

SQL Sentry Condition Settings

Last Modified on 18 November 2021

Condition Settings define rules that a condition must meet for the condition to be fully satisfied. [Conditions](#) describe the various states of monitored objects within your environment. SQL Sentry defines [actions](#) to be taken when conditions are met. With advanced filtering, the **Condition Settings** tab provides complete control of when an action should be taken for a given condition.

Condition Settings define any number of criteria surrounding captured events. The criteria you define must be met for the condition to be satisfied. **Condition Settings** are accessed in the **Condition Settings** tab of any configured condition/action combination. Logical operators are used to form complex conditional groups, allowing you to define criteria across any number of columns.

When you configure a condition you also specify an action to take place when that condition is met. If an event doesn't meet the **Condition Settings** criteria, the condition isn't considered to be met, and the associated action won't be taken.

Configuring Condition Settings

Condition Settings are available for most [General Conditions](#) and all [Audit](#) and [Failsafe Conditions](#). **Condition Settings** are accessed from the [General Conditions pane](#) for configured condition/action combinations. The **Condition Settings** tab contains a **Visual Filter Editor** and a **Text Filter Editor**.

Conditions ✕

VM-SUP-AG1

General Conditions ▼

Global\Default Site\Sup AG's

Inherited

| Condition | Status | Action |
|--|---------|-------------|
| Reporting Services Report: Failure | Enabled | Send Email |
| Reporting Services Report: Runtime Threshold Max | Enabled | Send Email |
| Reporting Services Report: Runtime Threshold Min | Enabled | Send Email |
| SQL Server Agent Alert: Alert Fired | Enabled | Send Email |
| SQL Server Agent Job: Block | Enabled | Send Email |
| SQL Server Agent Job: Failure | Enabled | Execute SQL |

▼

Explicit

| Condition | Action | Behavior |
|--------------------------------|-------------|----------------------------|
| SQL Server: Top SQL: Completed | Execute SQL | Override Inherited Actions |

Delete Add

▼

Action Settings Condition Settings Ruleset Description Message

Condition Filter

New Condition Filter Save Cancel

Visual Text

Or +

And +

[Application Name] Equals ProductionApp ✕

Application Name greater than 10 ✕

And +

CPU p1, WebApp2, WebApp 3 ✕ ✕

Database Name greater than 35 ✕

Duration Statistics... ▶

End Time

Event Duration... ▶

Host Name

Host Process ID

Login Name

Message Text

Object Type

Reads

SPID

Start Time

Text Data

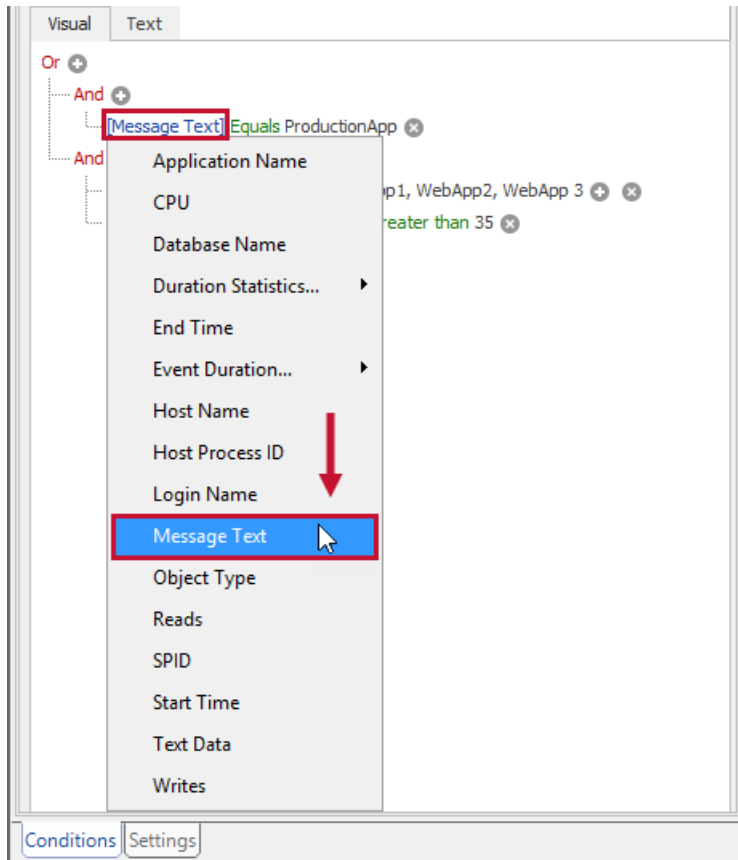
Writes

Conditions Settings

[🔗 Additional Information:](#) For more information about the **Text Filter Editor**, see the [Condition Filter](#) topic.

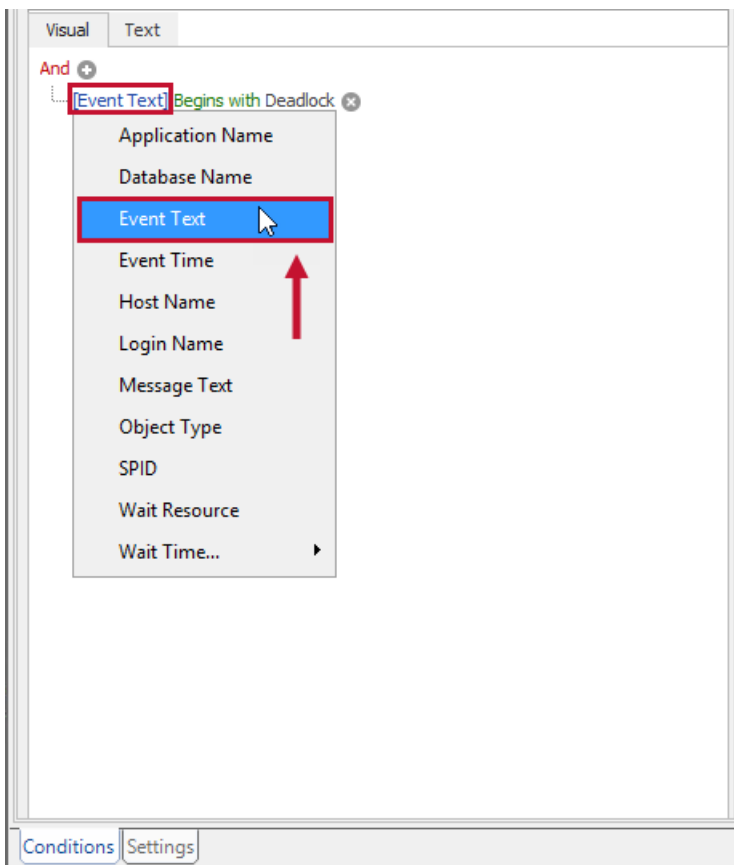
Message Text

Message Text is a useful column to configure a **Condition Settings** filter. **Message Text** corresponds to the entire body of an alert notification received for each respective condition. Any text in the body of alert emails is evaluated as part of the **Message Text** filter.



Event Text

The **Event Text** filter is available for select conditions. **Event Text** commonly corresponds to the raw output text of an event. For **Deadlock** events, this filters the captured raw XML of the **Deadlock**. For **Defragmentation Events**, the filter is applied to the last operation text that contains events like a successful offline rebuild or error information that happened during the operation.



Condition Settings Examples

The following section contains example walkthroughs for configuring different filters within the **Condition Settings** tab.

Completed Index Defragmentation Example

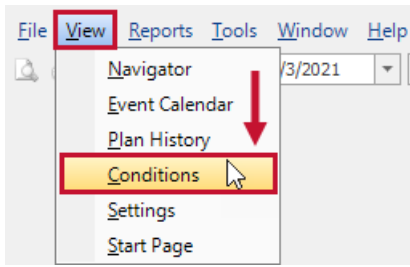
The **Fragmentation Manager** performs reorganization or rebuild operations based on fragmentation thresholds you specify. You may wish to be notified if the fragmentation level of an index is still above a certain threshold after a **Defragmentation Operation** has completed.

In this example, we configure an email alert for the **Index: Defragmentation Completed Condition** that fires after a **Defragmentation Operation** has taken place and the fragmentation level remains above 15 percent.

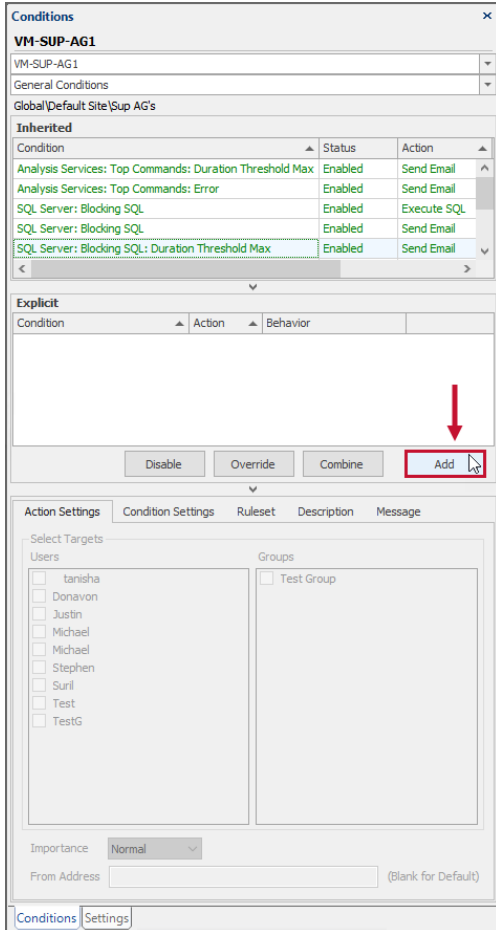
Note: There are several actions that can be taken in response to conditions being met. Choose to execute an Agent Job or a T-SQL command, either in conjunction with, or instead of sending an alert email.

Adding a Condition

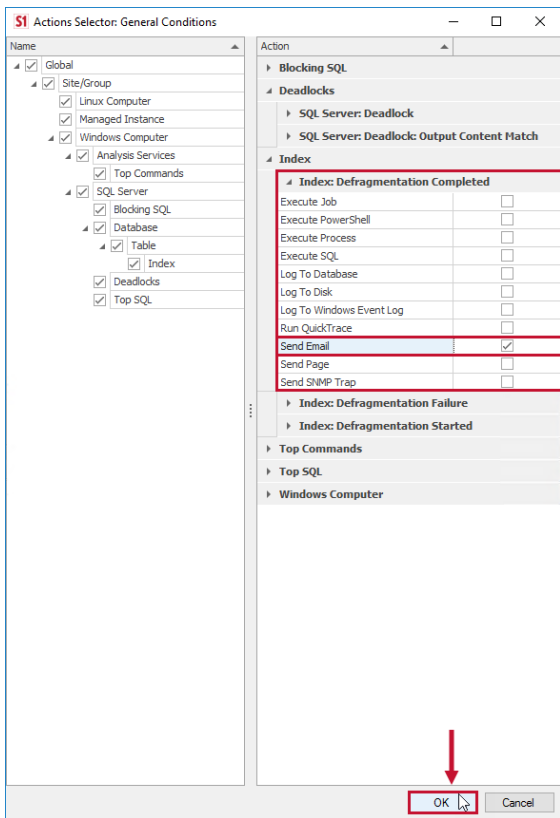
1. Select the **Shared Groups** node in the **Navigator** pane to set the action globally. Open the **General Conditions** pane (**View > Conditions**).



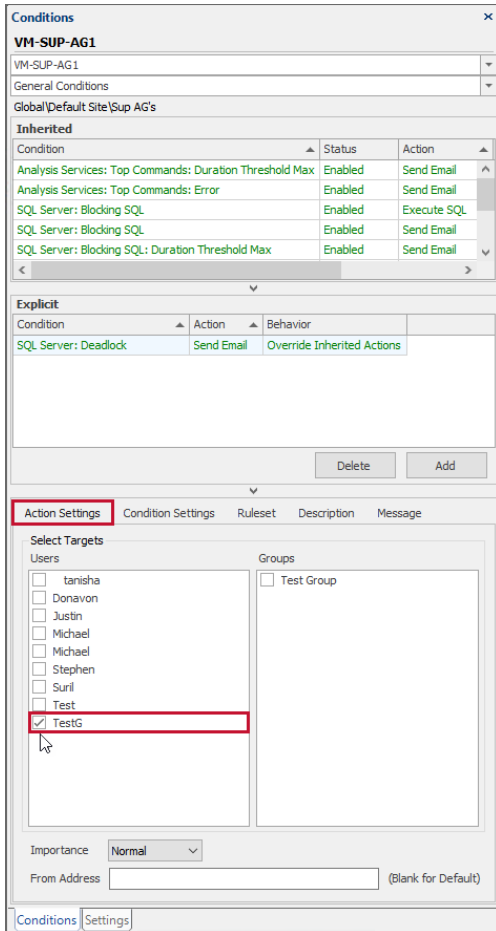
2. In the **Conditions** pane, select **Add** to open the **Actions Selector**.



3. Expand the **Index Group**, and then expand the **Index: Defragmentation Completed**. Select the checkbox next to the **Send Email**, and then select **OK**.



4. In the **Actions Settings** tab, select your desired user or group to target the **Send Email Action**.



The configured **Send Email Action** for the **Index: Defragmentation Completed Condition** is listed in the **Conditions** pane.

Adding a Condition Setting to the Condition

Note: Condition Setting options will vary depending on your Condition selection.

In this example, we will add a Condition Setting to an **Index: Defragmentation Completed Condition**. Add a Condition setting to **Index: Defragmentation Completed Condition** by completing the following steps:

1. Select your desired Condition, and then select the **Condition Settings** tab in the Conditions pane.

The screenshot shows the 'Conditions' pane for 'VM-SUP-AG1'. The 'Condition Settings' tab is selected, and the 'Index: Defragmentation Completed' condition is highlighted. The 'Condition Settings' tab is also highlighted, and a mouse cursor is pointing at it. The 'Condition Filter' is set to 'None'.

| Condition | Status | Action |
|---|---------|-------------|
| Analysis Services: Top Commands: Duration Threshold Max | Enabled | Send Email |
| Analysis Services: Top Commands: Error | Enabled | Send Email |
| SQL Server: Blocking SQL | Enabled | Execute SQL |

| Condition | Action | Behavior |
|----------------------------------|------------|----------------------------|
| Index: Defragmentation Completed | Send Email | Override Inherited Actions |
| Index: Defragmentation Failure | Send Email | Override Inherited Actions |
| SQL Server: Deadlock | Send Email | Override Inherited Actions |

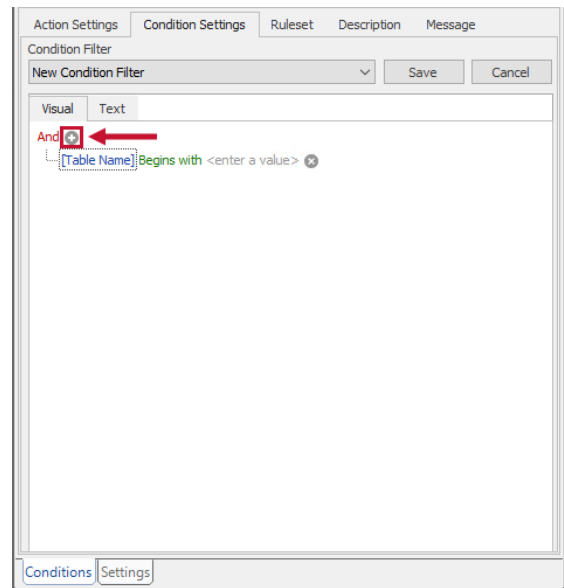
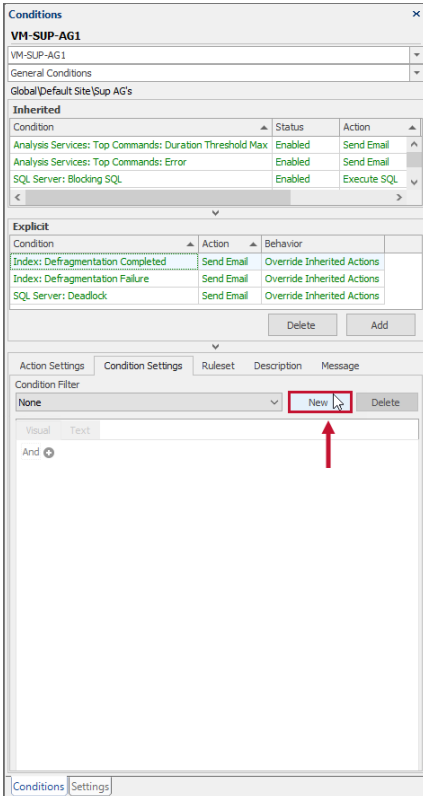
Condition Filter: None

Visual | Text

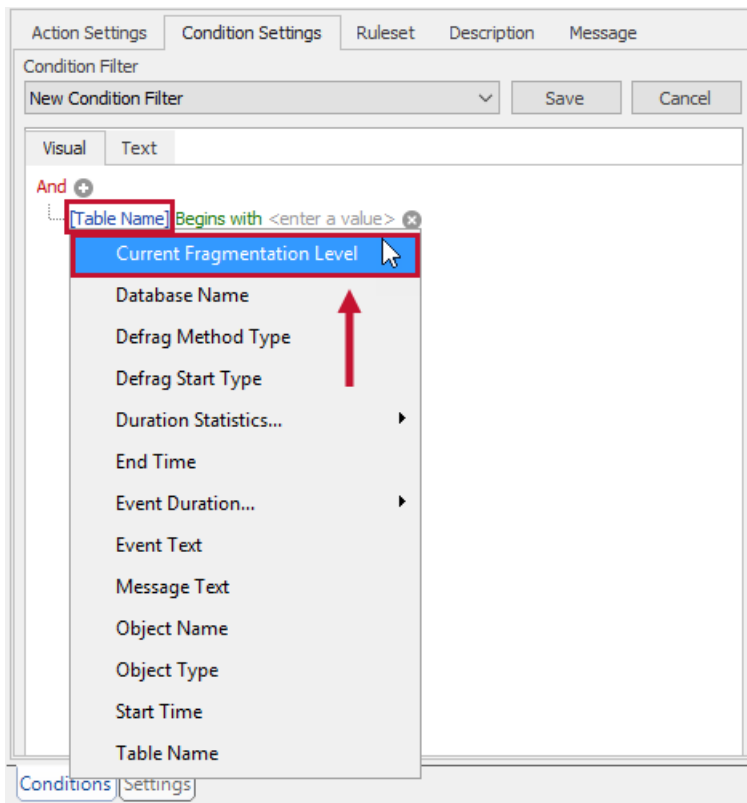
And +

Note: The Condition Filter is initially empty.

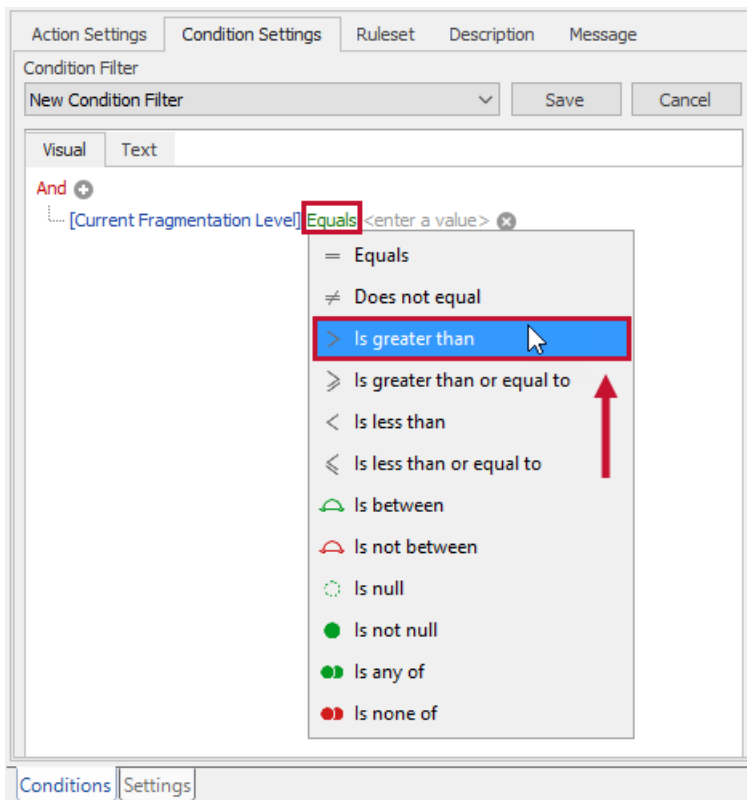
2. Select **New** to create a new Condition Filter. Select the plus symbol **+** next to the **And** logical operator to add a new Condition Filter.



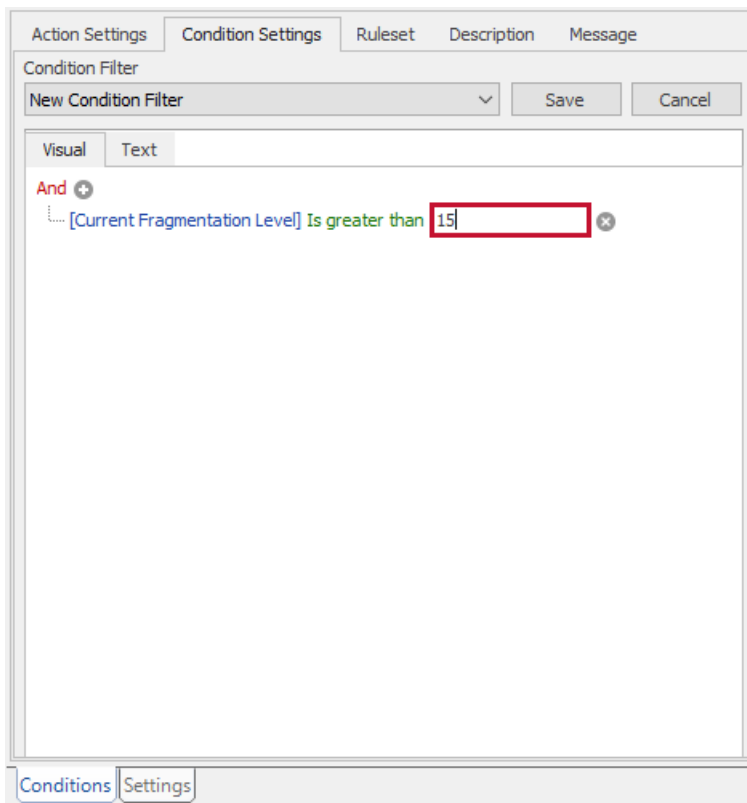
3. Select the Column name **[Table Name]** to open a drop-down list of available choices, and then choose **[Current Fragmentation Level]**.



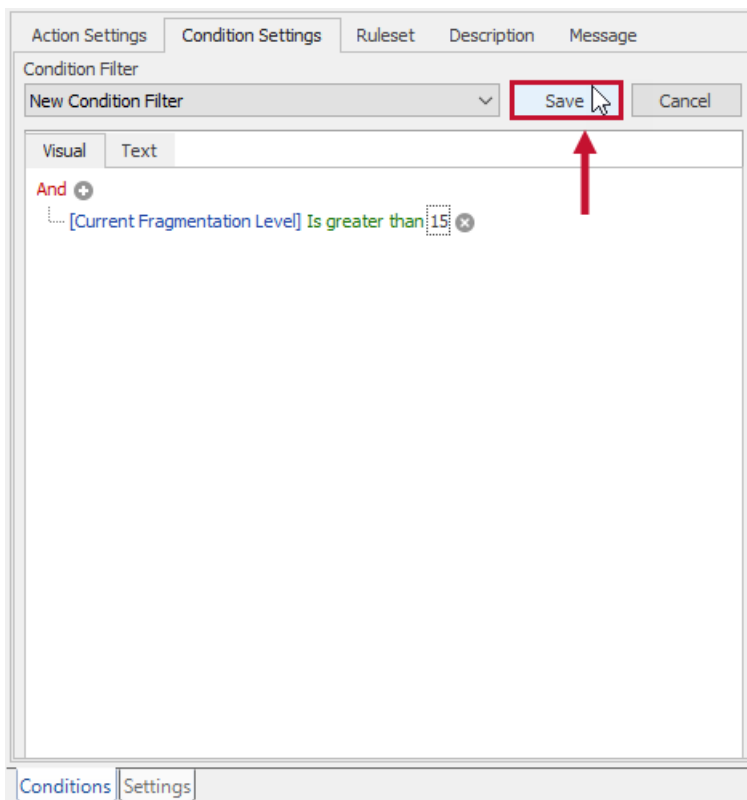
4. Select the **Equals** Criteria Operator to open the context menu, and then select **Is greater than**.



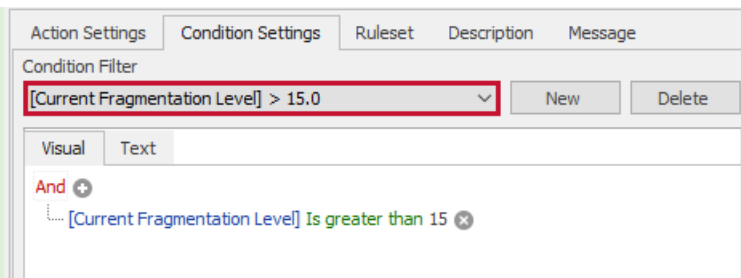
5. Select the field, and then enter your desired value. For this example, enter **15**.



6. Select **Save** to save your Condition Filter.



Success: The **Condition Setting** is now configured. If you select the **Condition Settings Text** tab, you should see the filter string defined as follows:



[Current Fragmentation Level.Value] > 15.0

With the **Condition Setting** defined, the **Index: Defragmentation Completed Condition** is met, and its associated **Send Email Action** fires for the following case:

A Defragmentation Operation has completed on an index, and the fragmentation level is above 15 percent.

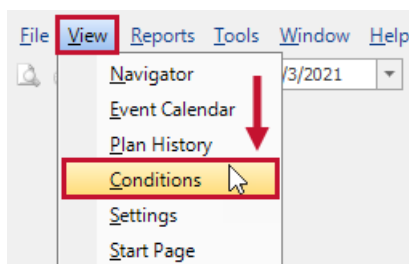
SQL Server Deadlock Example

SQL Sentry collects information about deadlocks happening within your environment. View this information in the **Deadlocks** tab of **Performance Analysis**.

Depending on your environment, you may not wish to be alerted about every deadlock that takes place. In this example, configure an alert for the **SQL Server: Deadlock Condition** and limit the notifications you receive to a specific database using **Condition Settings**.

Adding a Condition

1. Select the **Shared Groups** node in the **Navigator** pane to set the action globally. Open the **Conditions** pane (**View > Conditions**).



2. In the **Conditions** pane, select **Add** to open the **Actions Selector**.

Conditions ✕

VM-SUP-AG1

VM-SUP-AG1

General Conditions

Global\Default Site\Sup AG's

Inherited

| Condition | Status | Action |
|---|---------|-------------|
| Analysis Services: Top Commands: Duration Threshold Max | Enabled | Send Email |
| Analysis Services: Top Commands: Error | Enabled | Send Email |
| SQL Server: Blocking SQL | Enabled | Execute SQL |
| SQL Server: Blocking SQL | Enabled | Send Email |
| SQL Server: Blocking SQL: Duration Threshold Max | Enabled | Send Email |

Explicit

| Condition | Action | Behavior |
|-----------|--------|----------|
| | | |

Action Settings | Condition Settings | Ruleset | Description | Message

Select Targets

Users

- tanisha
- Donavon
- Justin
- Michael
- Michael
- Stephen
- Suril
- Test
- TestG

Groups

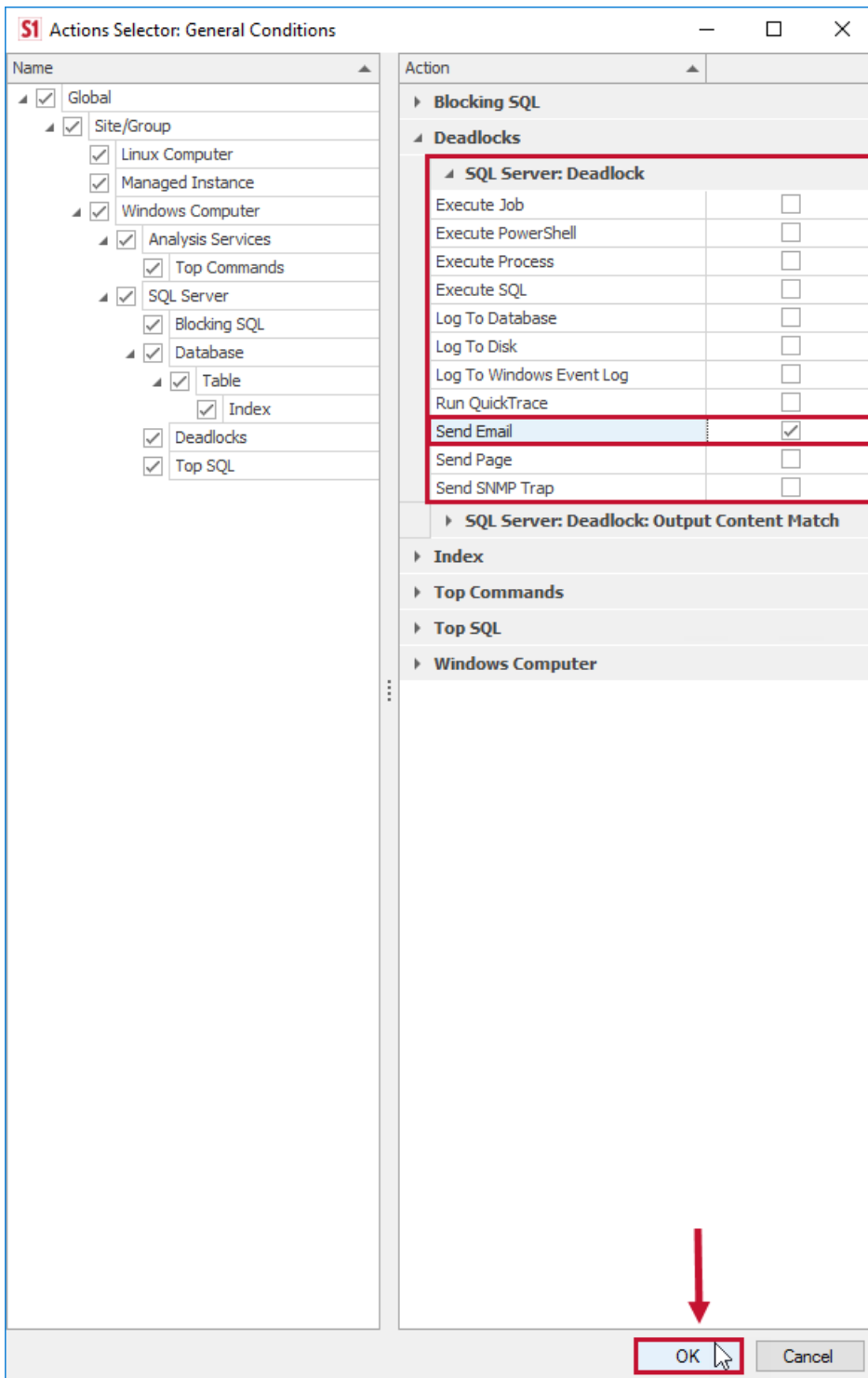
- Test Group

Importance

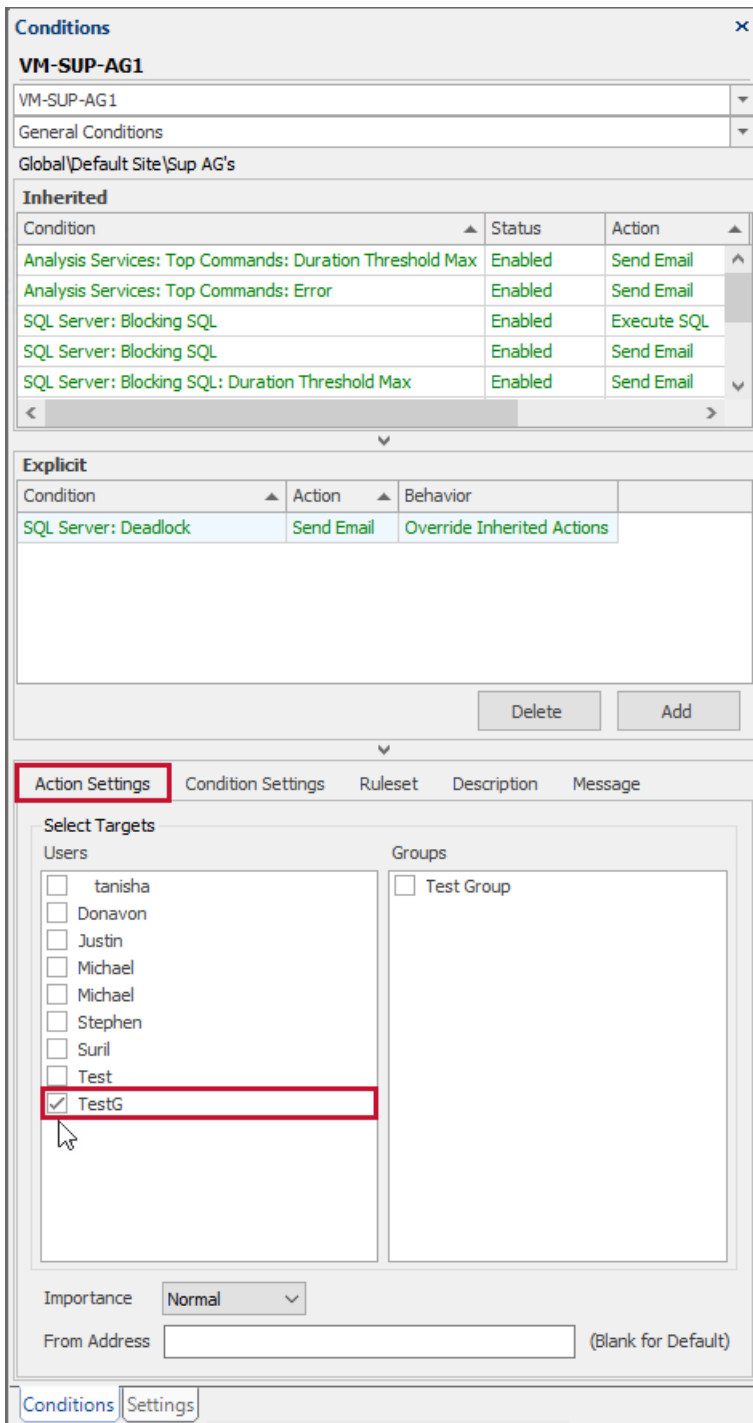
From Address (Blank for Default)

Conditions | Settings

3. Expand the **Deadlock** group, and then expand the **SQL Server: Deadlock Condition**. Select the checkbox next to the **Send Email**, and then select the **OK**.



4. In the **Actions Settings** tab, select your desired user or group to target the **Send Email Action**.



Adding a Condition Setting to the Condition

In this example, use the Condition Settings to limit the notifications you receive to a specific database for your **SQL Server: Deadlock Condition**. Add a Condition Settings to your **SQL Server: Deadlock Condition** by completing the following steps:

1. Select your desired Condition, and then select the **Condition Settings** tab in the Conditions pane.

Conditions ✕

VM-SUP-AG1

VM-SUP-AG1

General Conditions

Global\Default Site\Sup AG's

Inherited

| Condition | Status | Action |
|---|---------|-------------|
| Analysis Services: Top Commands: Duration Threshold Max | Enabled | Send Email |
| Analysis Services: Top Commands: Error | Enabled | Send Email |
| SQL Server: Blocking SQL | Enabled | Execute SQL |

Explicit

| Condition | Action | Behavior |
|----------------------------------|------------|----------------------------|
| Index: Defragmentation Completed | Send Email | Override Inherited Actions |
| SQL Server: Deadlock | Send Page | Override Inherited Actions |

Delete Add

Action Settings Condition Settings Ruleset Description Message

Condition Filter

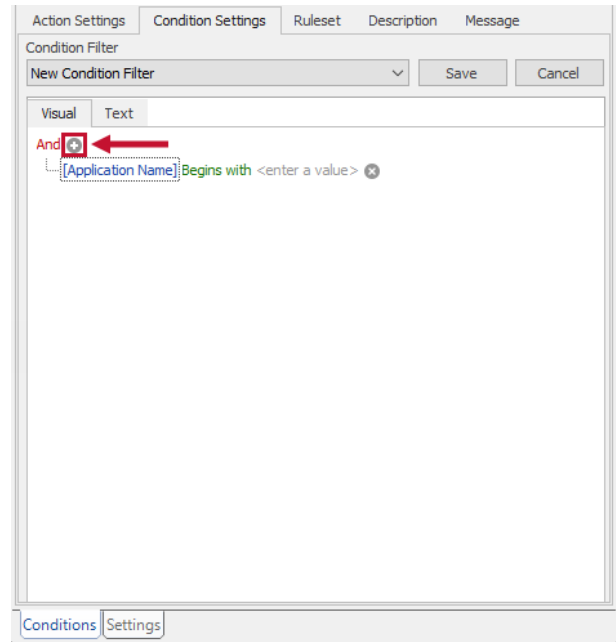
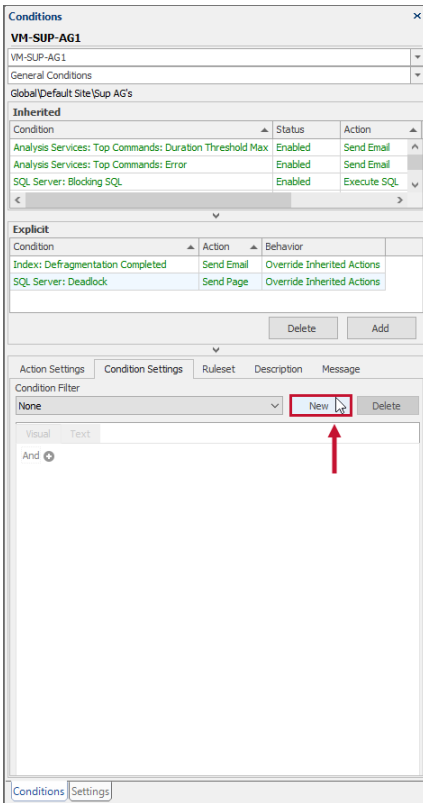
None New Delete

Visual Text

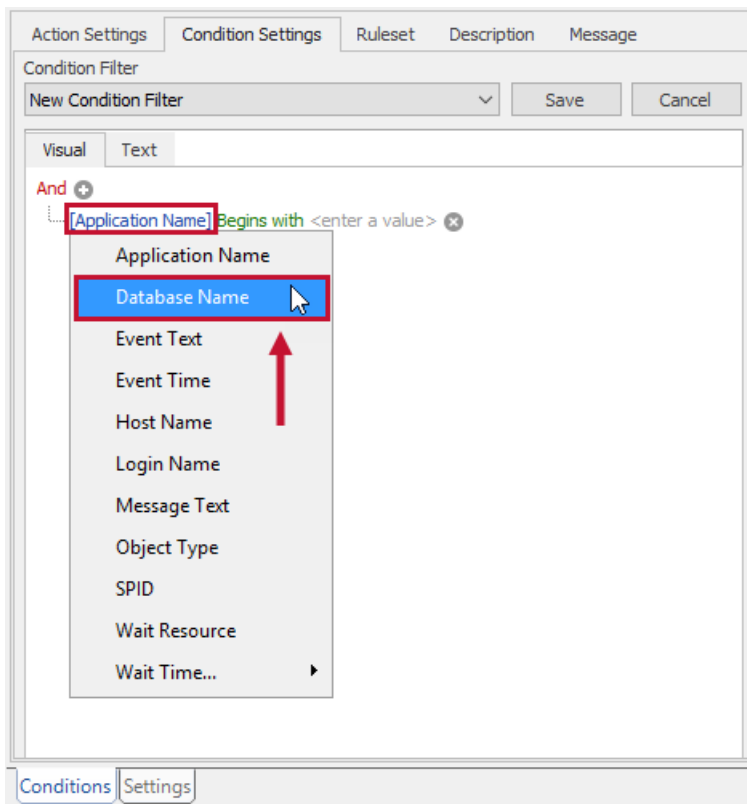
And +

Conditions Settings

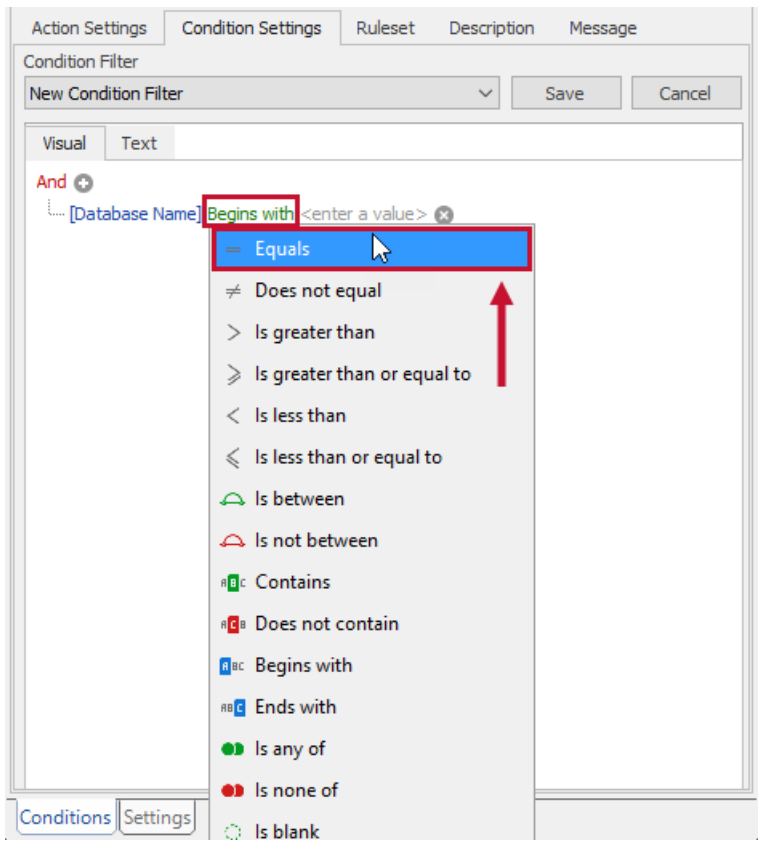
2. Select **New** to create a new Condition Filter. Select the plus symbol **+** next to the **And** logical operator to add a new Condition Filter.



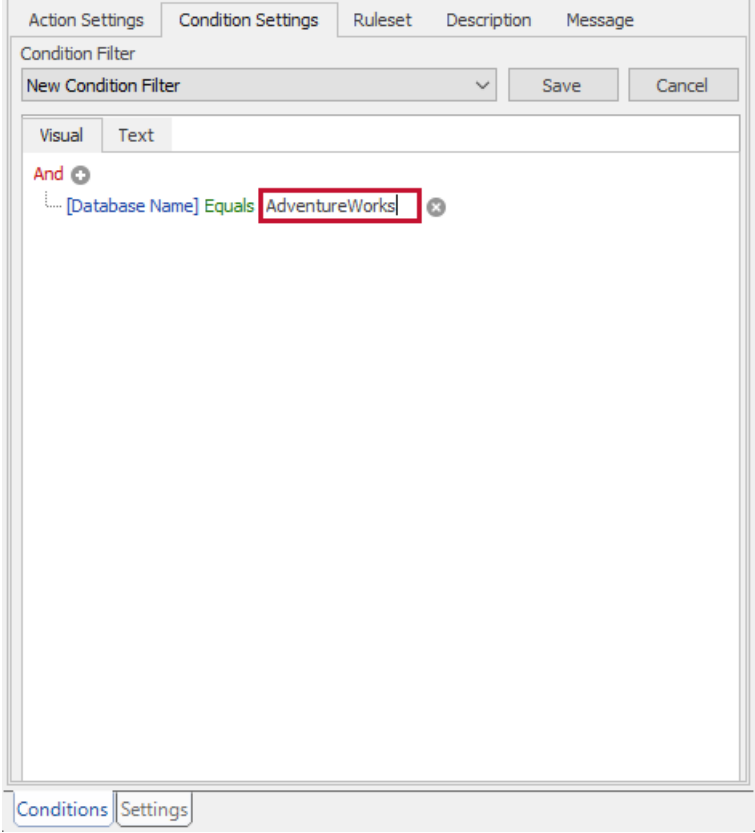
3. Select the Column name **[Application Name]** to open a drop-down list of available choices, and then choose **[Database Name]**.



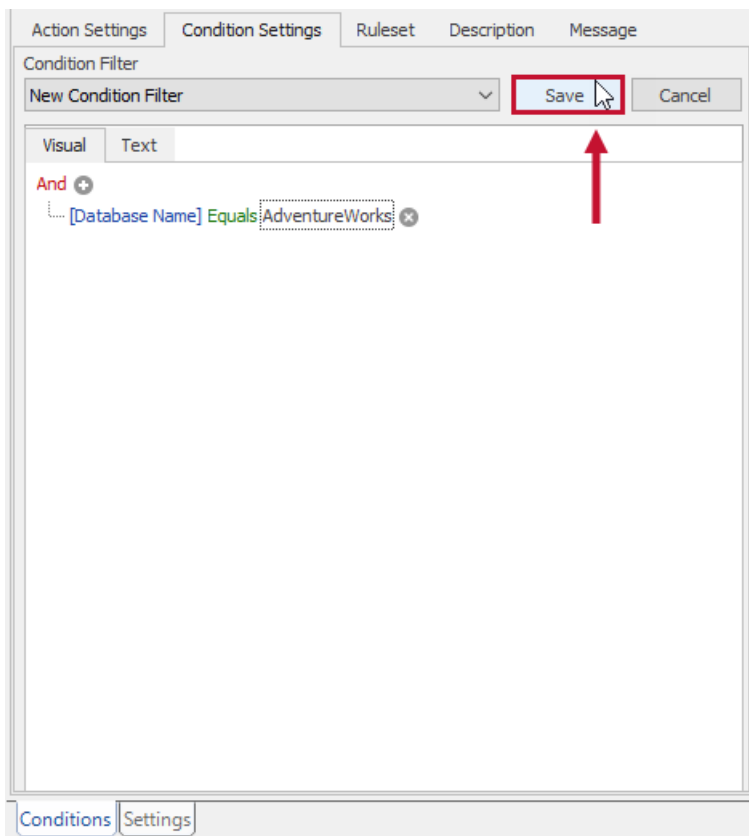
4. Select the **Equals** Criteria Operator to open the context menu, and then select **Equals**.



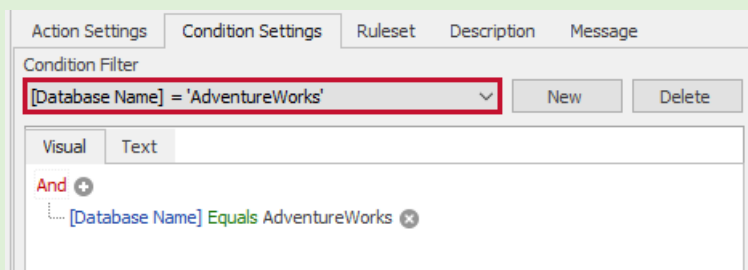
5. Select the field, and then enter your desired value. For this example, enter **AdventureWorks**.



6. Select **Save** to save your Condition Filter.



👏 **Success:** The **Condition Setting** is now configured. If you select the **Condition Settings Text** tab, you should see the filter string defined as follows:



[Database Name] = 'AdventureWorks'

With the **Condition Setting** defined previously, the **SQL Server: Deadlock Condition** is met, and its associated **Send Email Action** fires for the following case:

A Deadlock is detected and the victim database is the AdventureWorks.

📌 **Note:** String comparison is case insensitive.

📌 **Note:** As you become more comfortable with the **Condition Setting** process, you may wish to build

[Condition Filters](#) within the **Text** tab. If you make a mistake while doing so, the syntax error is underlined and the **Visual** tab becomes disabled until you fix the problem.