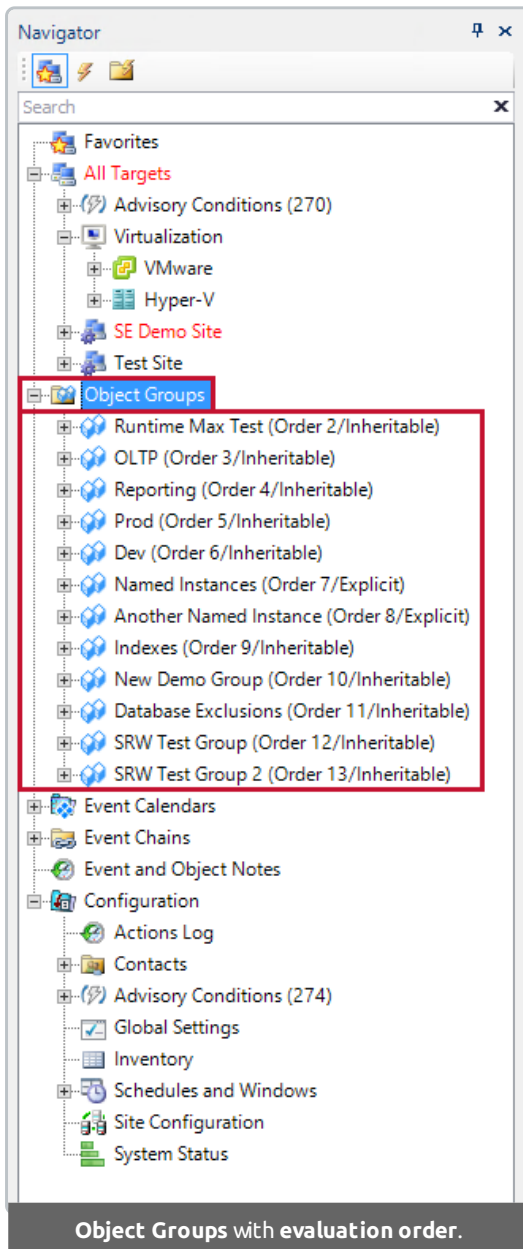


Object Groups

Last Modified on 30 March 2022

Introduction

Object Groups are useful when you need to apply similar policies to a set of objects that don't exist within the same hierarchical group in your SQL Sentry environment. These objects could be dispersed throughout your environment in different sites, groups, or subtrees, but if they share a similar set of characteristics, they may be ideal candidates for a similar set of policies.



Take for example all of the transaction log backup jobs within your environment. You may wish to suppress **Runtime Threshold** alerts generated when these jobs run by disabling the **Send Email** action. **Object Groups** allow you to do this without affecting any other agent jobs. In the past, this would involve you having to go and touch each of the transaction log backup job settings individually. **Object Groups** allow you to avoid the tedious process, saving you time because you can quickly locate objects by name or category, add them to a group, and apply a set of policies.

As **Object Groups** exist outside of the normal SQL Sentry hierarchy, a condition or setting configured for an **Object Group** is applied last after any inherited or explicitly defined condition or setting. In this way, **Object Group** conditions and settings have the final say within your environment.

As mentioned earlier, typical uses include disabling **Runtime Threshold** alerts for all transaction log backup jobs and applying policies to QA servers that may exist within several sites of your environment. When you create an **Object Group**, you choose how configured conditions and settings are applied in the group. You can create an **Object Group** in which configured conditions and settings apply strictly to the group members. Alternatively, you may create an **Object Group** in which configured conditions and settings apply to the group members and their children. For more information, see the following section **Creating New Object Groups**.

Evaluation Order

Each **Object Group** you create also has an associated evaluation order. This evaluation order is specific to just the set of **Object Groups** in your environment. This is valuable when you have an object that belongs to multiple **Object Groups** with a divergent set of policies. The evaluation order allows you to control which group's configured conditions and settings are applied. A configured setting or condition from the **Object Group** with the highest evaluation order will be applied last, and therefore will be the effective setting or condition. For example, if you have ten object groups, with orders 1 through 10, then 10 will be the highest/last applied.

Note: With object groups, the alert is applied to the first in the list. For example, if you have a *Top SQL: Duration Threshold Max* alert and it was applied in 2 object groups; *group 1* and *group 5*, then the alert in *group 1* will eclipse the alert in *group 5*, so only the *Top SQL: Duration Threshold Max* alert defined in *group 1* is sent.

Settings

When you add new objects to an **Object Group**, any setting that is applicable to that object type will be configurable in the **Settings** pane. By default, each object in an **Object Group** still receives its base set of inherited settings from its parent objects in the SQL Sentry hierarchy. If you would like to override these settings, define a setting within the **Object Group** itself. Change the **Defined** setting to **True**. After you define a setting for an **Object Group**, that setting is reflected in any applicable objects that belong to that group.

Conditions

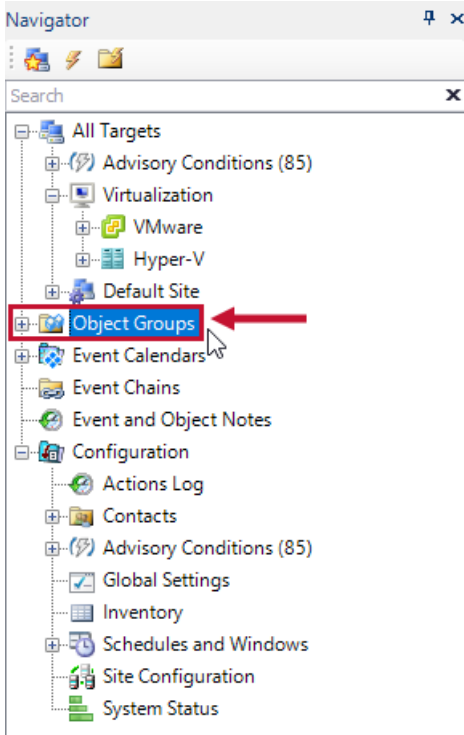
Just like settings, condition/action combinations become available as appropriate when objects are added to a group. With any **Object Group** selected, you can configure conditions through the **Conditions** pane. For more information, see the following section **Configuring Conditions and Settings for an Object Group**.

Creating New Object Groups

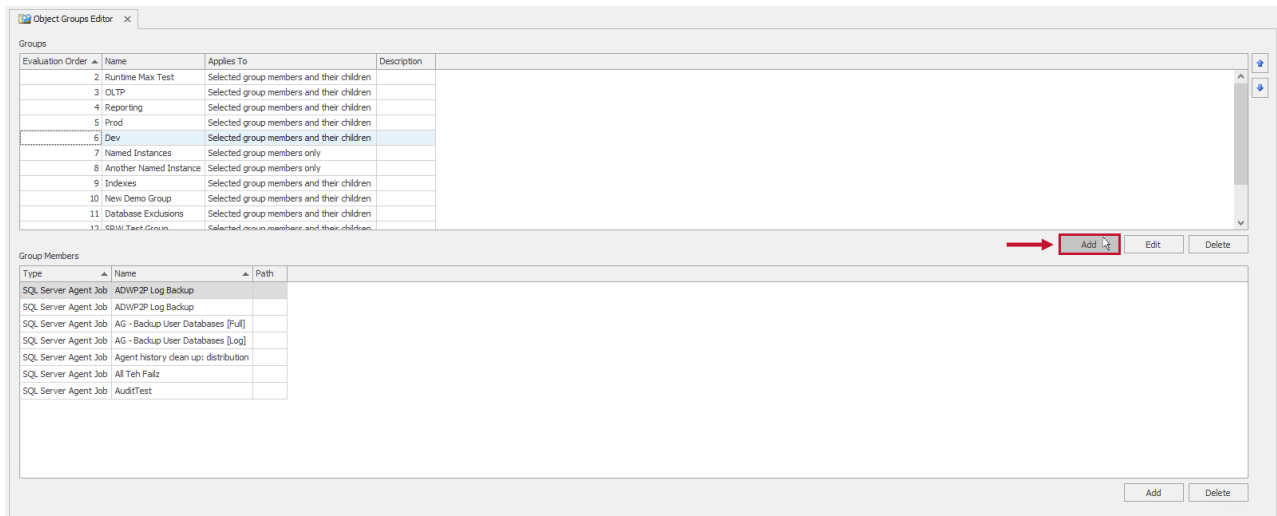
[Additional Information:](#) See the blog post on [Creating Object Groups in SQL Sentry](#).

To create a new **Object Group**, complete the following steps:

1. Double-click the **Object Groups** node in the **Navigator** pane to open the **Object Groups Editor** window.

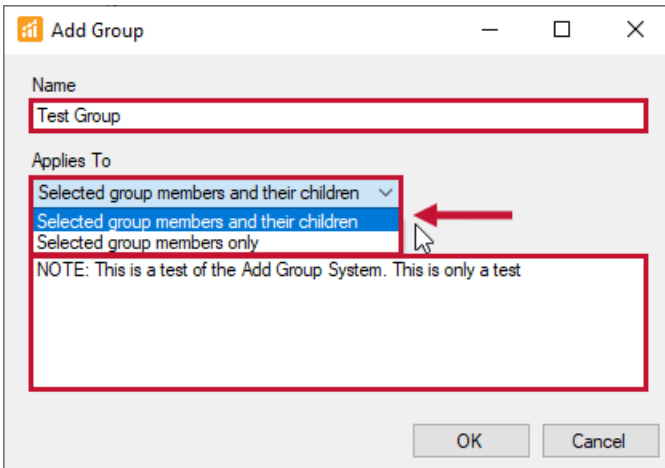


2. Select **Add** in the **Groups** section of the **Object Groups Editor** window to open the **Add Group** dialog window.



Note: The **Groups** section of the editor window contains any previously configured groups in your environment. The **Group Members** section of the editor window allows you to view individual members of any selected Object Group.

3. Enter a name and description for the group. Also take note of the **Applies To** drop-down box. The following options are available:

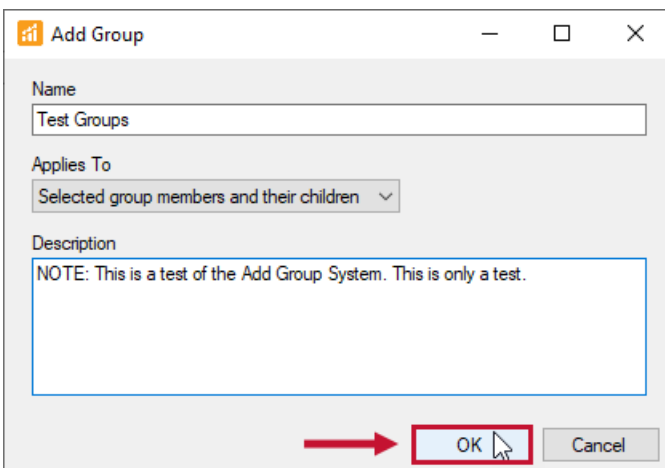


Option	Description
Selected group members and their children	Any conditions or settings you configure for the group also apply to the children of the group members. This makes the group's settings and conditions inheritable.
Selected group members only	Any conditions or setting you configure for the group only apply to the specific members of the group and have no effect on their children. This makes the group's settings and conditions noninheritable.

Note: If you're creating an **Object Group** to set policies for all of the QA servers in your environment, choose **Selected group members and their children** because this gives you access to applicable settings and conditions from the instance level down to the object level.

If you're creating an **Object Group** to set policies for just a finite set of objects and do not wish to impact or have access to children of the object, choose **Selected group members only**. Given the transaction log backup job use case discussed earlier, it wouldn't matter which **Applies To** option you chose because the agent job object is a leaf node in the SQL Sentry hierarchy.

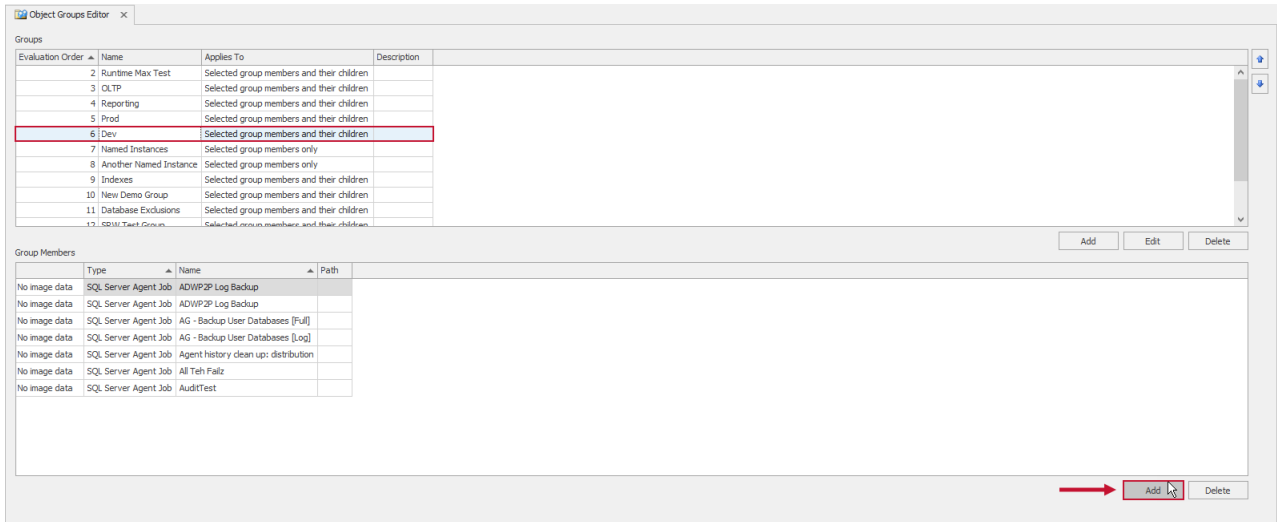
4. Select **Ok** to add your Object Group.



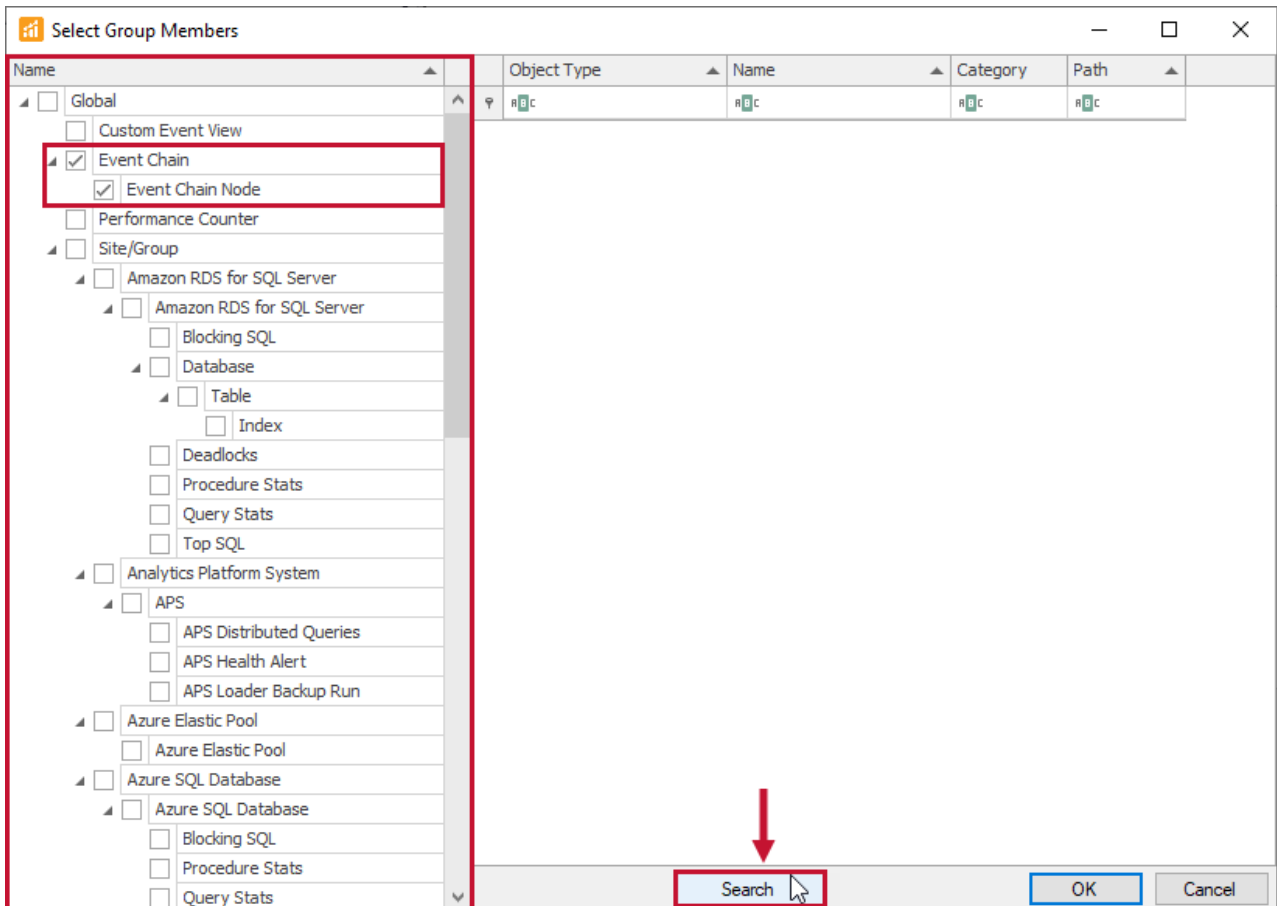
Adding Object Group Members

To add new members to a group:

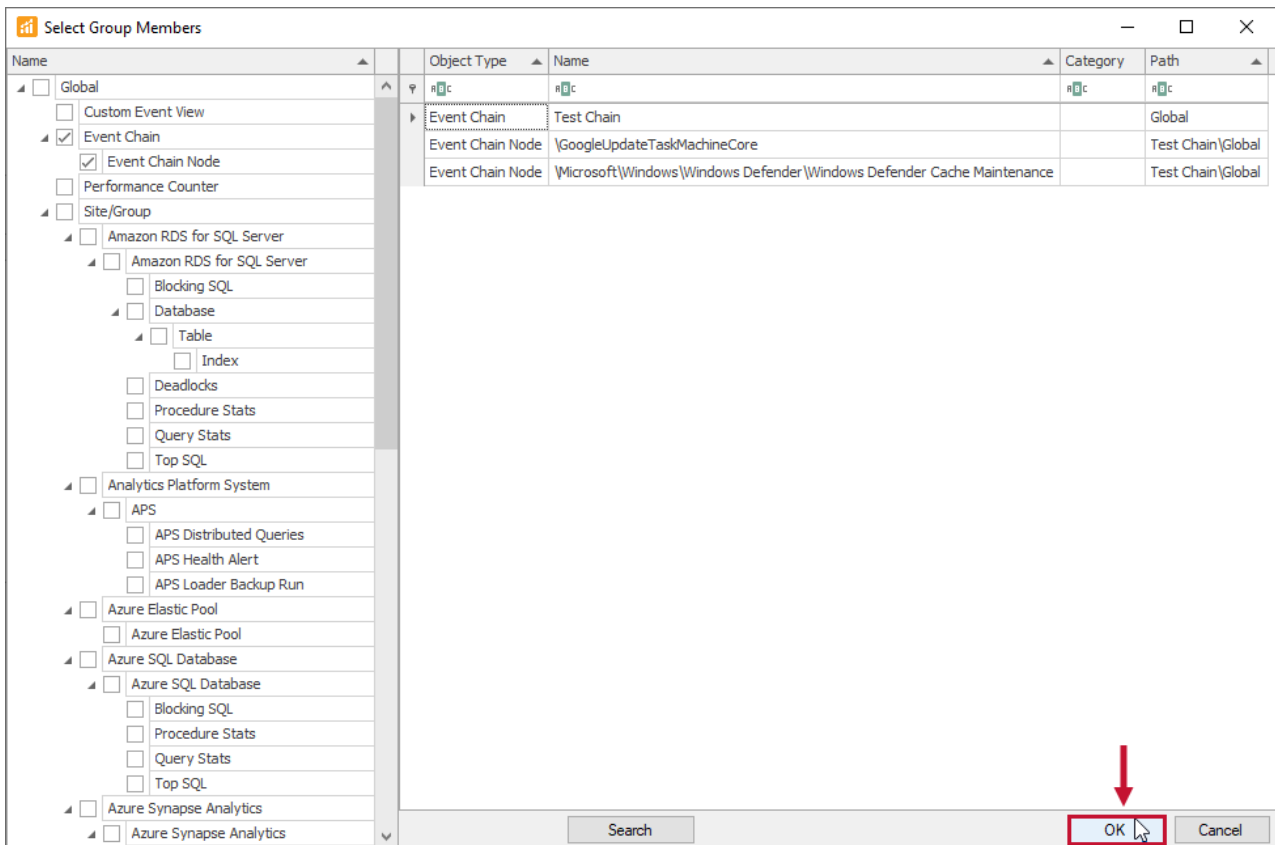
1. Select a group in the **Groups** section of the editor window, and then select **Add** in the **Group Members** section of the editor window.



2. In the **Select Group Members** dialog screen, choose the object type of the group member you'd like to add from the tree view, and then select **Search** to view objects of that type in your monitored environment.



3. Select an object, and then select **OK** to add it to the group.



Note: You can add multiple objects by pressing **Ctrl** or **Shift**.

Configuring Conditions and Settings for an Object Group

Once you've added your desired objects to the group, you can begin configuring the conditions and settings for the group. Configure conditions and settings, through the **Conditions** and **Settings** pane. If you don't see the **Conditions** and **Settings** pane, use the **View** (View > Conditions) menu to open it.

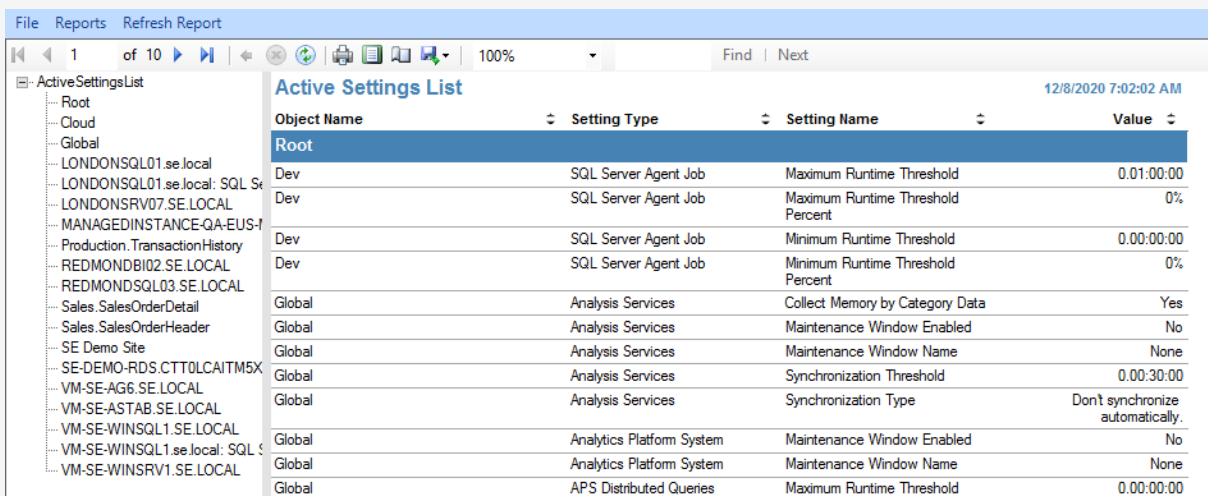
To add a condition to a group, complete the following steps:

1. Select **Object Groups** in the Navigator Pane, and select the desired Object Group.
2. Open the Conditions pane, and then select **Add** to open the Actions Selector window.
3. Choose your desired condition/action combinations, and then select **OK**.

Note: Any conditions that are applicable to the objects within your group are available for you to configure. You can also configure settings for the **Object Group** through the **Settings** pane. Any settings that are applicable to the objects in your group are available for you to configure. As discussed earlier, by default, each object in an **Object Group** still receives its base set of inherited settings from its superior parent objects in the SQL Sentry hierarchy. If you would like to override these settings, you must first define a setting within the **Object Group**. To do this you need to set the **Defined** flag to true in the **Settings** pane. After you define a setting for an **Object Group**, that setting is reflected in any applicable objects that belong to that group.

[Additional Information](#): For more information about **Object Groups**, see the [Manipulating Runtime Thresholds with Object Groups](#) blog post.

Note: If you're ever unsure of where an object is receiving its inherited settings, you can check. The **Active Settings List** report lists this information for each setting in the **Object Name** column. Access this report from the **Reports** menu (**Reports > General > Active Settings List**).



Object Name	Setting Type	Setting Name	Value
Root			
Dev	SQL Server Agent Job	Maximum Runtime Threshold	0.01:00:00
Dev	SQL Server Agent Job	Maximum Runtime Threshold Percent	0%
Dev	SQL Server Agent Job	Minimum Runtime Threshold	0.00:00:00
Dev	SQL Server Agent Job	Minimum Runtime Threshold Percent	0%
Global	Analysis Services	Collect Memory by Category Data	Yes
Global	Analysis Services	Maintenance Window Enabled	No
Global	Analysis Services	Maintenance Window Name	None
Global	Analysis Services	Synchronization Threshold	0.00:30:00
Global	Analysis Services	Synchronization Type	Don't synchronize automatically.
Global	Analytics Platform System	Maintenance Window Enabled	No
Global	Analytics Platform System	Maintenance Window Name	None
Global	APS Distributed Queries	Maximum Runtime Threshold	0.00:00:00

Alternatively, you can view this information directly at the individual object level. Select the desired object in the **Navigator** pane, and then open the **Settings** pane. The **Inherent From Parent** section of each setting group contains the name of the superior object that is passing down that group of settings. In the case where the object's setting has been defined with an **Object Group**, it lists that **Object Group** as the parent.