:
Units:
0
Seconds
Maximum Size:
0
MB
Stop when all data collectors have finished.
OK
Cancel
Apply

The report directory:



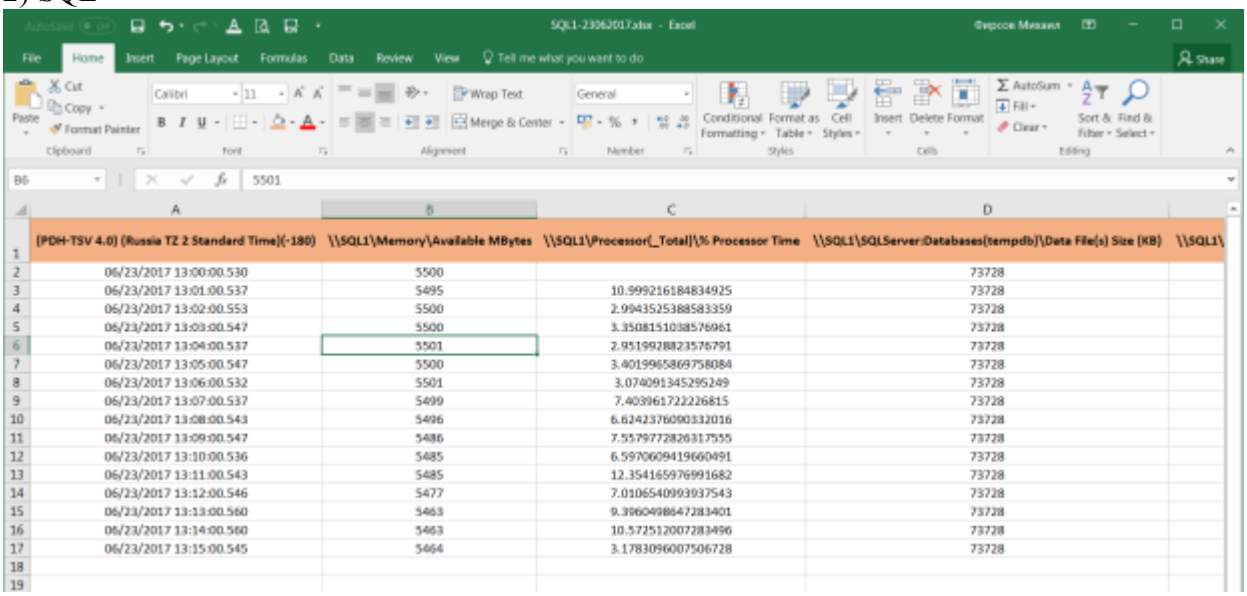
The reports:

1) SQL-MEMORY



	L	M	N	O
1	\\SQL1\LogicalDisk(C:)\% Free Space	\\SQL1\LogicalDisk(C:)\Avg. Disk Queue Length	\\SQL1\LogicalDisk(C:)\Avg. Disk Read Queue Length	\\SQL1\LogicalDisk(C:)\Avg. Disk Write Queue Length
2	1.0902196239532311			
3	1.0902196239532311	0.18517025849609814	0.0001228247561422790	0.18504743373995586
4	1.0902196239532311	0.002674636125031097	0	0.002674636125031097
5	1.0902196239532311	0.002955985440178790	0	0.002955985440178790
6	1.0902196239532311	0.002032770019258855	0	0.002032770019258855
7	1.0902196239532311	0.002327219354962363	0	0.002327219354962363
8	1.0902196239532311	0.001960211250904720	0	0.001960211250904720
9	1.0902196239532311	0.001226533517652285	0	0.001226533517652285
10	1.0902196239532311	0.001536851331873545	0	0.001536851331873545
11	1.0902196239532311	0.002274769368172292	0	0.002274769368172292
12	1.0902196239532311	0.002156934731988769	0	0.002156934731988769
13	1.0902196239532311	0.002174214838566890	0	0.002174214838566890
14	1.0902196239532311	0.000426784660083150	0	0.000426784660083150
15	1.0902196239532311	0.001156784268080474	0	0.001156784268080474
16	1.0902196239532311	0.003363801271904601	0	0.003363801271904601
17	1.0902196239532311	0.001452662587919239	0	0.001452662587919239
18	1.0902196239532311	0.001152472158615017	0	0.001152472158615017
19	1.0902196239532311	0.001498987203667844	0	0.001498987203667844
20	1.0902196239532311	0.001736769102663692	0	0.001736769102663692
21	1.0902196239532311	0.001651681526956482	0	0.001651681526956482
22	1.0902196239532311	0.002125276828653612	0	0.002125276828653612
23				

2) SQL



(PDH-TSV 4.0) (Russia TZ 2 Standard Time)(-180)	\\SQL1\Memory\Available MBytes	\\SQL1\Processor(_Total)\% Processor Time	\\SQL1\SQLServer:Databases(tempdb)\Data File(s) Size (KB)	\\SQL1\
06/23/2017 13:00:00.530	5500		73728	
06/23/2017 13:01:00.537	5495	10.999216184834925	73728	
06/23/2017 13:02:00.553	5500	2.9943525388583359	73728	
06/23/2017 13:03:00.547	5500	3.3508151038576961	73728	
06/23/2017 13:04:00.537	5501	2.9519928823576791	73728	
06/23/2017 13:05:00.547	5500	3.4019965869758084	73728	
06/23/2017 13:06:00.532	5501	3.074091345295249	73728	
06/23/2017 13:07:00.537	5499	7.403961722226815	73728	
06/23/2017 13:08:00.543	5496	6.6242376090332016	73728	
06/23/2017 13:09:00.547	5486	7.5579772826317555	73728	
06/23/2017 13:10:00.536	5485	6.5970609419660491	73728	
06/23/2017 13:11:00.543	5485	12.354165976991682	73728	
06/23/2017 13:12:00.546	5477	7.0106540993937543	73728	
06/23/2017 13:13:00.560	5463	9.3960498647283401	73728	
06/23/2017 13:14:00.560	5463	10.572512007283496	73728	
06/23/2017 13:15:00.545	5464	3.1783096007506728	73728	

Update: The following counters can be used for creating a DCS if you have an Always-ON group in your organization
(for the node named SQL2 in this example):

SQLServer:Availability Replica(AG1:SQL2)\Bytes Sent to Replica/sec

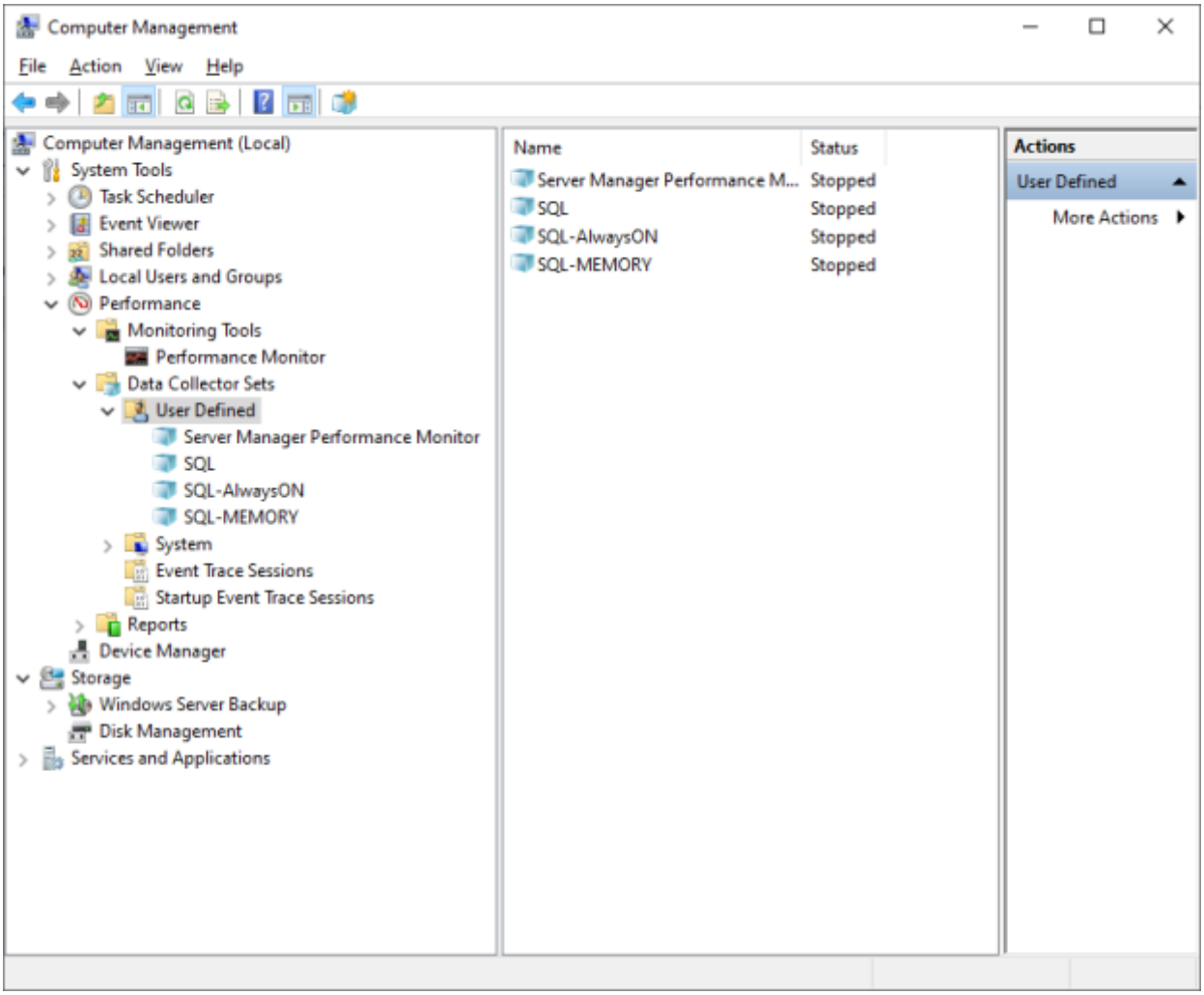
SQLServer:Availability Replica(AG1:SQL2)\Bytes Sent to Transport/sec

SQLServer:Database Replica(Resources)\Log Bytes Received/sec

SQLServer:Database Replica(Resources)\Redone Bytes/sec

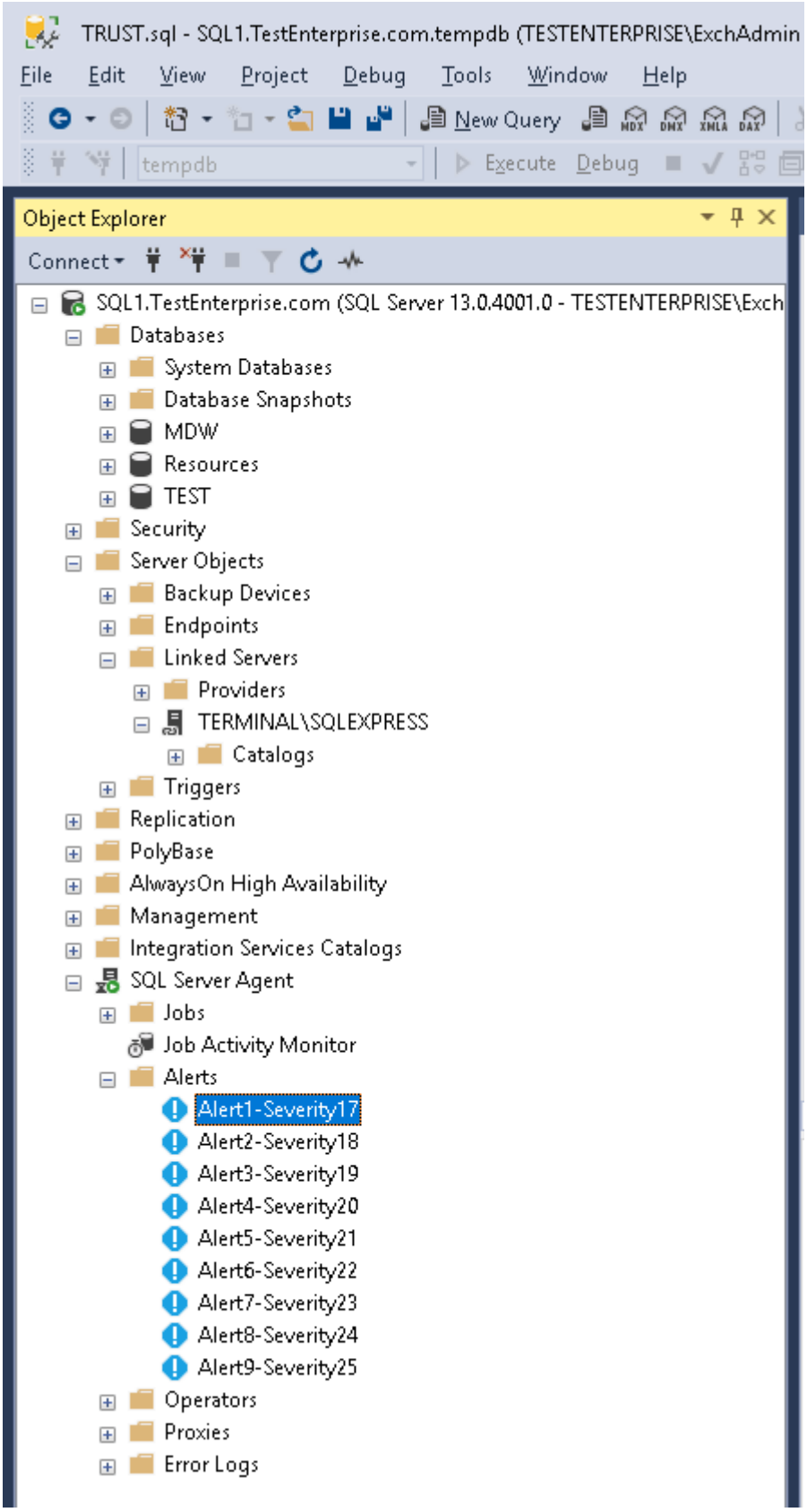
SQLServer:Databases(Resources)\Log Bytes Flushed/sec

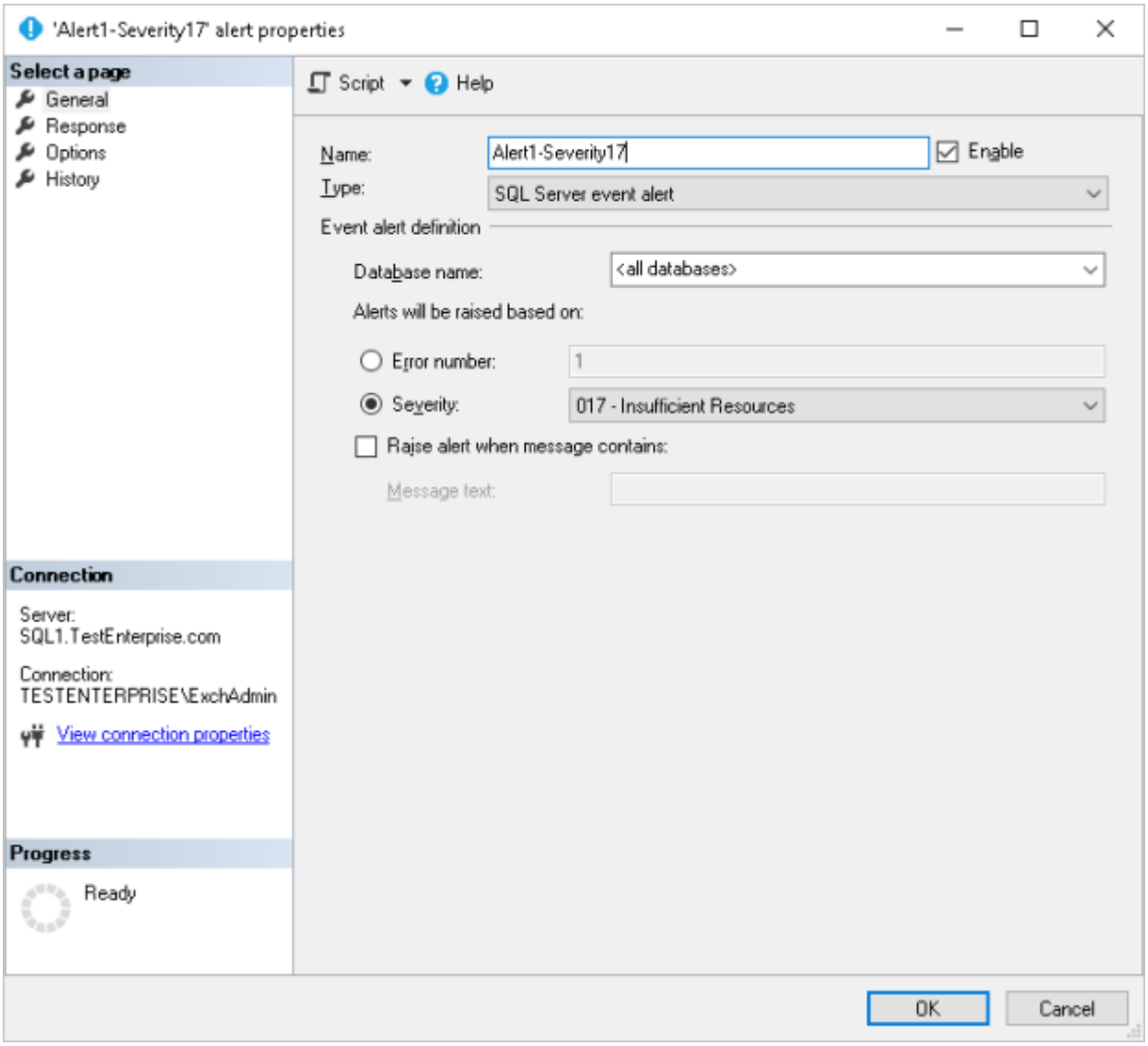
More information on monitoring Always-ON replication here.

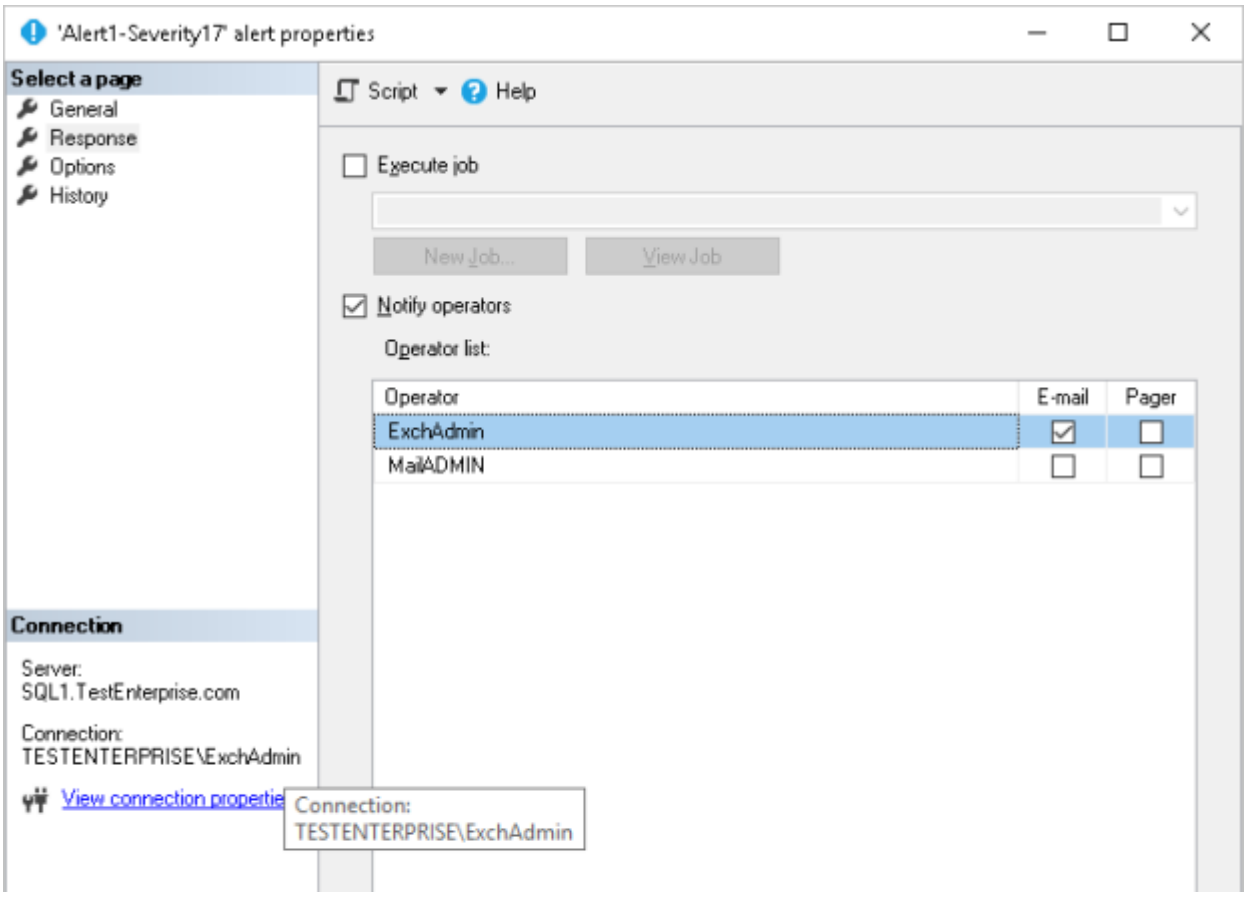


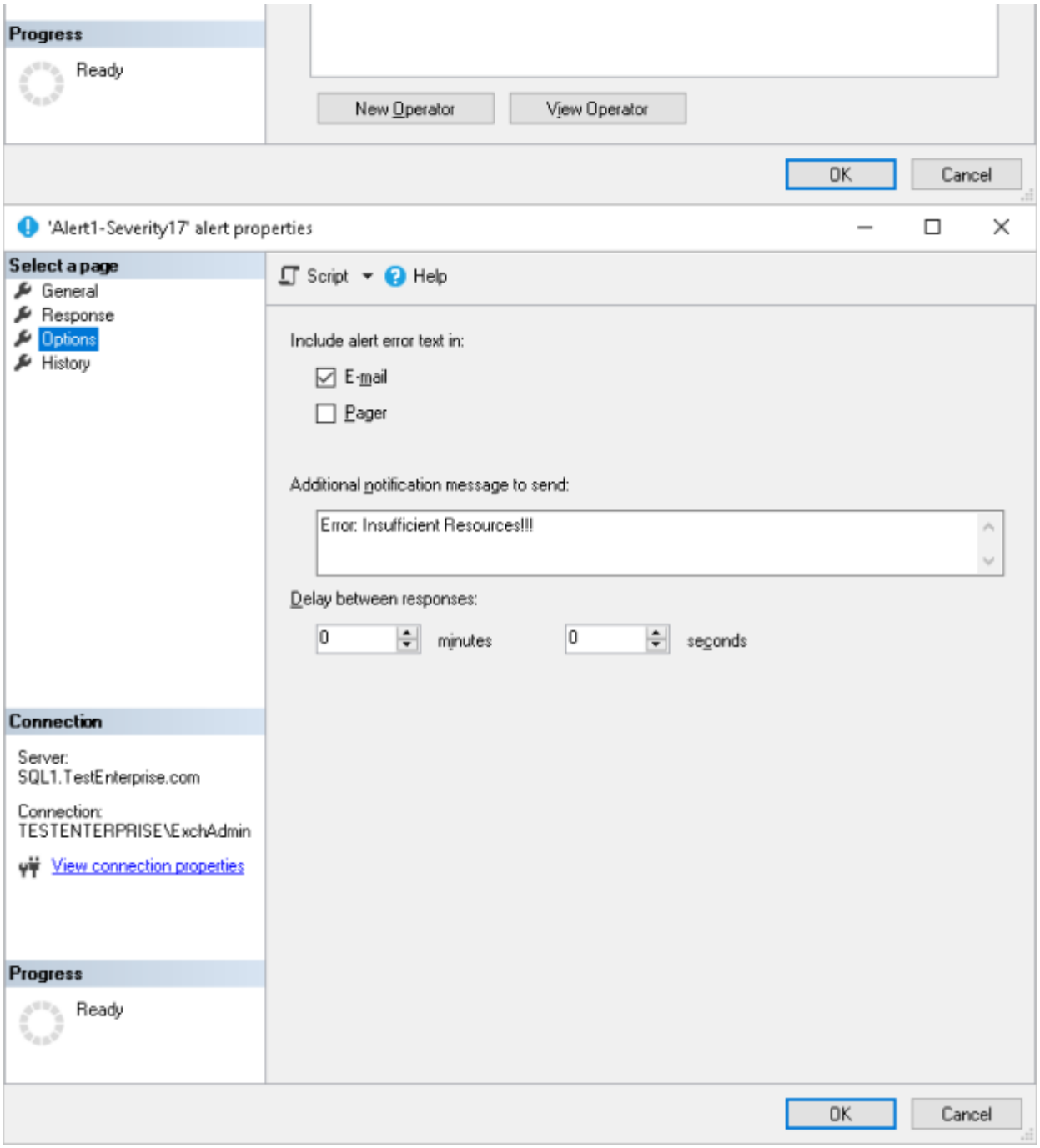
4-2) SQL Alerts

An administrator may create any user defined data collector set which will be monitoring any SQL-related counters in the same way as we use data collector sets for monitoring operating system related counters, but I think SQL server administrators should first configure monitoring of the SQL server installation by means of the SQL server itself: in the *SQL Server Agent\Alerts* we can create at least 9 alerts that would inform an administrator of an error with the severity level ranging from 17 to 25 – these alerts coupled with the database mail will send an administrator an alert should any error with the severity level 17-25 ever arise:

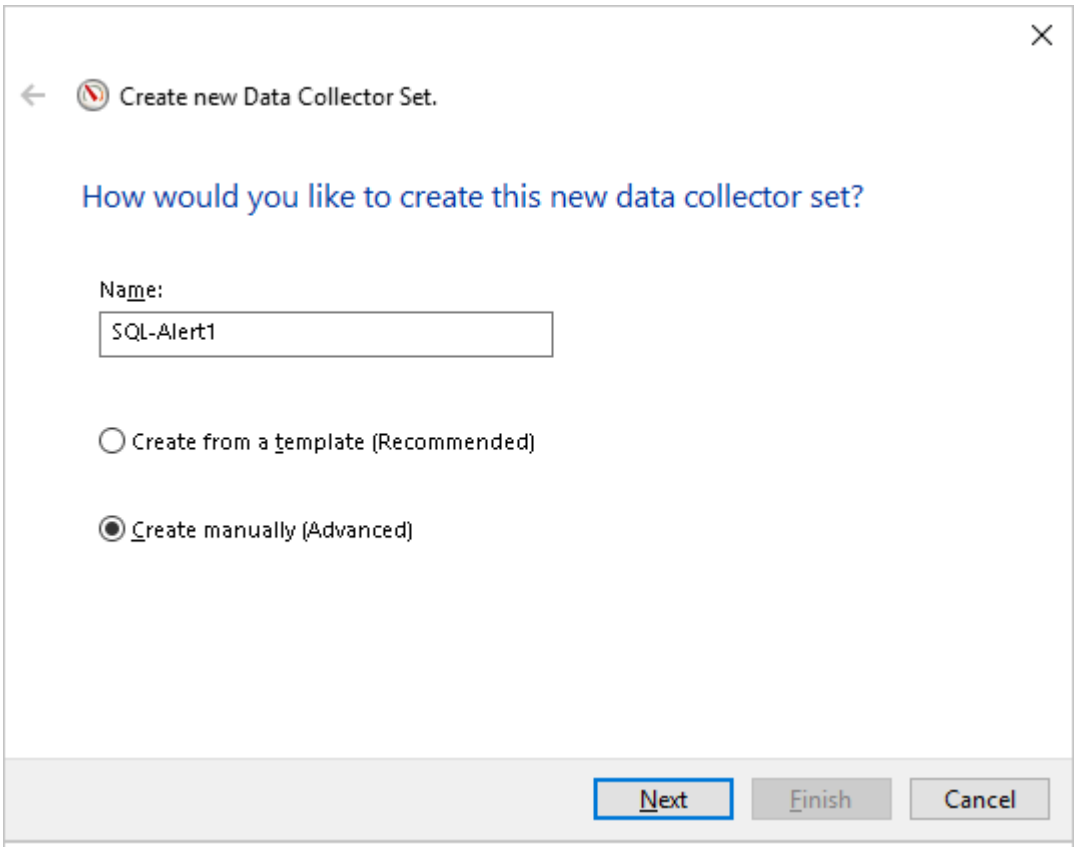

TRUST.sql - SQL1.TestEnterprise.com.tempdb (TESTENTERPRISE\ExchAdmin
File Edit View Project Debug Tools Window Help
New Query
tempdb
Execute Debug
Object Explorer
Connect
SQL1.TestEnterprise.com (SQL Server 13.0.4001.0 - TESTENTERPRISE\Exch
Databases
System Databases
Database Snapshots
MDW
Resources
TEST
Security
Server Objects
Backup Devices
Endpoints
Linked Servers
Providers
TERMINAL\SQLEXPRESS
Catalogs
Triggers
Replication
PolyBase
AlwaysOn High Availability
Management
Integration Services Catalogs
SQL Server Agent
Jobs
Job Activity Monitor
Alerts
Alert1-Severity17
Alert2-Severity18
Alert3-Severity19
Alert4-Severity20
Alert5-Severity21
Alert6-Severity22
Alert7-Severity23
Alert8-Severity24
Alert9-Severity25
Operators
Proxies
Error Logs

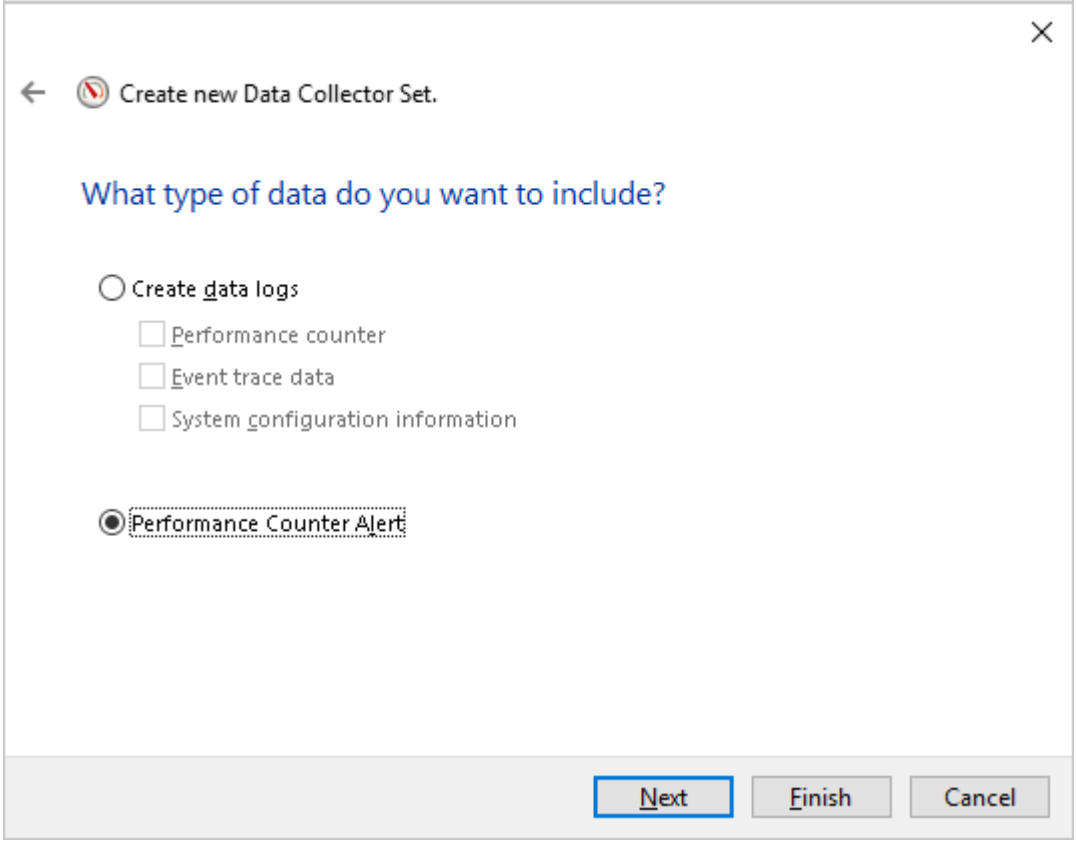

'Alert1-Severity17' alert properties
Select a page
General
Response
Options
History
Script
Help
Name:
Alert1-Severity17
Enable
Type:
SQL Server event alert
Event alert definition
Database name:
<all databases>
Alerts will be raised based on:
Error number:
1
Severity:
017 - Insufficient Resources
Raise alert when message contains:
Message text:
Connection
Server:
SQL1.TestEnterprise.com
Connection:
TESTENTERPRISE\ExchAdmin
View connection properties
Progress
Ready
OK
Cancel

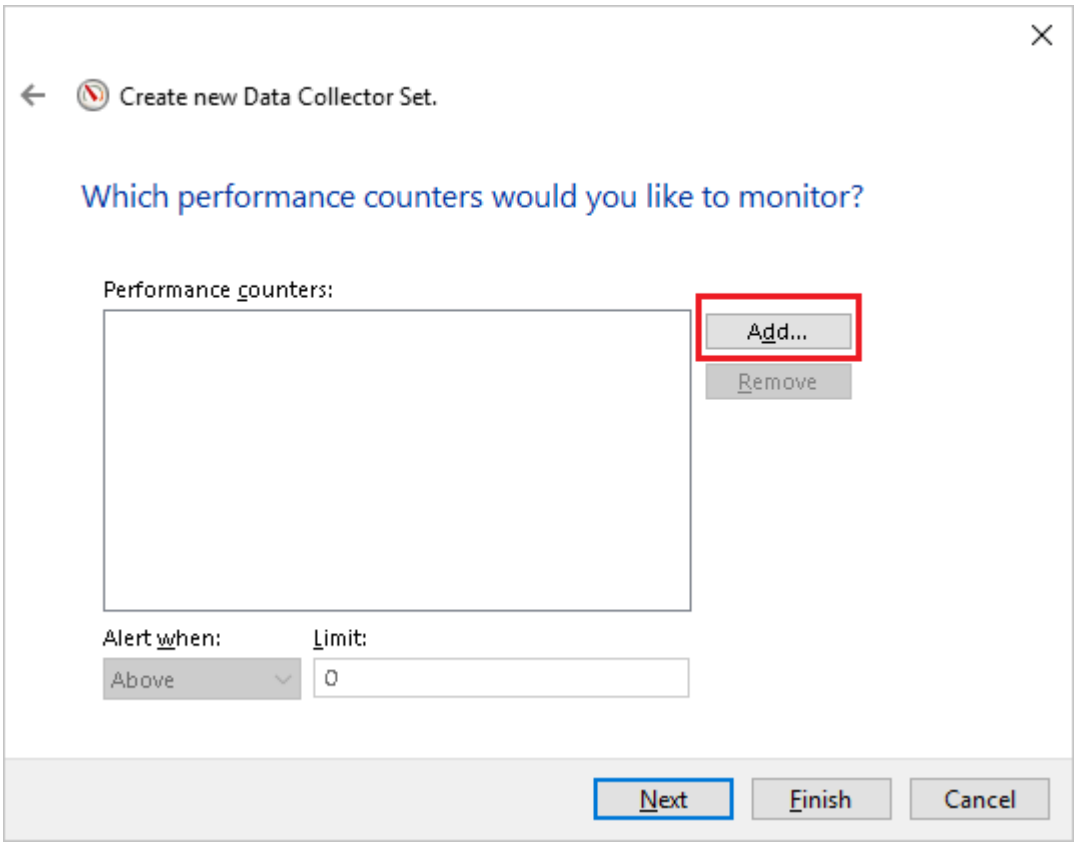

'Alert1-Severity17' alert properties
Select a page
General
Response
Options
History
Script
Help
Execute job
New Job...
View Job
Notify operators
Operator list:
Operator
E-mail
Pager
ExchAdmin
MailADMIN
Connection
Server:
SQL1.TestEnterprise.com
Connection:
TESTENTERPRISE\ExchAdmin
View connection properties
Connection:
TESTENTERPRISE\ExchAdmin

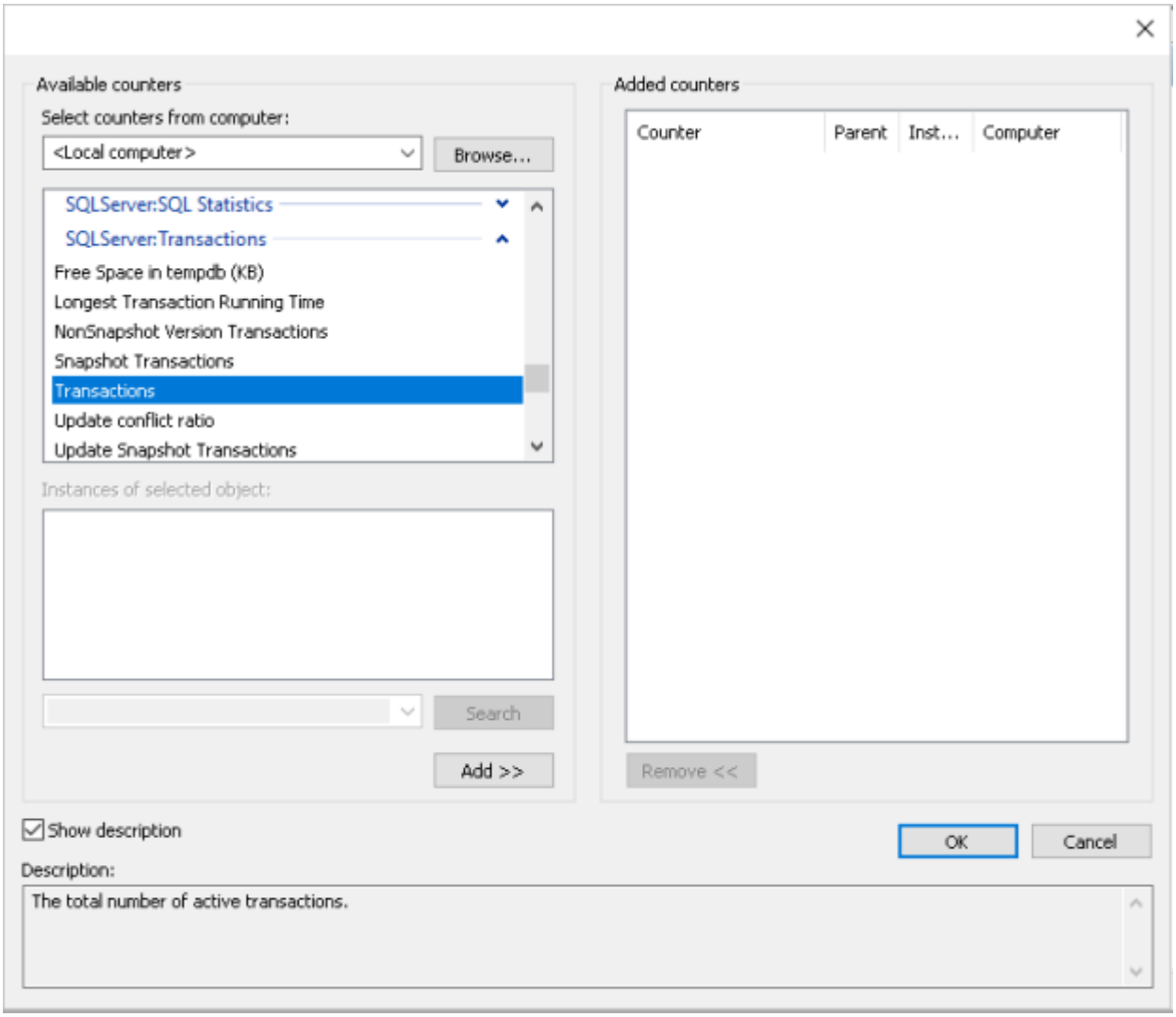


Of course, a number of alerts can be created based on the numerous performance counters that get added during SQL server installation, for example:


Create new Data Collector Set.
How would you like to create this new data collector set?
Name:
SQL-Alert1
Create from a template (Recommended)
Create manually (Advanced)
Next
Finish
Cancel


Create new Data Collector Set.
What type of data do you want to include?
Create data logs
Performance counter
Event trace data
System configuration information
Performance Counter Alert
Next
Finish
Cancel





Such alerts maybe configured to send email notifications exactly as we've seen in part1, for example:

```
$login = “sysadmin@testenterprise.com”
$password = “123456” | Convertto-SecureString -AsPlainText -Force
$creds = New-Object System.Management.Automation.Pscredential -Argumentlist $login,$password
Send-MailMessage -From “sysadmin@testenterprise.com” -To “michael_firsov@testenterprise.cpom” -Subject “SQL Server Transactions Alert
!” -Body “SQL Server Transactions Alert! – Transactions > X” -SmtpServer mail.testenterprise.cpom -Port 25 -Credential $creds
```

Part4