

How to Set Up Database Mail for SQL Server Job Completion

This document describes the necessary steps to setting up Database Mail in SQL Server 2012 to send an email when a SQL Server job fails or completes. These are the steps:

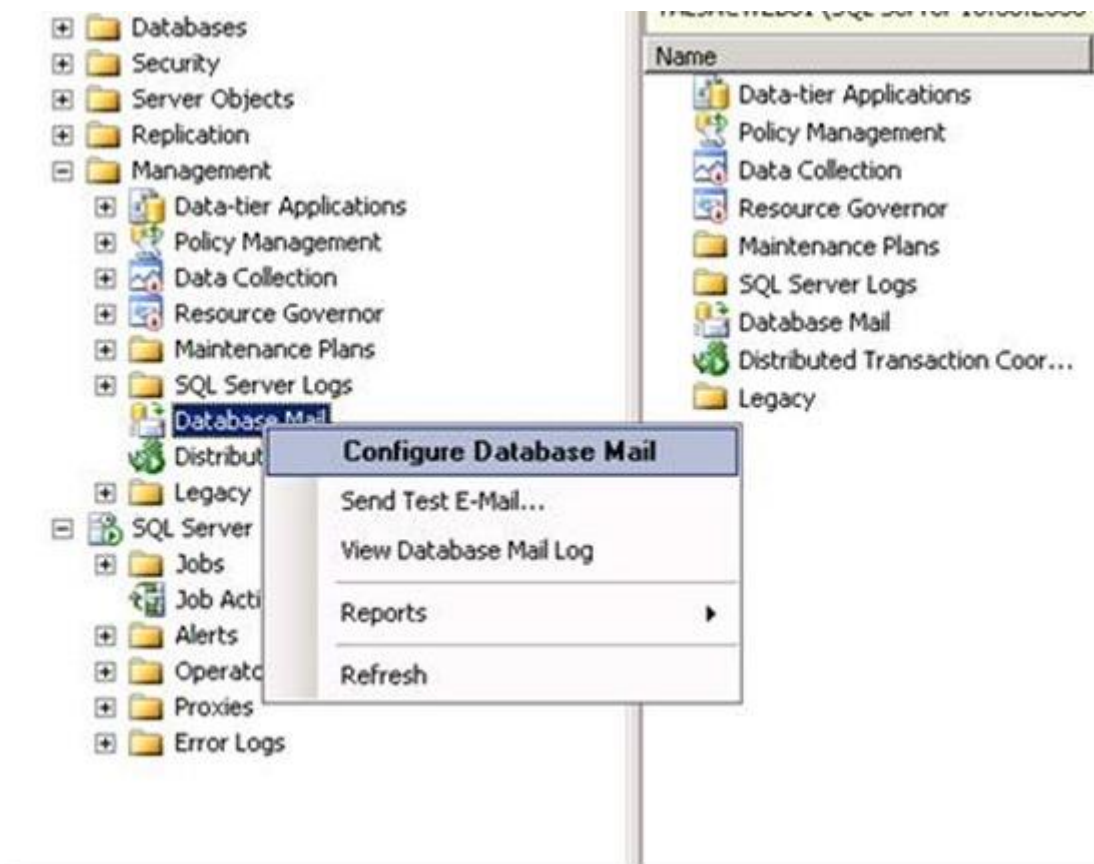
1. Configure Database Mail.
2. Create a SQL Server job.
3. Adjust the properties within the SQL Server Agent.
4. Create an Operator.
5. Adjust the SQL Server job to send on Failure.

Configuring Database Mail

To complete this step correctly, a properly configured mail server is needed. In most cases it is ok to use localhost, but that requires the installation of Microsoft IIS/SMTP. If your administrator will not allow the install of IIS on a server running SQL Server, get the name of a mail server that can be used.

Step. 1

Open SQL Server Management Studio (SSMS), connect to the Database Engine, expand Management, right-click on Database Mail and then click **Configure Database Mail**.



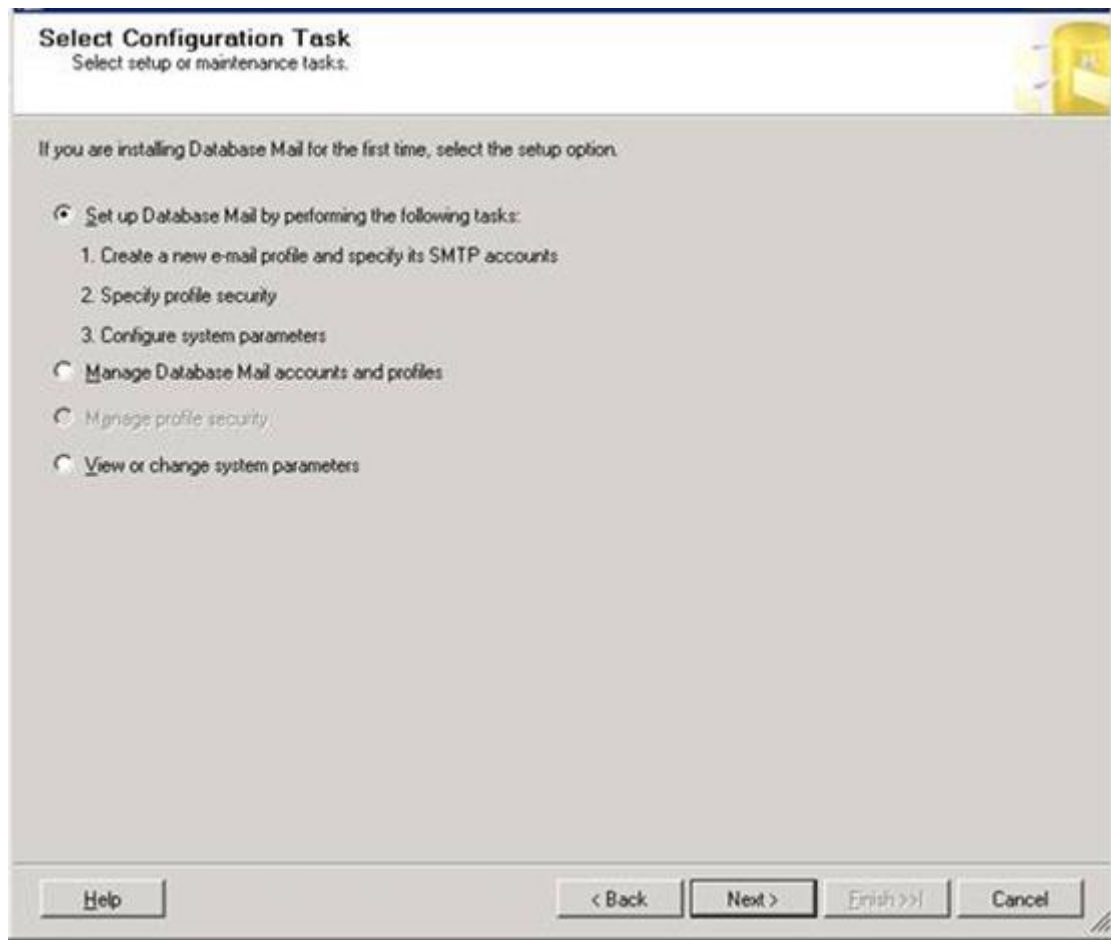
Step. 2

You will then see the splash screen below. Click **Next**.



Step. 3

Leave the default option of “Set up Database Mail by performing the following tasks” selected. Click **Next**.



Step. 4

You will get a pop-up below if mail hasn't been configured yet. Click **Yes** and then click **Next**.



Step. 5

Give the **New Profile** a name and then click **Add**.

New Profile
Specify the profile name, description, accounts, and fallover priority.

Profile name:

Description:

A profile may be associated with multiple SMTP accounts. If an account fails while sending an e-mail, the profile uses the next account in the priority list. Specify the accounts associated with the profile, and move the accounts to set the fallover priority.

SMTP accounts:

Priority	Account Name	E-mail Address
----------	--------------	----------------

Step.6

Fill in the correct information as show below.

- *Email address*– This should be the sender email address.
- *Display name*– This should be the name displayed when the email arrives.
- *Reply email*– This should be set properly too.
- *Server name*– The mail server address. In this case, I am using localhost.
- *SMTP Authentication*– I leave this section set to the default, but you can specify it to your needs if it is required.

Note that the **OK** button will remain greyed out until all required fields are completed. Once all the fields are completed, click **OK**.

Specify name, description, and attributes for your SMTP account.

Prof Account name: Email for SQL Jobs

Des: Description: Email for SQL Jobs

Outgoing Mail Server (SMTP)

E-mail address:

Display name: Desired

Reply e-mail:

Server name: 127.0.0.1 Port number: 25

This server requires a secure connection (SSL)

SMTP Authentication

Windows Authentication using Database Engine service credentials

Basic authentication

User name:

Password:

Confirm password:

Anonymous authentication

OK Cancel Help

Help < Back Next > Finish >>| Cancel

Step. 7

Click Next.

New Profile

Specify the profile name, description, accounts, and failover priority.

Profile name:

Description:

A profile may be associated with multiple SMTP accounts. If an account fails while sending an e-mail, the profile uses the next account in the priority list. Specify the accounts associated with the profile, and move the accounts to set the failover priority.

SMTP accounts:

Priority	Account Name	E-mail Address
1	Email for SQL J...	[REDACTED]

Step. 8

Check the option to make the new profile **Public**. Profiles are either public or private. A private profile is accessible only to specific users or roles. A public profile allows any user or role with access to the mail host database (**msdb**) to send e-mail using that profile. If the profile isn't set to **Public**, permissions will need to be assigned properly.

Manage Profile Security
Specify database users or roles that have access to profiles.

Public Profiles | Private Profiles

A public profile can be accessed by all users of any mail-host database.

Select public profiles. You can also specify the default public profile.

Public	Profile Name	Default Profile
<input checked="" type="checkbox"/>	Email	No

Show only existing public profiles

Help < Back Next > Finish >> Cancel

Step. 9

Review the System Parameters. I always use the default settings, but they can be adjusted as needed. You can review the settings in depth [here](#). Click **Next**.

Configure System Parameters
View or change Database Mail system parameters.

System parameters:

Parameter	Value
Account Retry Attempts	1
Account Retry Delay (seconds)	60
Maximum File Size (Bytes)	1000000
Prohibited Attachment File Extensions	exe,dll,vbs,js
Database Mail Executable Minimum Lifetime (seconds)	600
Logging Level	Extended

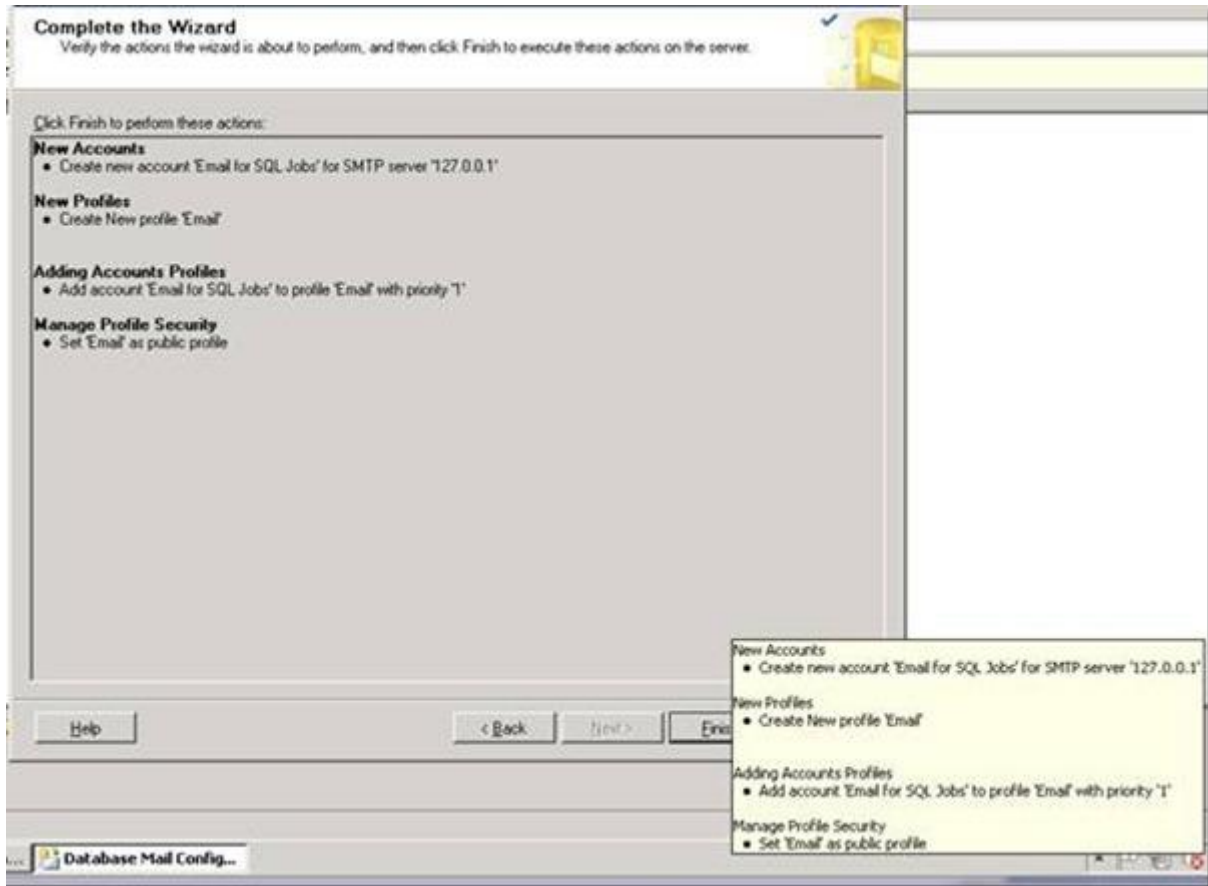
Number of retry attempts for a mail server to send email.

Reset All

Help < Back Next > Finish >> Cancel

Step. 10


Click **Finish**.







Step. 11

You should see that each *Action* was completed and has a *Status* of **Success**. Click **Close**.

Configuring...
Click Stop to interrupt the operation.

 **Success** 4 Total 0 Error
4 Success 0 Warning

Details:

Action	Status	Message
 Create new account 'Email for SQL Jobs' for SMTP server ...	Success	
 Create New profile 'Email'	Success	
 Add account 'Email for SQL Jobs' to profile 'Email' with prio...	Success	
 Grant 'guest' access to 'Email'	Success	

Test set up

Right-click **Database Mail** and select **Send Test E-Mail**. Type in an email address to use for testing and click **Send Test E-Mail**.



The screenshot shows a dialog box titled "Send Test E-Mail" with the following fields and buttons:

- Database Mail Profile:** A dropdown menu set to "Email".
- To:** An empty text input field.
- Subject:** A text input field containing "Database Mail Test".
- Body:** A text area containing the text "This is a test e-mail sent from Database Mail on" followed by a redacted line.
- Buttons:** "Send Test E-Mail" and "Close".

You will then see the screen below.



The screenshot shows a dialog box titled "Database Mail Test E-Mail" with the following content:

- Sent e-mail:** A text input field containing the number "1".
- Message:** "Your test e-mail has been queued for processing. Depending on the network speed and the backlog of the SMTP server, it may take several minutes before the e-mail is delivered to the recipient." followed by "If the e-mail is successfully received, click OK. If the e-mail is not received, click Troubleshoot."
- Buttons:** "OK" and "Troubleshoot...".

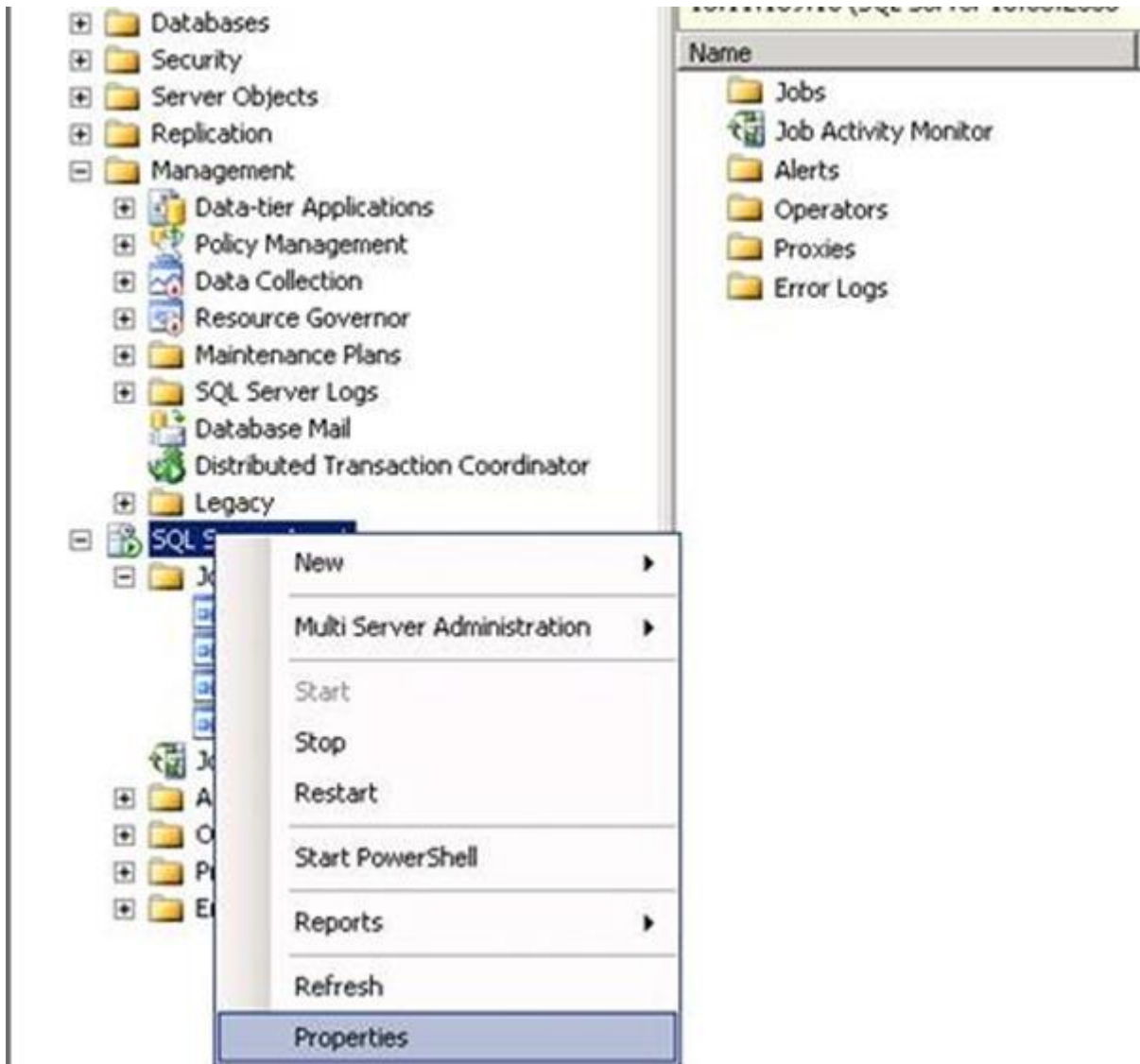
If the email doesn't arrive, check the mail server to make sure it is properly configured and will accept mail from the SQL Server.

Create SQL Server Job

At this point we know the SQL Server can send email. Now we need to setup a SQL Job and then set SQL so it sends emails if the job fails. The easiest way to test this is to setup a Backup job and set it to notify on "Success". Once it is working properly, change the job to notify on Failure or Completed.

Adjust the properties within the SQL Server Agent

Right-click SQL Server Agent and select Properties.

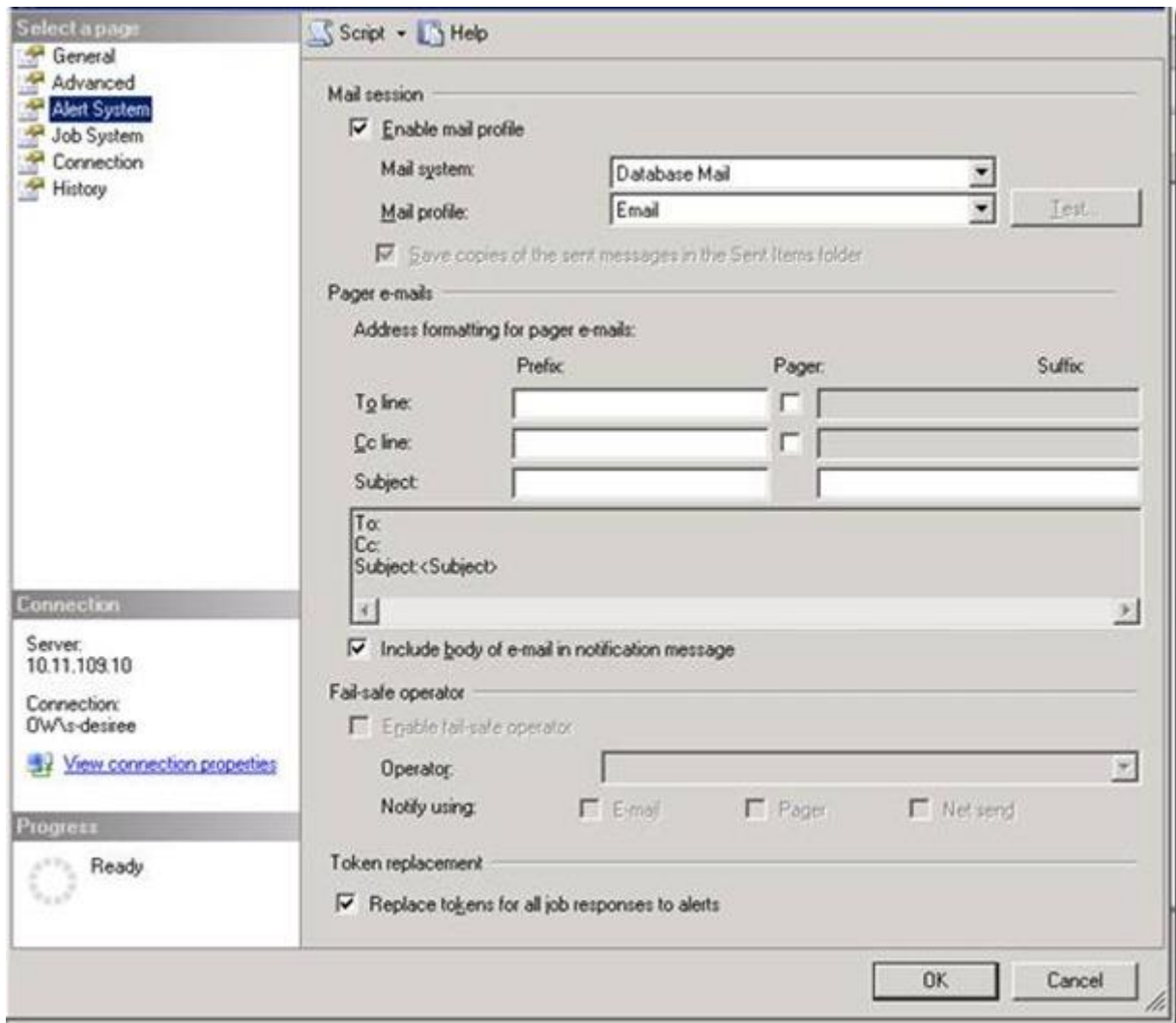


Click on **Alert System** under *Select a page*.

· Under *Mail session*, select the option to Enable mail profile. Ensure the correct Mail profile is selected.

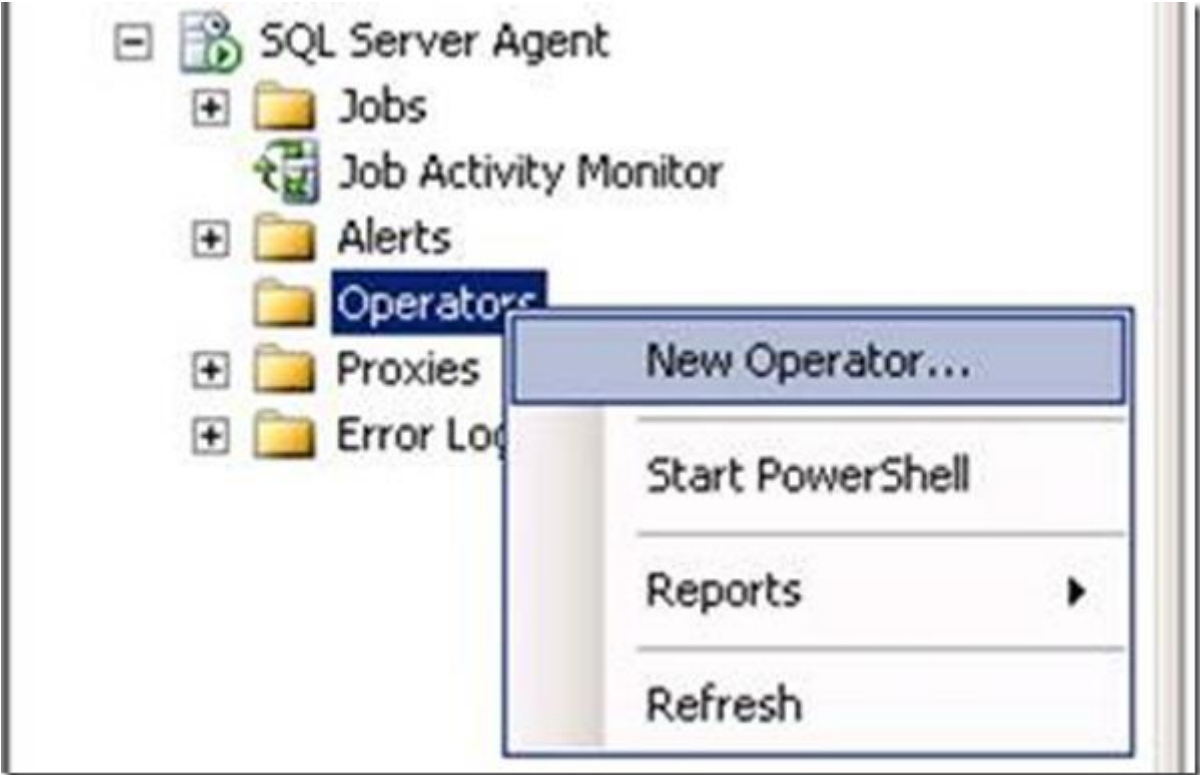
· Under *Token replacement*, enable Replace tokens for all job responses to alerts.

Click **OK**. Restart the SQL Server Agent service.

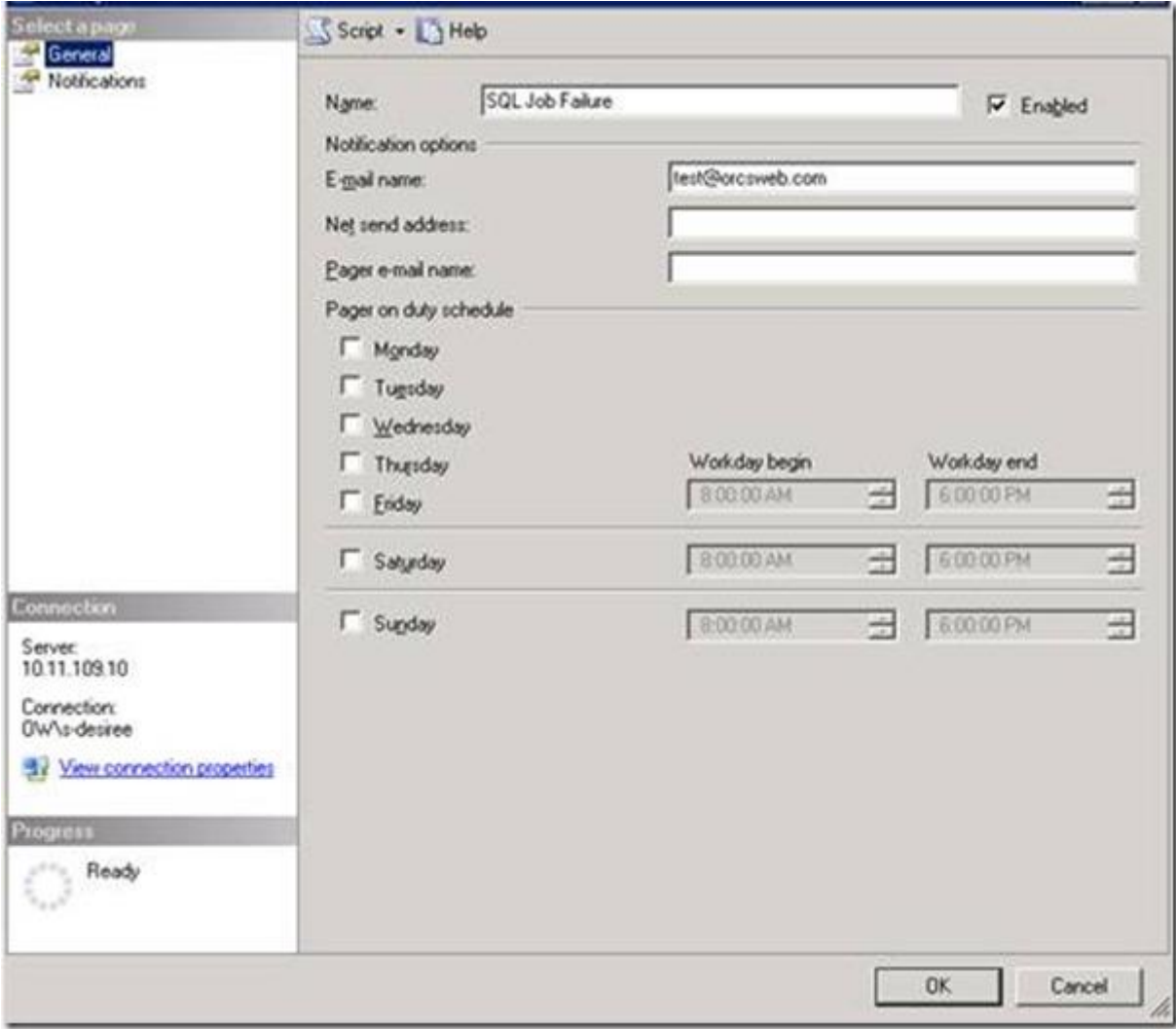


Create an Operator

Under the SQL Server Agent, right-click **Operators** and select **New Operator**.



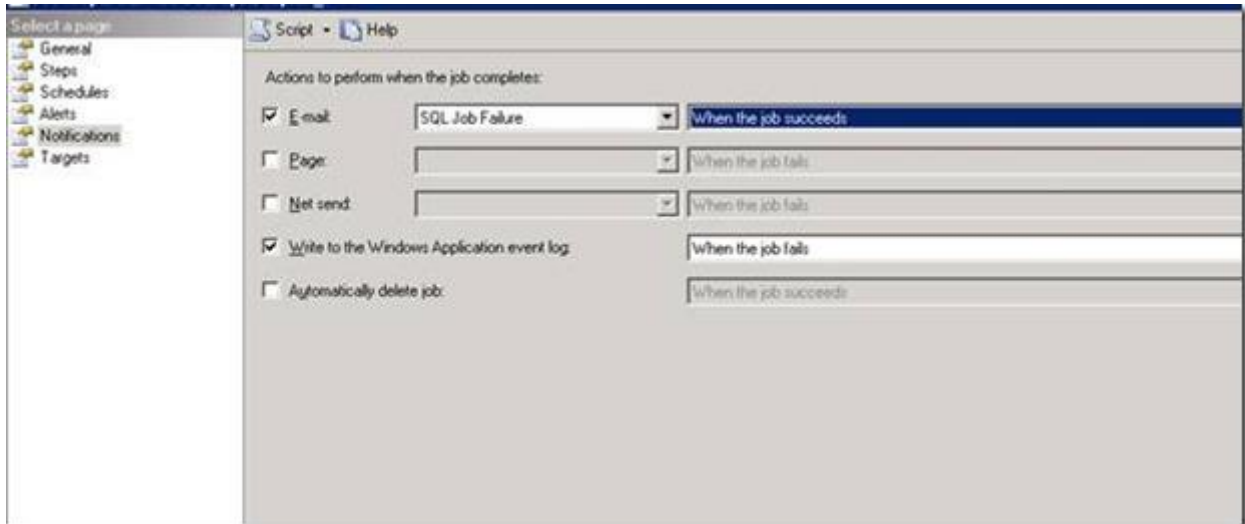
Type in the recipient email address in the E-mail name and click OK. Separate multiple recipient addresses with semi colon if applicable.



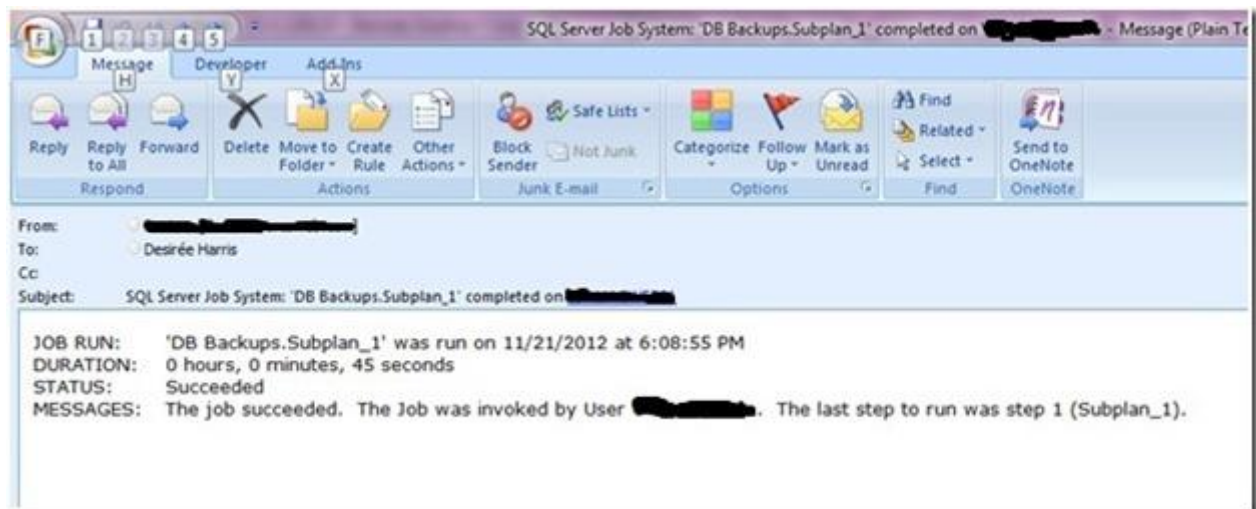
Adjust the SQL Server job to send the email on Success (for testing purposes only).

Expand SQL Server Agent and click on Jobs. Find the job you created above, right click it and select **Properties**.

Under Select a page, select Notifications. Enable the first option, E-mail. Then select the Operator we just created and change the job to email when the job succeeds (for testing purposes only). Click OK.



Test by doing the following: Run the test backup job created. You should receive an email similar to the one below once it completes:



If an email isn't received, restart the SQL Server Agent once more and then test again.

Adjust the SQL Server job to send on failure

After the test succeeded go back and change the SQL Server job.

Expand SQL Server Agent and click on Jobs. Find the job you created above, right click it and select Properties.

Go to the first option, E-mail. Change the job to email When the job fails Click **OK**

