

Alarms & Events Plug-In

© 2011 Kepware Technologies

Table of Contents

Table of Contents	2
Alarms & Events Plug-In	3
Overview	3
AE Plug-In Terminology	3
AE Plug-In Conditions	3
The AE Plug-In from the AE Clients' Perspective	4
Real World Examples	5
User Interfaces	7
Plug-In Interface	7
Area Interface	7
Source Interface	8
Condition Interface	9
Data Types Description	15
Error Descriptions	16
The condition '<condition>' is not supported	16
The deadband range is invalid. The high value must be higher than the low value	16
Failed to add the condition '<condition>' to source '<source>'. The source already has this con- dition	16
The name '<name>' already exists	16
The name '<name>' is invalid	17
Please enter a valid name	17
Please enter a valid tag reference	17
The severity must range from '<value>' to '<value>'	17
References to array tags are not supported '<tag name >'	17
References to complex tags are not supported '<tag name>'	17
Tag '<tag name>' does not exist or you are trying to add a dynamic references which is not allowed	18
Index	19

Alarms & Events Plug-In

Help version 1.018

CONTENTS

[Overview](#)

What is the Alarms & Events Plug-In and how is it used?

[User Interfaces](#)

How can I use the different interfaces to configure alarms and sub-conditions?

[Data Types Description](#)

Which data types are supported?

[Error Descriptions](#)

What error messages does the Alarms & Events Plug-In produce?

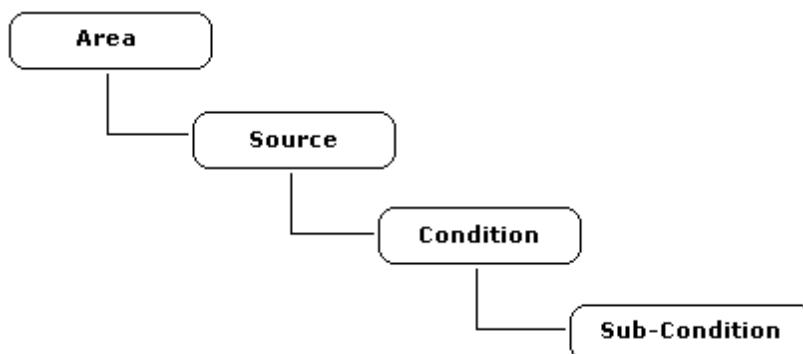
Overview

The Alarms & Events Plug-In provides custom alarm support to all devices connected to the OPC server, regardless of whether or not the device natively supports AE. It also allows users to configure alarms using device data. AE clients can then interact with the alarm: when the alarm conditions are met, the AE Plug-In will send an event to AE clients. Alarm support may be added to any device.

AE Plug-In Terminology

Plug-In Hierarchy

The Alarms & Events Plug-In is organized into areas, sources, conditions and sub-conditions.



Descriptions are as follows:

- **Area:** Areas are the root of the hierarchy and are used strictly for organizational purposes. They can be sections of a plant, sub-sections, parts of a machine and so forth. Areas may contain other areas in addition to sources.
- **Source:** Sources belong to an area and are linked to an OPC server tag. They receive data updates from an OPC server tag and pass the update to conditions for processing.
- **Condition:** Conditions are synonymous with alarms and are used to generate events. They take updates from the source and feed them to sub-conditions in order to determine the state of the condition. Changes in the condition's state will produce events, which are then sent to AE clients.
- **Sub-Condition:** Sub-conditions compare data updates from the source to a trigger. They represent the state of a condition and hold the comparison logic that controls whether the sub-condition is on or off. A sub-condition is on when its trigger criteria is satisfied.

Events

The AE Plug-In generates **Conditional** events. A conditional event represents the state of a condition and transitions between sub-conditions. Conditional events are produced by the hierarchy discussed above.

AE Plug-In Conditions

The AE Plug-In supports nine conditions that are separated into three categories: Multi Level, Rate of Change and Deviation. Each condition has a unique name and set of sub-conditions.

MULTI_LEVEL

MULTI_LEVEL conditions support multiple sub-conditions, which include HI_HI, LO_LO, HI, and LO. This condition should be used if the source has multiple states of interest (and if users want to know when the condition transitions between states). For example, this condition should be used when a temperature tag with multiple temperatures of interest is being used.

The sub-conditions' priority is as follows (from highest to lowest): HI_HI, LO_LO, HI, and LO. The priority is not configurable. It controls the active state of the condition if multiple sub-conditions are true. For example, if both HI_HI and HI are true, the condition is in the HI_HI state because HI_HI has a higher priority.

HI_HI, HI, LO, LO_LO

These single level conditions contain a sub-condition that matches the condition name. These conditions should be used if there is a single state of interest for the source. For example, this condition should be used if a temperature tag with a single temperature of interest is being used. Because the conditions are the same functionally, utilize one with a name that makes the most sense for the application.

ROC_HI, ROC_LO

Rate of Change (ROC) conditions have a single sub-condition that matches the condition name. They are used to compare the source to a static or dynamic ROC. For example, this condition should be used if using a source tag that represents production output (and when users want to trigger the condition if the output falls below 100 units a minute). Because the conditions are the same functionally, utilize one with a name that makes the most sense for the application.

DEV_HI, DEV_LO

Deviation conditions have a single sub-condition that matches the condition name. They are used to monitor the deviation of the source and trigger the condition if the source exceeds the limits set by the deviation. For example, this condition should be used if using a source that monitors power consumption (and when users want to trigger the condition if the power consumption is outside of 100W \pm 20%). Because the conditions are the same functionally, utilize one with a name that makes the most sense for the application.

Note: For examples on using the conditions, refer to [Real World Examples](#).

The AE Plug-In from the AE Clients' Perspective

Filtering Options

Filtering allows AE clients to focus on specific areas, sources or event types. The Alarms & Events Plug-In supports client filtering on the following event properties:

- Area
- Source
- Severity
- Event Type (Simple and Conditional)
- Categories

Filtering Categories

The Alarms & Events Plug-In also supports filtering categories, which differ by event type. The supported filtering categories are as follows:

Simple Events

- **Server Information Events:** These are the same information events that appear in the OPC server event view.
- **Server Warning Events:** These are the same warning events that appear in the OPC server event view.
- **Server Error Events:** These are the same error events that appear in the OPC server event view.

Conditional Events

- **Level Conditions:** Events generated by Level conditions (MULTI_LEVEL, HI_HI and so forth).
- **Rate of Change Conditions:** Events generated by ROC conditions (ROC).
- **Deviation Conditions:** Events generated by Deviation conditions (DEV).

Real World Examples

The examples below are based on the following hypothetical situation.

An industrial facility uses an OPC AE client to monitor process alarms. Plant engineers would like to include temperature events from a new boiler. The boiler's temperature is measured with a Modbus-enabled temperature sensor; however, the sensor does not support alarms. The Alarms & Events Plug-In may be used to provide alarm support to the sensor in the following manner: the OPC server (and Modbus driver) polls the sensor for the current temperature in degrees Celsius and maintains a "TEMP" tag for the value. The Alarms & Events Plug-In creates an alarm for that tag. The alarm is configured to send events to the OPC AE clients.

Note: In all examples, the Alarms & Events Plug-In has a single 'plant' area with a single TEMP source. Conditions are added to the TEMP source.

Example 1: MULTI_LEVEL

Users are interested in the temperature exceeding the following limits: greater than 150° C and 175° C, and less than 125° C and 100° C. These limits are grouped by severity. 150° C and 125° C are warnings. 175° C and 100° C are cause for evacuation. The MULTI_LEVEL condition will be used along with sub-conditions configured as follows:

- HI_HI
Threshold: 175
Comparison: Source > Threshold
- HI
Threshold: 150
Comparison: Source > Threshold
- LO
Threshold: 125
Comparison: Source < Threshold
- LO_LO
Threshold: 100
Comparison: Source < Threshold

The OPC server's TEMP tag is currently 145° C. The condition is inactive. The next update is 196° C. The MULTI_LEVEL condition transitions into the HI_HI state and generates an AE event that is sent to the AE client. Although both the HI_HI and HI sub-conditions are true, HI_HI has a higher priority and so the condition takes on the HI state. The next update is 95° C. The MULTI_LEVEL condition transitions out of the HI_HI state and into the LO_LO state. Although both the LO_LO and LO sub-conditions are true, the condition takes on the LO_LO state because it has a higher priority. The condition sends a state change event to the AE client.

Example 2: HI_HI, HI, LO, LO_LO

Users are interested in temperatures exceeding 225° C. A single HI_HI condition will be used with a HI_HI sub-condition configured as follows:

- HI_HI
Threshold: 225
Comparison: Source > Threshold

The OPC server's TEMP tag is 100° C. The condition is inactive. The next update is 226° C. The HI_HI condition becomes active and sends an event to the AE client.

Example 3: ROC_HI, ROC_LO

Users are interested in a maximum rate of change in the boiler's temperature. They do not want the temperature to change by more than 40° C/minute. If it does, the AE client must be notified. The ROC_HI condition will be used with a single ROC_HI sub-condition configured as follows:

- ROC_HI
Threshold: 40
Interval: Minute
Comparison: Source > Threshold

The OPC Server TEMP tag updates at a rate of once a minute. The first update is 100° C. The condition is inactive because this is the first update. When the value changes to 120° C a minute later, it is an absolute change of 20° C/minute. The condition is still inactive. When the value changes to 70° C a minute later, it is an absolute change of 50° C/minute. The condition then becomes active and sends an event to the AE client.

Example 4: DEV_HI, DEV_LO

Users are interested in keeping the boiler temperature within a maximum/minimum limit. It should be between 100° C and 200° C. Although this may be done with MULTI_LEVEL or single level conditions, it is better suited for a deviation condition. Thus, the DEV_HI condition will be used with a single DEV_HI sub-condition configured as follows:

- DEV_HI
Value: 150
+/-: 50
Type: Absolute

The OPC Server TEMP tag begins with a value of 100° C, which is within the specified range. The condition is inactive. The next update is 90° C. The condition then becomes active and sends an event to the AE client.

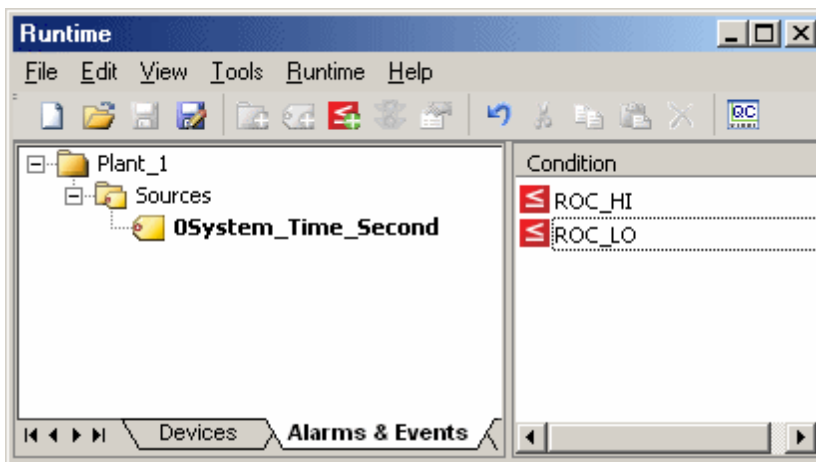
User Interfaces

For more information on a specific user interface, select a link from the list below.

[Plug-In Interface](#)
[Area Interface](#)
[Source Interface](#)
[Condition Interface](#)

Plug-In Interface

The Alarms & Events Plug-In interface consists of a Project view and a Detail view, located on the left and right panes respectively. Its toolbar contains all of the options available to the AE objects (such as area, source and so forth). These options may also be accessed in both views by right-clicking on the AE objects.



Note: Drag and drop can be used to move objects. To copy objects, press the Control key during drag and drop.

Project View

The Project view displays the Alarms & Events server's hierarchy, which consists of Areas, Source Folders and Sources. Descriptions are as follows:

1. **Area:** Areas are the root of the Alarms & Events hierarchy and may contain sources in addition to other areas. The area's name must be unique among sibling areas. Users can create a new area, edit an existing area, toggle, copy, cut, paste, delete and drag and drop an area in the Project view to another area.

Note: In the image above, the area is labeled "Plant1."

2. **Source Folder:** Source Folders are used to organize all the sources that belong to an area. Users can neither create nor rename source folders.

Note: In the image above, the source folder is labeled "Sources."

3. **Source:** Sources belong to the source folder. The source name must be unique among sibling sources. Users can create a new source, edit an existing source, toggle, copy, cut, paste and drag and drop a source to an area or source folder. Only one source folder is allowed per area.

Note: In the image above, the source is labeled "0System_Time_Second."

Detail View

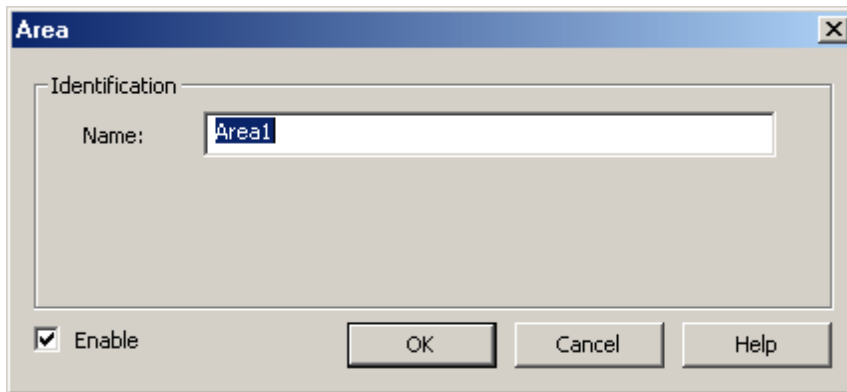
The Detail view displays the conditions that belong to the source selected in the Project view. Users can create a new condition, edit an existing condition, toggle, copy, cut, paste, delete and drag and drop a condition to a source in the Project view.

Toolbar

The Toolbar provides convenient access to all of the AE object functions (such as adding objects, editing objects, toggling objects, cutting, copying, pasting and undoing events).

Area Interface

Areas organize sources and conditions in the AE Plug-In.

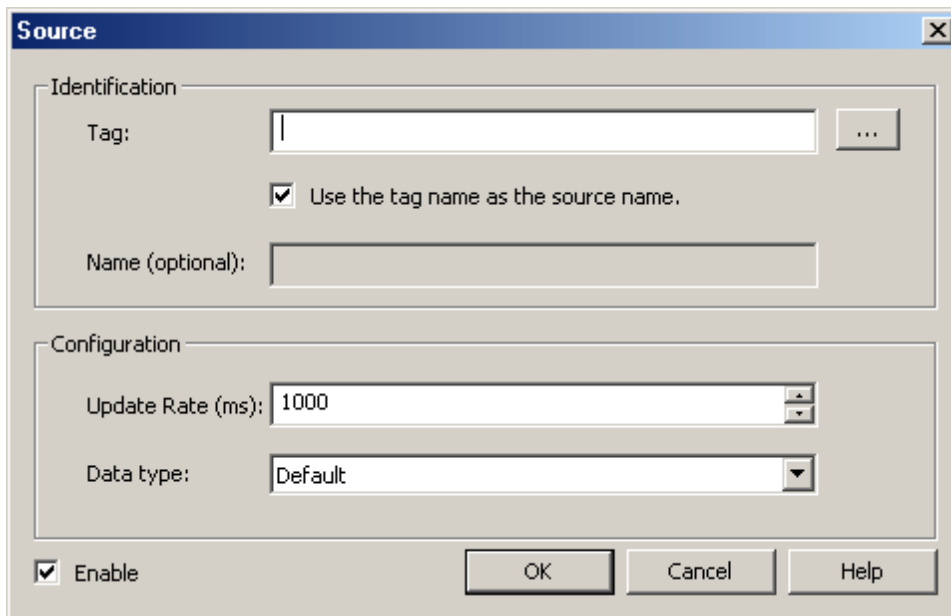
The 'Area' dialog box has a title bar with a close button. It contains an 'Identification' section with a 'Name' text box containing 'Area1'. At the bottom, there is an 'Enable' checkbox which is checked, and three buttons: 'OK', 'Cancel', and 'Help'.

Description is as follows:

- **Name:** The Area's name must be unique. The maximum length of characters is 1024.

Source Interface

Each source is linked to an OPC server tag.

The 'Source' dialog box has a title bar with a close button. It contains two sections: 'Identification' and 'Configuration'. In the 'Identification' section, there is a 'Tag' text box, a 'Use the tag name as the source name' checkbox which is checked, and a 'Name (optional)' text box. In the 'Configuration' section, there is an 'Update Rate (ms)' spinner box set to 1000 and a 'Data type' dropdown menu set to 'Default'. At the bottom, there is an 'Enable' checkbox which is checked, and three buttons: 'OK', 'Cancel', and 'Help'.

Descriptions are as follows:

1. **Name:** This parameter specifies the name of the source, which must be unique among sibling sources. It is recommended that the source name be the same as the tag to which it is linked. If the **Use tag name as the source name** parameter is checked, this parameter will be greyed out.
2. **Tag:** This parameter specifies the address of the OPC server tag that is linked to the source.
3. **Tag Browser:** When clicked, this button (...) launches the server Tag Browser. Once a tag is selected, its address will be placed in the Tag parameter.
4. **Use the tag name as the source name:** This checkbox controls whether or not the specified Name parameter will be used. When checked, the Name edit control will be greyed out and the source will be named after the tag address. When unchecked, the user must supply a valid source name.

Note: When checked, the tag address will be altered in the following manner in order to create a valid source name.

- a. Leading underscores will be changed to 0.
- b. Periods will be converted to '_'.
- c. Double underscores will be converted to single underscores.

Example: `_Channel1.Device1.Tag1` changes to `0Channel1_Device1_Tag1`.

5. **Update Rate:** This parameter specifies the fastest rate at which the source gets data updates from the OPC server tag. The range is 100-60000 ms. The default setting is 1000 ms.
6. **Data Type:** This parameter specifies the data type of the selected tag. For more information, refer to [Data Types Description](#).
7. **Enable:** Sources are enabled or disabled. When enabled, the source allows child conditions to produce Conditional events. When disabled, the source stops child conditions from producing Conditional events. If a source is enabled/disabled, all child conditions will be enabled/disabled as well.

Condition Interface

Conditions organize a number of sub-conditions that hold the condition's state information. A condition's sub-conditions may be configured when a condition is created or edited. The Condition Wizard is used to create conditions; the condition's Properties is used to edit them. When both creating and editing conditions, there are three sequential dialogs: Condition, Sub-Condition General and Sub-Condition Triggers.

Condition

Descriptions are as follows:

1. **Condition:** This parameter specifies the type of condition being used. It is only configurable during the creation of a new condition. The Alarms & Events Plug-In defines a number of conditions available to the user, of which one can be selected. The condition type selected will affect the type of sub-conditions made available.

Note: A source can have a maximum of one condition type; that is, one MULTI_LEVEL condition. Duplicate condition types per source are not allowed. For a list of available conditions, refer to [AE Plug-In Conditions](#).

2. **Enable:** This parameter turns the condition on or off. When checked, the condition is on and can produce Conditional events. When unchecked, the condition is off and cannot produce events.
3. **Deadband Type:** This parameter specifies the deadband type. There are three options: None, Absolute and Percent. The default setting is None. Descriptions are as follows:
 - b. **None:** No deadband is used.
 - c. **Absolute:** If the difference between two consecutive data updates from the source is more than the amount specified in the Value field, the update is valid and can produce events. Otherwise, the update is ignored.
 - d. **Percent:** This option is similar to Absolute except that the Value field is a percentage of the specified range. For example, if the Value is 10 (10%) and the expected range is 0 to 10, the Absolute deadband would be +/- 1.
4. **Value:** This parameter specifies the deadband value. Any positive number is supported. If the Deadband Type is set to None, this parameter will be greyed out.
5. **Range:** This parameter specifies the upper and lower range of the percent deadband. All real numbers are supported, but the first number must be less than the second. This parameter is only accessible when the Deadband Type is set to Percent.

Sub-Condition General

The Sub-Condition General dialog contains common settings such as messages, descriptions and severity.

The screenshot shows the 'Sub-Condition General' dialog box. On the left, under 'Sub-Conditions', there is a list box containing 'HI HI'. On the right, under 'General', there is a 'Description:' text field, a 'Severity:' spinner box set to '1', and two checkboxes: 'Enable' (checked) and 'Auto-Acknowledge' (unchecked). Below the 'General' section is a 'Messages' section with three text fields: 'In-to:', 'Out-of:', and 'Ack:'. At the bottom of the dialog are four buttons: '< Back', 'Next >', 'Cancel', and 'Help'.

Descriptions are as follows:

1. **In-To:** The string sent to AE clients when the condition transitions in-to the selected sub-condition. The maximum length is 1024.
2. **Out-Of:** The string sent to AE clients when the condition transitions out-of the selected sub-condition. The maximum length is 1024.

3. **Ack:** The string sent to AE clients when the condition's transition into this sub-condition is acknowledged. The maximum length is 1024.
4. **Description:** The description of what the sub-condition is for. This description is sent to AE clients when they query for the state of a condition. The maximum length is 1024.
5. **Severity:** The user-defined severity of the sub-condition. The higher the number, the more severe. The range is from 1-1000. The default setting is 1. The value is used by some AE clients to interpret the threat presented by the alarm. It does not affect the fixed priority of sub-conditions defined by the Alarms & Events Plug-In.
6. **Enable:** This parameter enables/disables the sub-condition. If a sub-condition is disabled, it cannot become active.
7. **Auto-Acknowledge:** This parameter automatically acknowledges the transition into this sub-condition. Sub-condition transitions can be acknowledged by the server or by AE clients.

Sub-Condition Triggers

The Sub-Condition Triggers dialog consists of a list of sub-conditions supported by the parent condition along with settings for each sub-condition. There are three types of sub-conditions: Value, Rate of Change and Deviation. Each sub-condition has unique settings.

Value

Value sub-conditions compare the source to a threshold. The threshold may be either a static value or a dynamic value (such as an OPC server tag). If the criteria defined by the comparison is met, the sub-condition will become active.

1. **Threshold:** A static value or tag to which the data updates from the parent source are compared. If the threshold is a tag, any OPC server tag is available aside from array tags and complex tags.
2. **Value/Tag:** This parameter specifies what a threshold is. The default setting is Value.

3. **Comparison:** This parameter is a comparison between the source and the threshold. The default setting is $>$. Options include the following:

Source $>$ Threshold: The source tag value is greater than the threshold tag value.

Source \geq Threshold: The source tag value is greater than or equal to the threshold tag value.

Source $<$ Threshold: The source tag value is less than the threshold tag value.

Source \leq Threshold: The source tag value is less than or equal to the threshold tag value.

Source $=$ Threshold: The source tag value is equal to the threshold tag value.

Source \neq Threshold: The source tag value is not equal to the threshold tag value.

4. **Data Type:** This parameter specifies the data type of the threshold. The default setting is Default. For more information, refer to [Data Types Description](#).

Important: The parameters described below only apply to tag thresholds.

1. **Update Rate:** This parameter specifies the tag's update rate. This is the fastest rate at which updates are received from the tag.
2. **Deadband Type:** This parameter specifies the tag's deadband type. Options include None, Absolute and Percent. The default setting is None. Descriptions are as follows:
 - a. **None:** No deadband is used.
 - b. **Absolute:** If the difference between two consecutive tag data updates is more than the amount specified in the Value field, the update is valid and can produce events. Otherwise, the update is ignored.
 - c. **Percent:** This option is similar to Absolute except that the Value field is a percentage of the specified range. For example, if the Value is 10 (10%) and the expected range is 0 to 10, the Absolute deadband would be ± 1 .
3. **Deadband:** This parameter is only visible when the Deadband Type is not set to None. Any positive real numbers are supported.
4. **Range:** This parameter is only visible when the Deadband Type is set to Percent. Any real number is supported. The lower range must be less than or equal to the higher range.

Rate of Change

The Rate of Change sub-condition compares the source to a static or dynamic rate of change. If the criteria defined by the comparison is met, the sub-condition becomes active.

1. **Threshold:** The source of the ROC. This field defines the threshold ROC for the comparisons. It is a value or an OPC server tag address. If the threshold is a value, it represents a rate of change in units/time. If the threshold is a tag, it is the ROC of the OPC server tag. All OPC server tags are available except complex and array tags.
2. **Value/Tag:** This parameter specifies the interpretation of the Threshold as either a value or tag. The default setting is Value.
3. **Interval || Update Rate (ms):** If the Threshold is a value, this field contains a list box of allowed intervals. The options are ms, sec and min. The default setting is ms. If the Threshold is a tag, this field will hold the update rate for the tag in milliseconds. The range is 100-60000 ms. The default setting is 1000 ms.
4. **Comparison:** This parameter specifies the comparison type between the ROC of the Threshold and the ROC of the Source. The default setting is >. Options include the following:
 - Source > Threshold:** The source tag value is greater than the threshold tag value.
 - Source >= Threshold:** The source tag value is greater than or equal to the threshold tag value.
 - Source < Threshold:** The source tag value is less than the threshold tag value.
 - Source <= Threshold:** The source tag value is less than or equal to the threshold tag value.
 - Source == Threshold:** The source tag value is equal to the threshold tag value.
 - Source != Threshold:** The source tag value is not equal to the threshold tag value.
5. **Data Type:** The data type of the tag. The default setting is Default.

Important: The parameters described below only apply to tags.

1. **Deadband Type:** This parameter specifies the tag's deadband type. Options include None, Absolute and Percent. The default setting is None. Descriptions are as follows:

- a. **None:** No deadband is used.
 - b. **Absolute:** If the difference between two consecutive tag data updates is more than the amount specified in the Value field, the update is valid and can produce events. Otherwise, the update is ignored.
 - c. **Percent:** This option is similar to Absolute except that the Value field is a percentage of the specified range. For example, if the Value is 10 (10%) and the expected range is 0 to 10, the Absolute deadband would be +/- 1.
2. **Deadband:** This parameter is only visible when the deadband type is not set to None. Any positive real numbers are supported.
 3. **Range:** This parameter is only visible if the Deadband Type is set to Percent. Any real number is supported. The lower range must be less than or equal to the higher range.

Deviation

The Deviation sub-condition compares the source to a static min/max range. If the source exceeds the range, the sub-condition becomes active.

The screenshot shows a software window titled "Sub-Condition Triggers". On the left, there is a list box labeled "Sub-Conditions" containing the entry "DEV_HI". On the right, there is a section labeled "Deviation" with two input fields: "Value" containing the number "10" and a field for the deviation amount containing the number "5", separated by a "+/-" symbol. Below these is a "Type" dropdown menu currently set to "Absolute". At the bottom of the window are four buttons: "< Back", "Finish", "Cancel", and "Help".

1. **Value:** The fixed point on which the deviation is based. Any real number is supported.
- Note:** The source must exceed the range for the sub-condition to be active. For example, if the max deviation is 50, the source must exceed 50.
2. **Type:** This parameter specifies how to interpret the '+/-' value. Options include Absolute and Percent. The default setting is Absolute.

Data Types Description

The Alarms & Events Plug-In supports the data types listed in the table below.

Data Type	Description
Default	The OPC server tag's data type.
Boolean	Single bit
Char	Signed 8 bit value bit 0 is the low bit bit 6 is the high bit bit 7 is the sign bit
Byte	Unsigned 8 bit value bit 0 is the low bit bit 7 is the high bit
Short	Signed 16 bit value bit 0 is the low bit bit 14 is the high bit bit 15 is the sign bit
Word	Unsigned 16 bit value bit 0 is the low bit bit 15 is the high bit
Long	Signed 32 bit value bit 0 is the low bit bit 30 is the high bit bit 31 is the sign bit
DWord	Unsigned 32 bit value bit 0 is the low bit bit 31 is the high bit
Float	32 bit floating point value bit 0 is the low bit bit 31 is the high bit
Double	64 bit floating point value bit 0 is the low bit bit 63 is the high bit
String	Typically null terminated, null padded or blank padded ASCII string.
BCD	Two byte packed BCD Value range is 0-9999. Behavior is undefined for values beyond this range.
LBCD	Four byte packed BCD Value range is 0-99999999. Behavior is undefined for values beyond this range.
Date	64 bit floating point value.

Error Descriptions

The following error/warning messages may be generated. Click on the link for a description of the message.

[The condition '<condition>' is not supported](#)

[The deadband range is invalid. The high value must be higher than or equal to the low value](#)

[Failed to add the condition '<condition>' to source '<source>'. The source already has this condition](#)

[The name '<name>' already exists](#)

[The name '<name>' is invalid](#)

[Please enter a valid name](#)

[Please enter a valid tag reference](#)

[The severity must range from '<value>' to '<value>'](#)

[References to array tags are not supported '<tag name>'](#)

[References to complex tags are not supported '<tag name>'](#)

[Tag '<tag name>' does not exist or you are trying to add a dynamic references which is not allowed](#)

The condition '<condition>' is not supported

Error Type:

Error

Possible Cause:

The project was altered to include a condition that is not supported by the AE Plug-In.

Solution:

Check the project file (*.xml) and then change the condition name to something valid.

The deadband range is invalid. The high value must be higher than the low value

Error Type:

Error

Possible Cause:

The deadband range in the condition or sub-condition dialog is invalid.

Solution:

Make sure that the range goes from high to low.

Failed to add the condition '<condition>' to source '<source>'. The source already has this condition

Error Type:

Error

Possible Cause:

An attempt was made to add a condition 'X' to a source that already contains a condition 'X'.

Solution:

A source cannot contain two conditions of the same type.

The name '<name>' already exists

Error Type:

Error

Possible Cause:

An attempt was made to add an area, condition or source with a duplicate name.

Solution:

Duplicate names are not allowed among sibling objects. Assign the object a unique name.

The name '<name>' is invalid

Error Type:

Error

Possible Cause:

The name of an area or source is invalid.

Solution:

Duplicate names are not allowed among sibling objects. Assign the object a unique name.

Please enter a valid name

Error Type:

Error

Possible Cause:

The Name parameter in the Source dialog is empty.

Solution:

Enter a valid name.

Please enter a valid tag reference

Error Type:

Error

Possible Cause:

The Tag parameter in the Source dialog is empty.

Solution:

Enter a valid tag address.

The severity must range from '<value>' to '<value>'

Error Type:

Error

Possible Cause:

The severity has been set to an invalid value.

Solution:

Change the severity to a value between 1 and 1000.

References to array tags are not supported '<tag name >'

Error Type:

Error

Possible Cause:

A source or sub-condition references an array tag.

Solution:

Change the type of tag being referenced by the source or sub-condition.

References to complex tags are not supported '<tag name>'

Error Type:

Error

Possible Cause:

A source or sub-condition references a complex tag.

Solution:

Change the type of tag being referenced by the source or sub-condition.

Tag '<tag name>' does not exist or you are trying to add a dynamic references which is not allowed

Error Type:

Error

Possible Cause:

A source or sub-condition references a tag that does not exist.

Solution:

Locate the tag in the Tag Browser to verify that it exists.

Index

A

AE Plug-In Conditions	3
AE Plug-In Terminology	3
Area Interface	7

C

Condition Interface	9
---------------------------	---

D

Data Types Description	15
------------------------------	----

E

Error Descriptions	16
--------------------------	----

F

Failed to add the condition '<condition>' to source '<source>'. The source already has this condition.	16
---	----

H

Help Contents	3
---------------------	---

O

Overview	3
----------------	---

P

Please enter a valid name	17
Please enter a valid tag reference	17
Plug-In Interface	7

R

Real World Examples	5
References to array tags are not supported '<tag name>'	17
References to complex tags are not supported '<tag name>'	17

S

Source Interface	8
------------------------	---

T

Tag '<tag name>' does not exist or you are trying to add a dynamic references which is not allowed.	18
The AE Plug-In from the AE Client's Perspective	4

The condition '<condition>' is not supported 16

The deadband range is invalid. The high value must be higher than the low value..... 16

The name '<name>' already exists..... 16

The name '<name>' is invalid..... 17

The severity must range from '<value>' to '<value>'..... 17

U

User Interfaces..... 7