

Implementing Policy Based Management: A Primer in SQL Server 2008

By Anil Mahadev



Senior SQL Server DBA / Infrastructure Lead

Elite SQL Server Club Member

Introduction

Policy Based Management was previously known as the Declarative Management Framework (DMF) and has made its name by ensuring compliance inside your SQL Server databases.

Why Compliance?

Today organizations are facing challenges for staying compliant. You either need tools, or the knowledge to use those tools, in order to ensure compliance in your company's infrastructure.

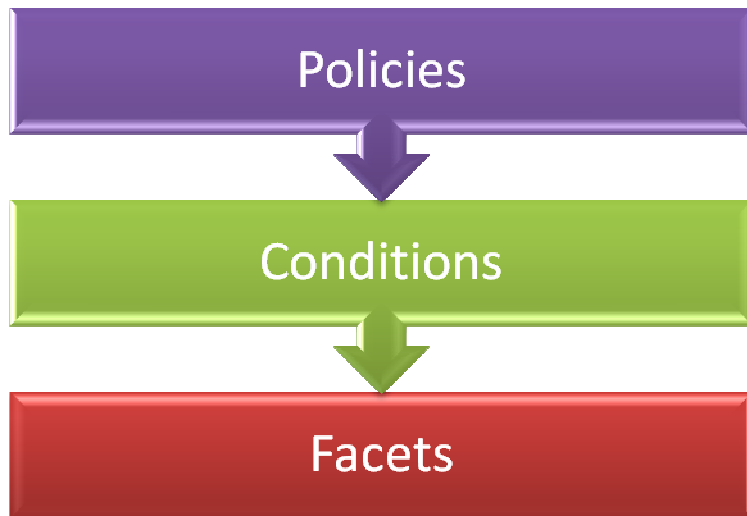


Globally, compliance is considered a serious business as it helps prevent unauthorized access to the resources and people that use the company's infrastructure.

Let's take a look at a figure below:



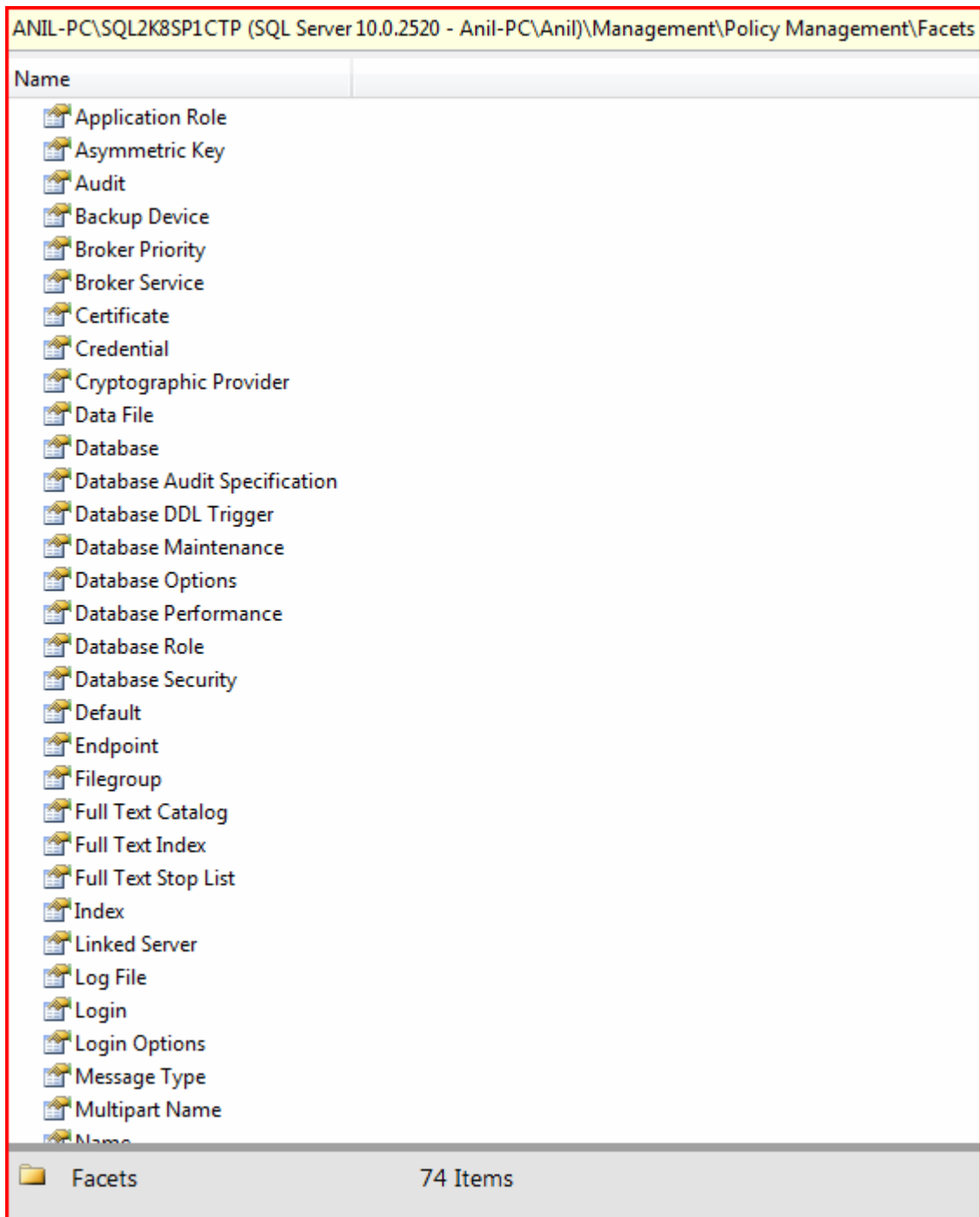
From here on I will be referring to Policy Based Management as PBM. For implementing PBM, you will need to understand the following three components:



At the foundation of it all are facets. Facets are basically a sub-system of pre-defined database and server properties.

Facets

A facet can have multiple properties. There are more than 70 facets in SQL Server 2008 as you can see from the figure below:



For example, the database facet contains multiple properties - one of them is the Owner property.

Once chosen this property will return the owner of the database.

Conditions

A condition(s) validates a facet or a series of facets.

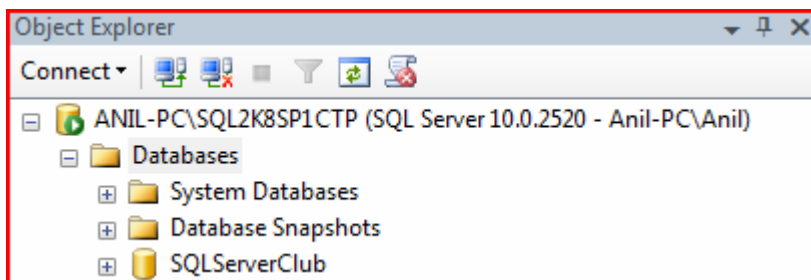
Policies

Policies implement conditions based on the facets defined.

Implementation

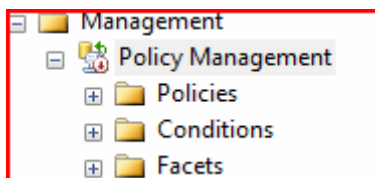
We shall now create a new database called SQLServerClub and apply a policy that ensures that the Database Recovery model is set to FULL Recovery:

- 1) Click on Start → All Programs → SQL Server 2008 → SQL Server Management Studio
- 2) Now create a new database called SQLServerClub and click on Ok.
- 3) You should now have a database SQLServerClub as shown in the figure below

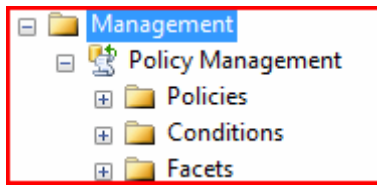


- 4) Now navigate to the Management folder and expand it. PBM is present at the first instance.

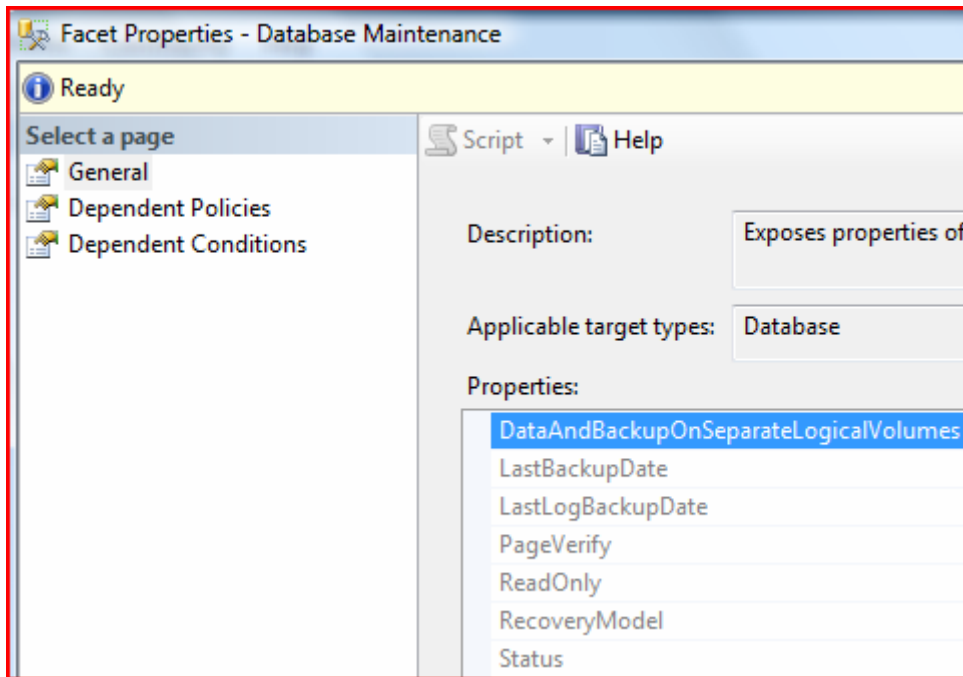
You will need to identify if it's enabled or disabled. The figure below indicates when PBM is disabled:



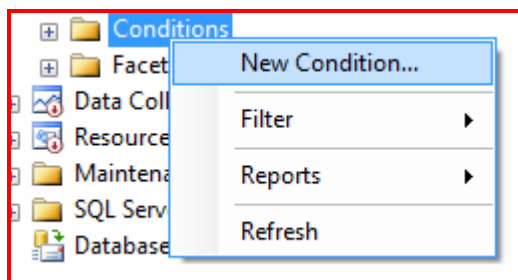
The figure below indicates when PBM is enabled:



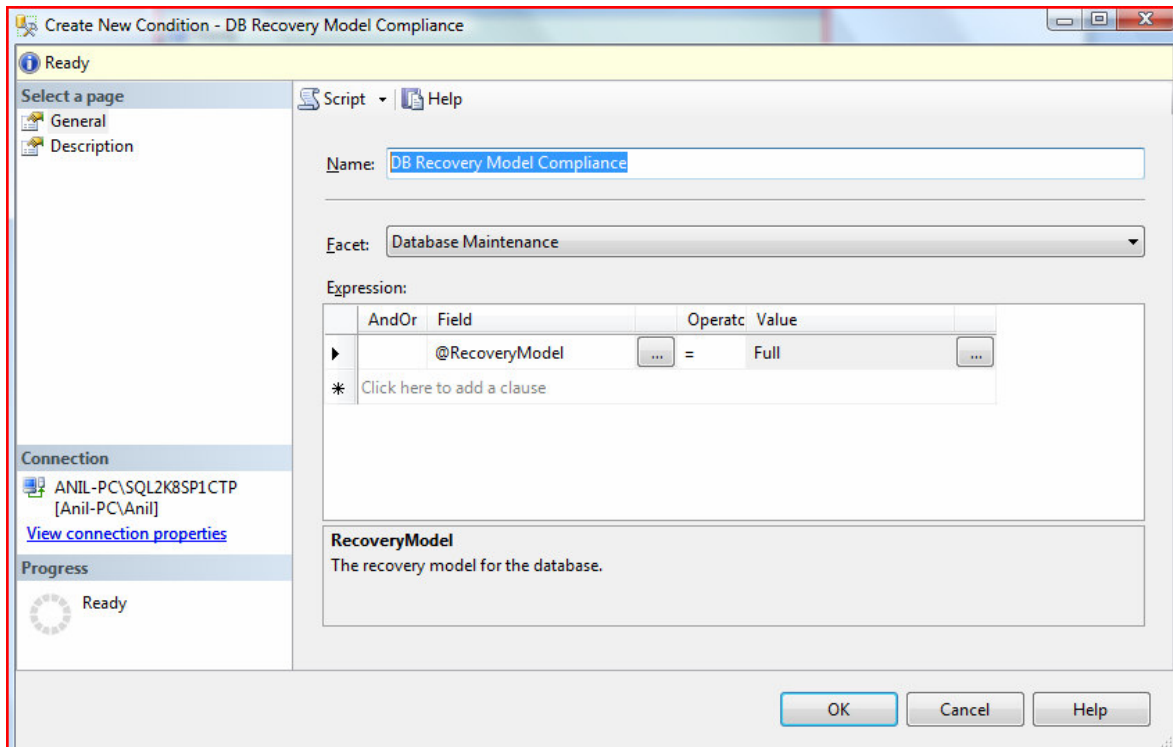
- 5) Now first we need to understand which facet enables us to check the Database Recovery Model. As you can see below the Database Maintenance Facet contains the Recovery Model:



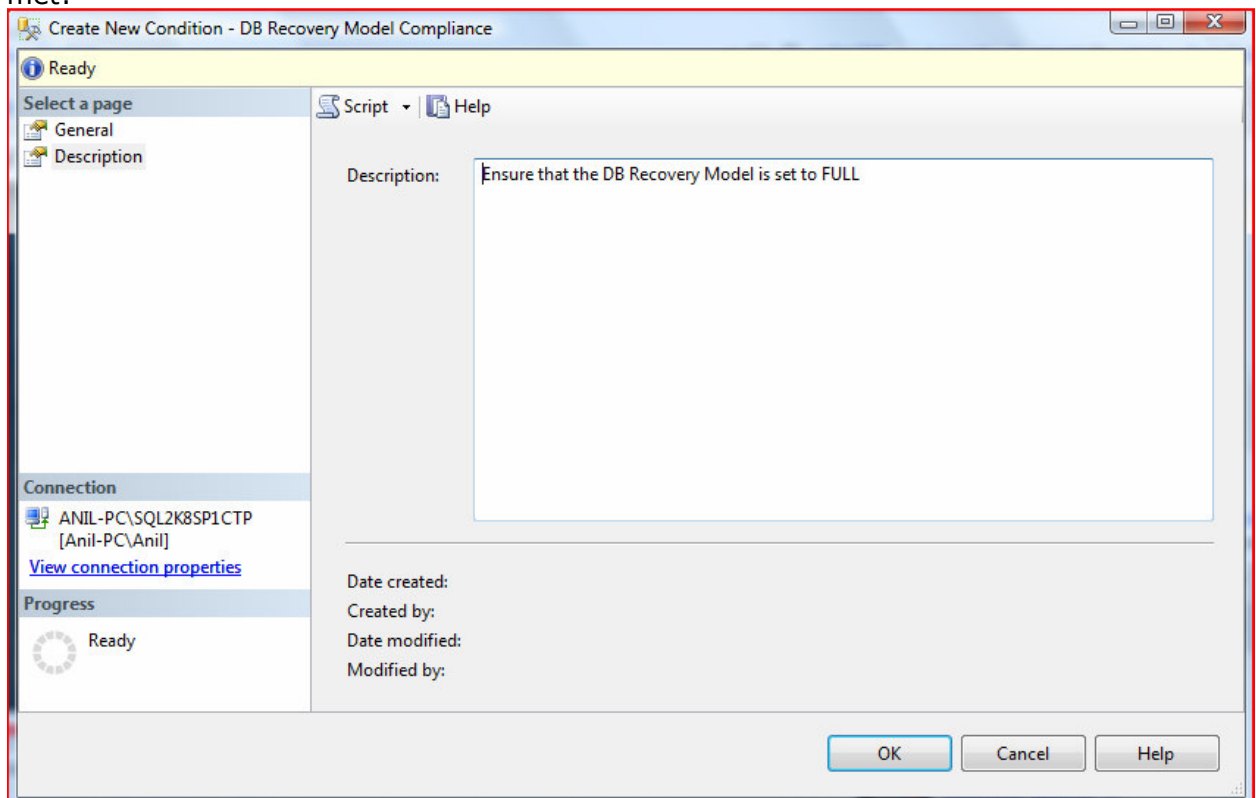
- 6) Now right click on Conditions → Click New Condition



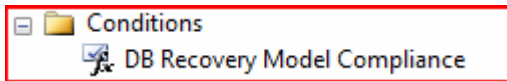
- 7) Now in the General section, enter the name for the condition:
Name: DB Recovery Model Compliance
Choose a Facet: Database Maintenance
Expression: Recovery = FULL



8) Now we need to give this condition a description in the event the compliance is not met:

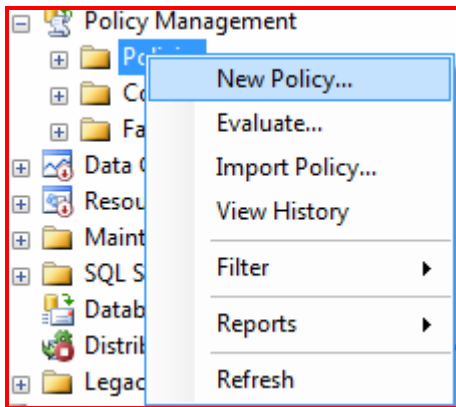


9) Now you can locate the new condition under the Conditions Folder:



10) Now that we have our condition in place, we will create a Policy that evaluates this condition.

11) Right Click on the Policy folder and choose New Policy:



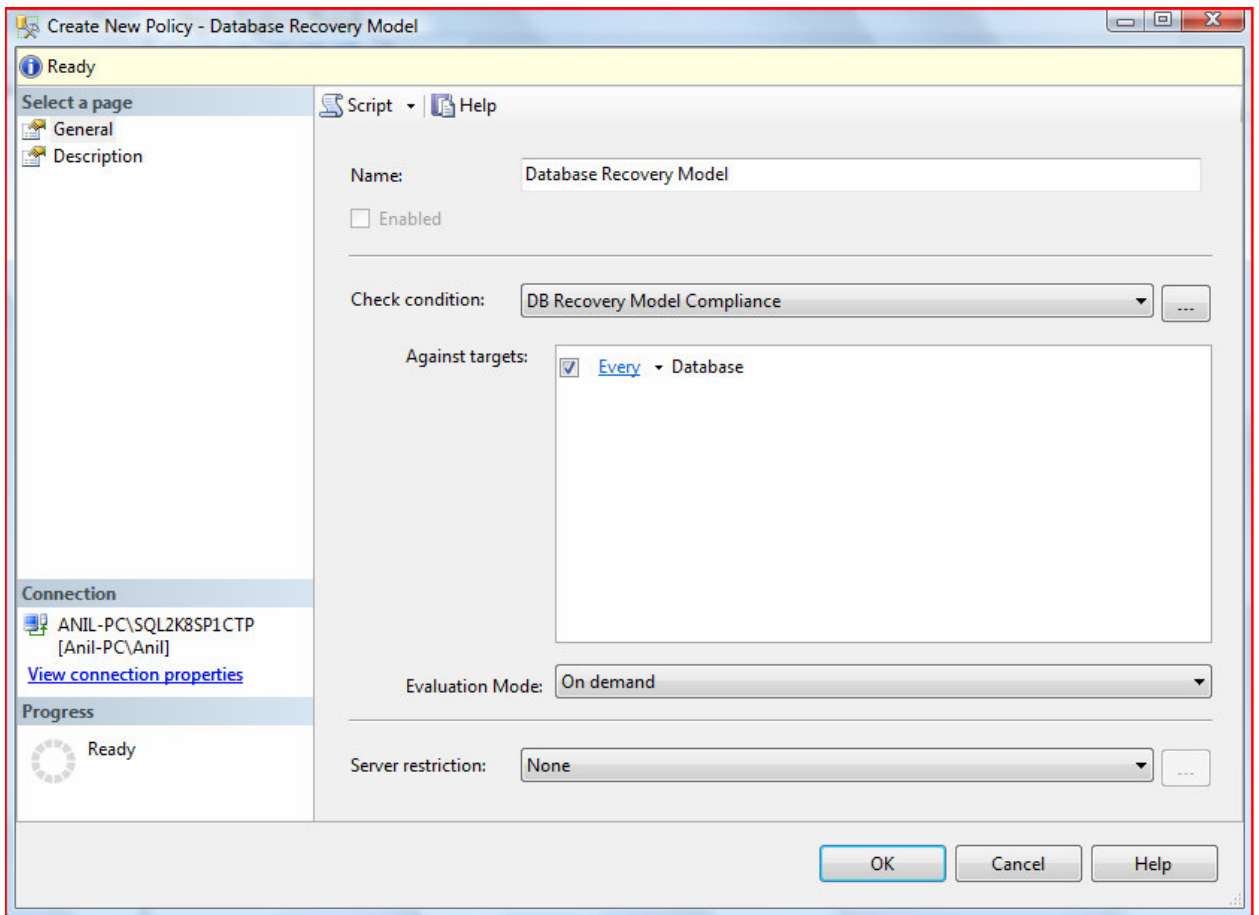
12) Now give the Policy name "**Database Recovery Model**". Choose the condition: DB Recovery Model Compliance.

Now under against targets → Choose Every Database

The Evaluation Mode can be of two types: **On Demand (Default) and On Schedule**

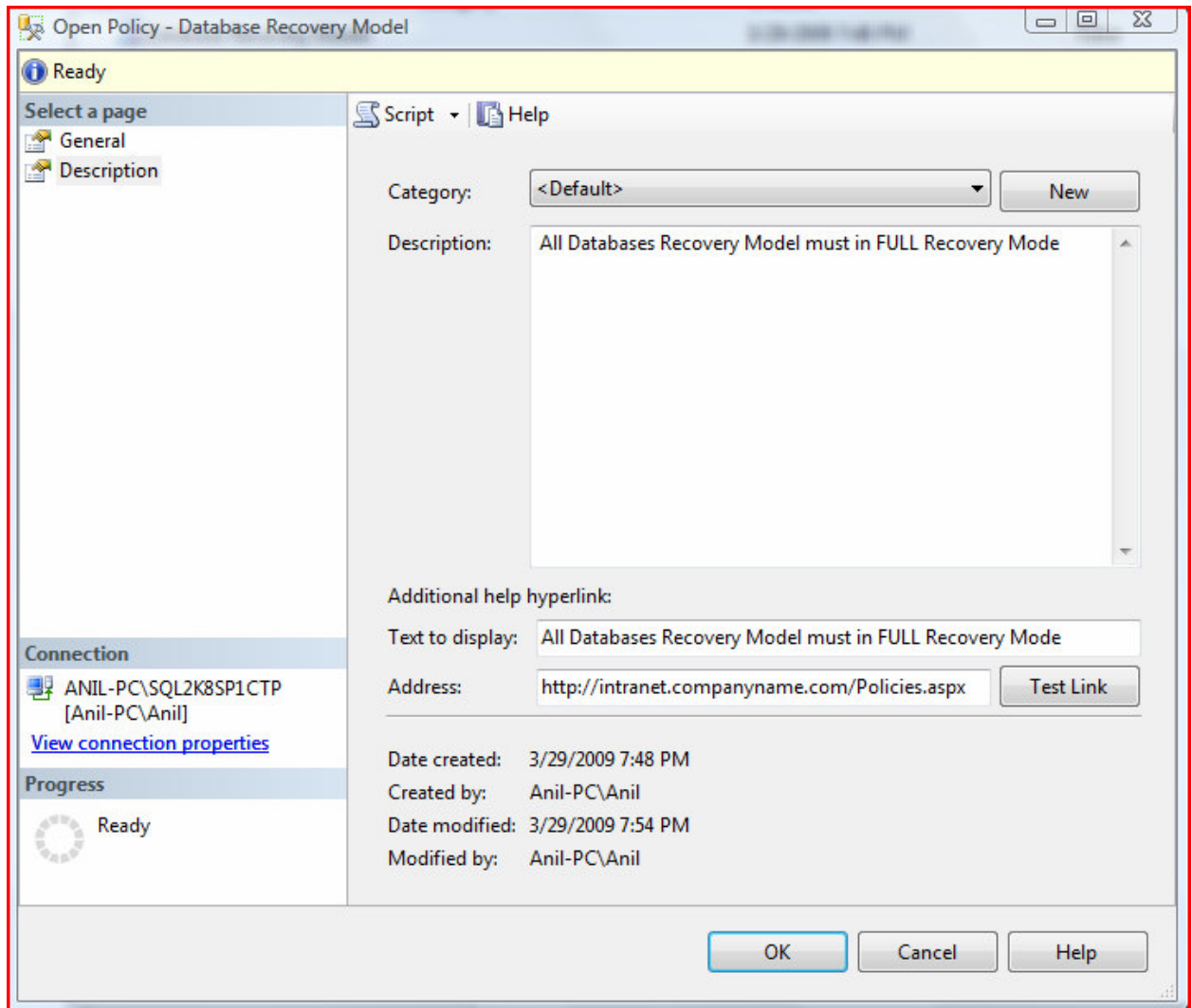
Leave the default as On Demand.

Server Restriction: **None**



13) Under the Description page leave the Category as default.

Description: You can write any custom message you wish to convey to the user. Under the **Text to Display:** You can write a custom message saying to either contact the DBA or look at a specific URL of the organization that defines these policies:

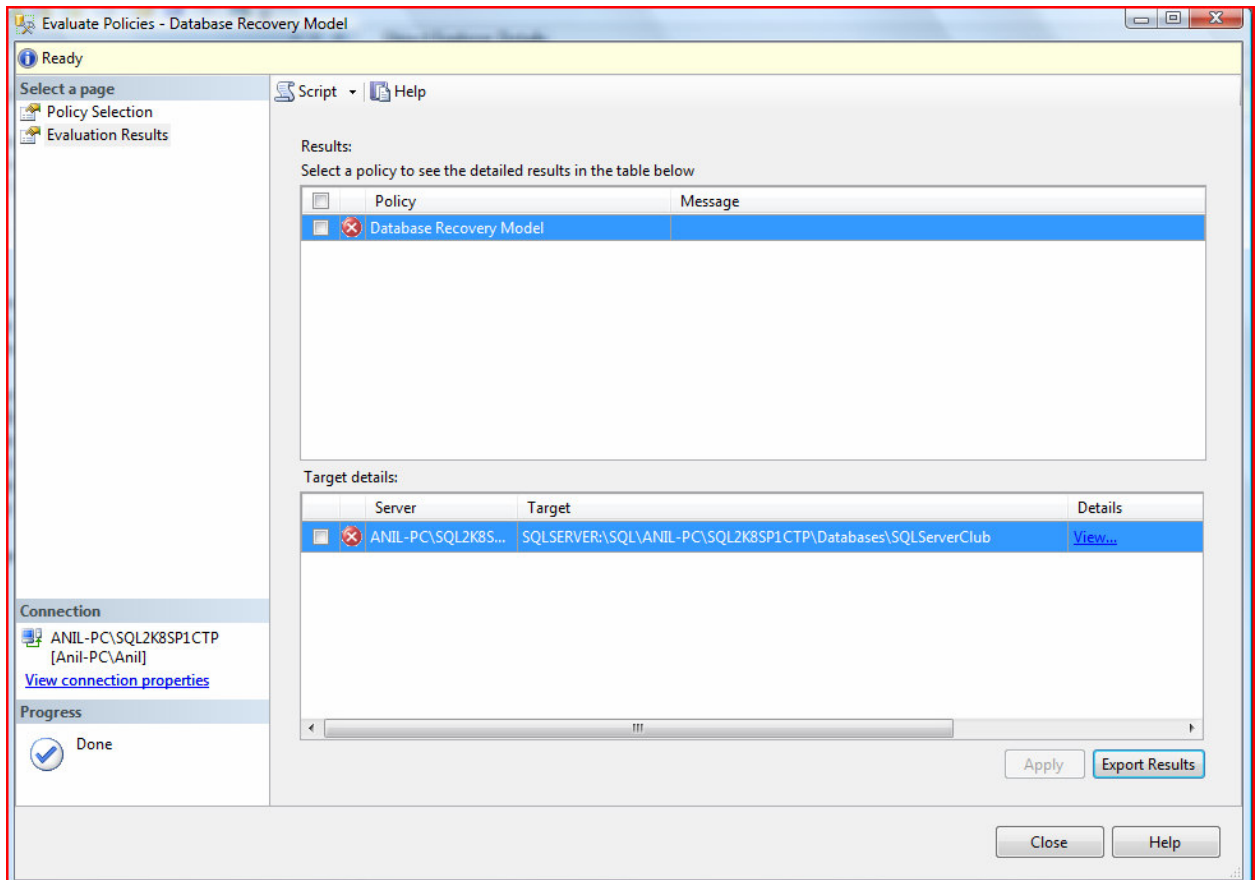


14) Now it's time to evaluate our policy to ensure it is working 😊!

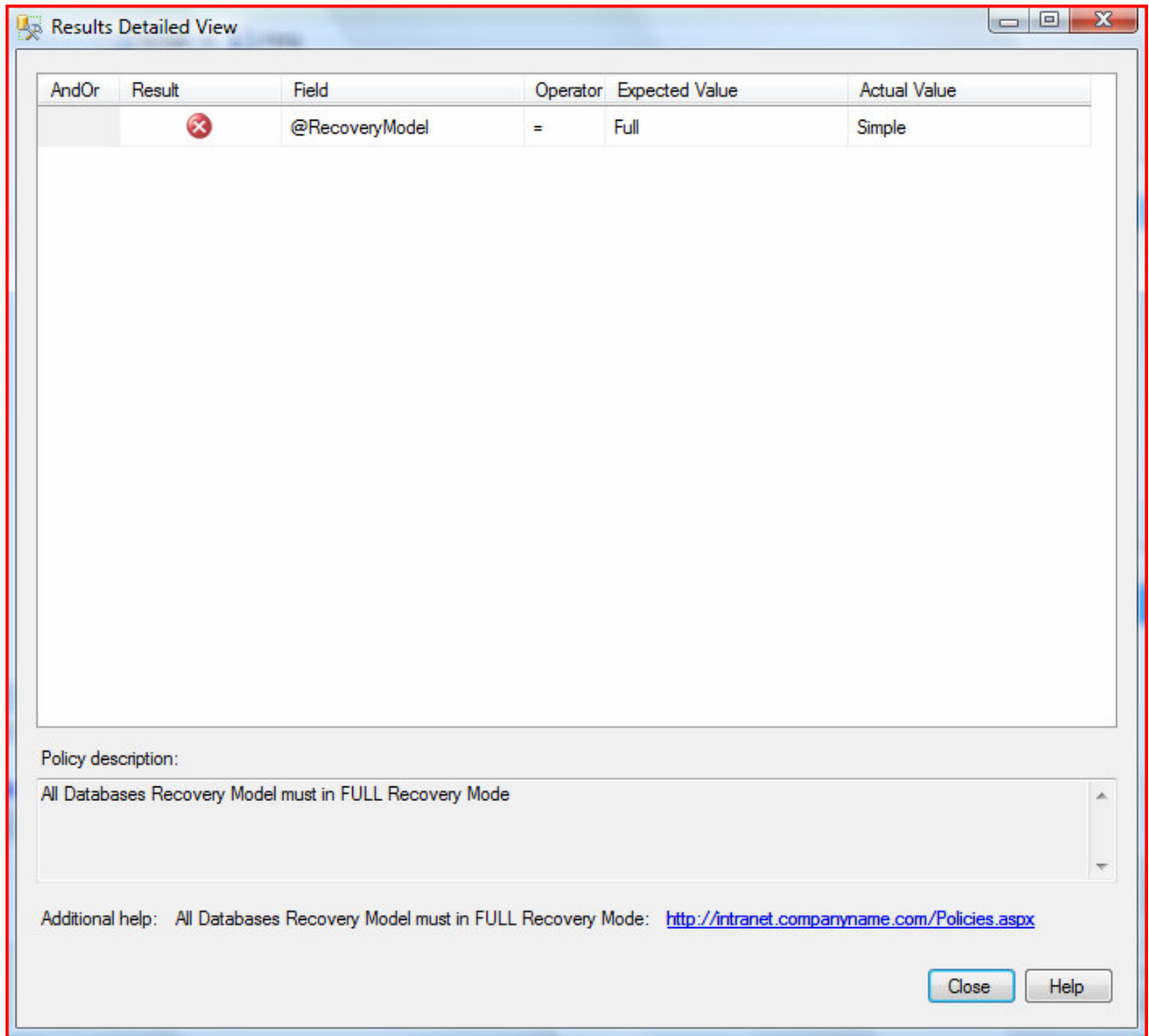
15) For demonstration purposes, I intentionally made the SQLServerClub Database Recovery mode to **Simple**.

16) To evaluate this, right click on your policy and choose **Evaluate**.

17) You should get an error as shown below:

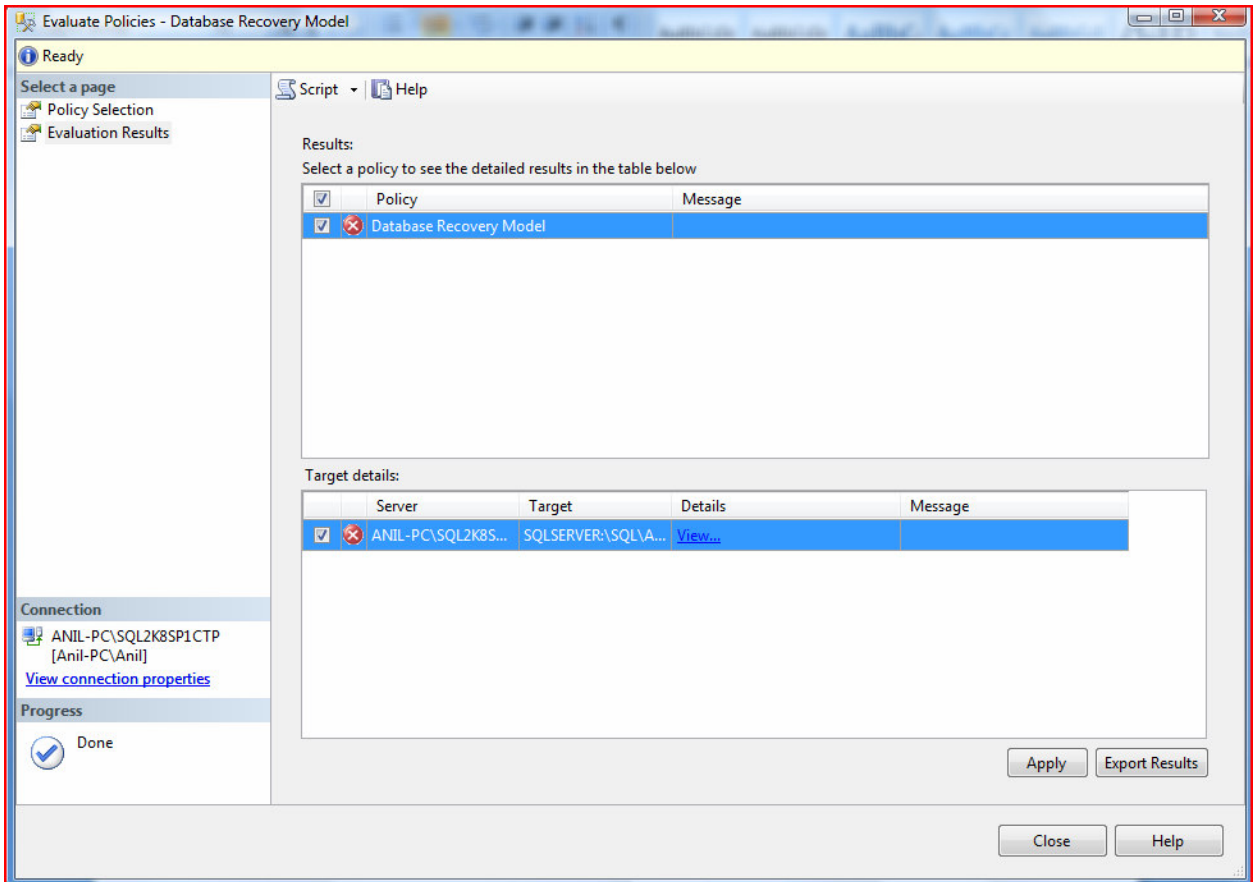


Now click on the **View** hyperlink to investigate the reason for the compliance failure.

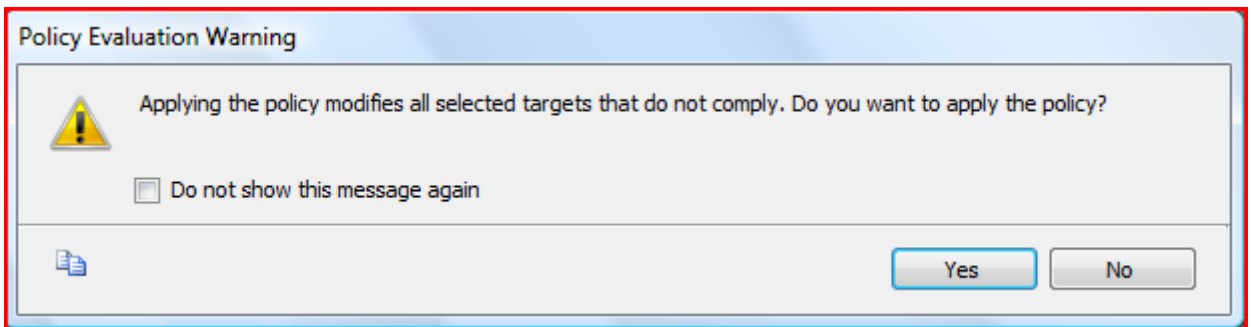


As you can see, the expected value was **Full** and the actual value was **Simple**.

18) Now let us get our database into a compliant state. To do this, click on the checkbox of the policy and click on **Apply**.



You will now get a warning, asking if you wish to apply the policy:



Click on the **Yes** button.

19) Then you will get a check mark as shown below:

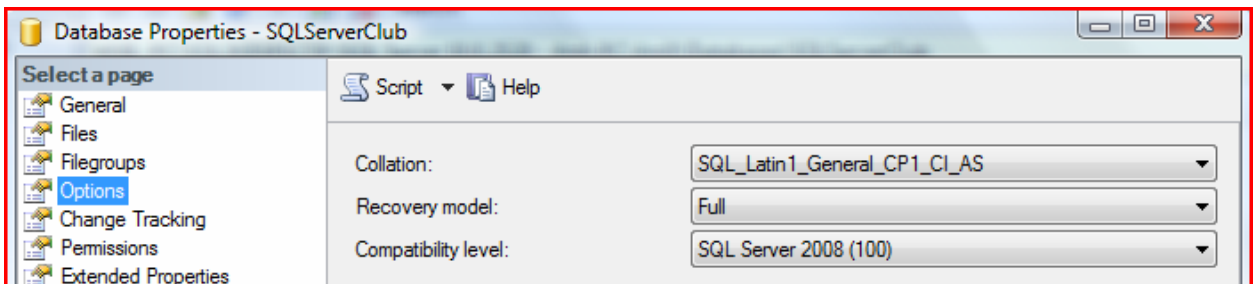
Results:
Select a policy to see the detailed results in the table below

<input type="checkbox"/>	Policy	Message
<input checked="" type="checkbox"/>	Database Recovery Model	

Target details:

<input type="checkbox"/>	Server	Target	Details
<input checked="" type="checkbox"/>	ANIL-PC\SQL2K8S...	SQLSERVER:\SQL\A...	View...

20) Right click on the SQLServerClub database and choose **Properties**. You should then navigate to the **Options** Page and see that the Database Recovery Model is now set to **Full**:



You have successfully implemented PBM☺! There are many other things to explore.



In the forth coming series, I shall explore Business Requirements implementation using PBM.

Stay tuned!

Please send me your valuable feedback to anil.mahadev2009@live.com