

# Alarms & Events Plug-In

© 2017 PTC Inc. All Rights Reserved.

# Table of Contents

<b>Alarms &amp; Events Plug-In</b> .....	<b>1</b>
<b>Table of Contents</b> .....	<b>2</b>
Alarms & Events Plug-In .....	4
Overview .....	4
OPC AE Plug-In Terminology .....	4
OPC AE Plug-In Conditions .....	5
The OPC AE Plug-In from the OPC AE Clients' Perspective .....	5
Real World Examples .....	6
<b>User Interfaces</b> .....	<b>9</b>
Plug-In Interface .....	9
Area Interface .....	10
Source Interface .....	10
Condition Interface .....	11
<b>CSV Import/Export</b> .....	<b>19</b>
Creating a Template .....	19
Exporting an OPC AE Area .....	19
Importing a CSV File into the Server .....	22
Using Other Characters as the Delimiter .....	22
<b>Data Types Description</b> .....	<b>23</b>
<b>Error Descriptions</b> .....	<b>24</b>
Failed to add the condition <condition> to source <source>. The source already has this condition. ....	24
Failed to reference server tag <tag name> on object <source name.tag name <condition> >. ....	24
Invalid reference to a tag with an unsupported or array data type (<tag name>). ....	25
Please enter a valid name. ....	25
Please enter a valid tag reference. ....	25
References to complex tags are not supported <tag name>. ....	25
Tag <tag name> does not exist or you are trying to add a dynamic reference which is not allowed. ....	25
The condition <condition> is not supported. ....	26
The deadband range is invalid. The high value must be higher than the low value. ....	26
The name <name> already exists. ....	26
The name <name> is invalid. ....	26
The severity must range from <value> to <value>. ....	27

**Index** ..... **28**

## Alarms & Events Plug-In

---

Help version 1.036

### 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?

#### CSV Import / Export

How do I import and export data in a CSV file?

#### Data Types Description

Which data types are supported?

#### Error Descriptions

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

### Overview

---

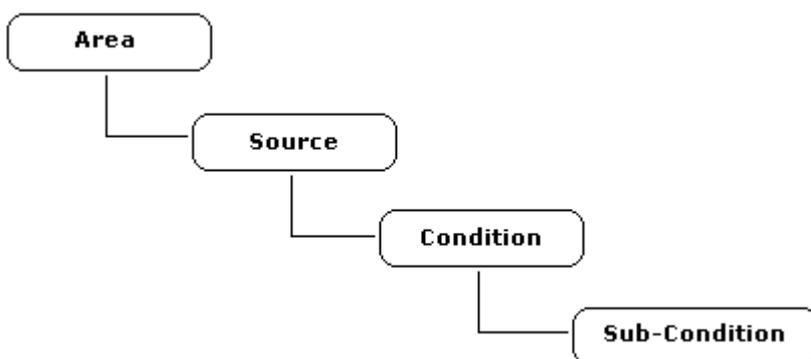
The Alarms & Events Plug-In provides custom alarm support to all devices connected to the OPC server, regardless of whether or not the device natively supports OPC AE. It also allows users to configure alarms using device data. OPC AE clients can then interact with the alarm: when the alarm conditions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

### OPC 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 updates 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 to determine the state of the condition. Changes in the condition's state produce events that are then sent to OPC 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 OPC 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.

## OPC AE Plug-In Conditions

---

The OPC 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 OPC AE Plug-In from the OPC AE Clients' Perspective

---

### Filtering Options

Filtering allows OPC 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 is 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 OPC AE event that is sent to the OPC 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 OPC AE client.

#### **Example 2: HI\_HI, HI, LO, LO\_LO**

Users are interested in temperatures exceeding 225° C. A single HI\_HI condition is 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 OPC 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 OPC AE client must be notified. The ROC\_HI condition is 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 OPC 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. The DEV\_HI condition is 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 OPC AE client.

## User Interfaces

---

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

[Plug-In Interface](#)

[Area Interface](#)

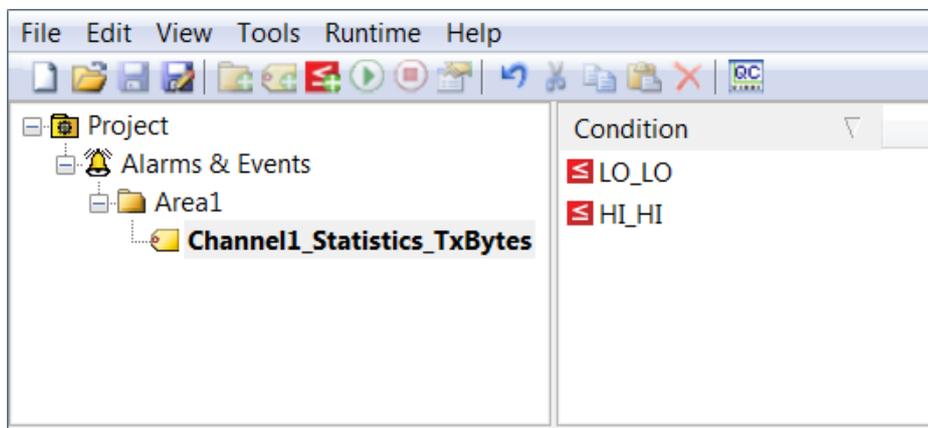
[Source Interface](#)

[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 OPC AE objects (such as area, source and so forth). These options may also be accessed in both views by right-clicking on the OPC 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 "Area1."

2. **Source:** Sources belong to an area. 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 any area.

● **Note:** In the image above, the source is labeled "Channel1\_Statistics\_TxBytes."

## Detail View

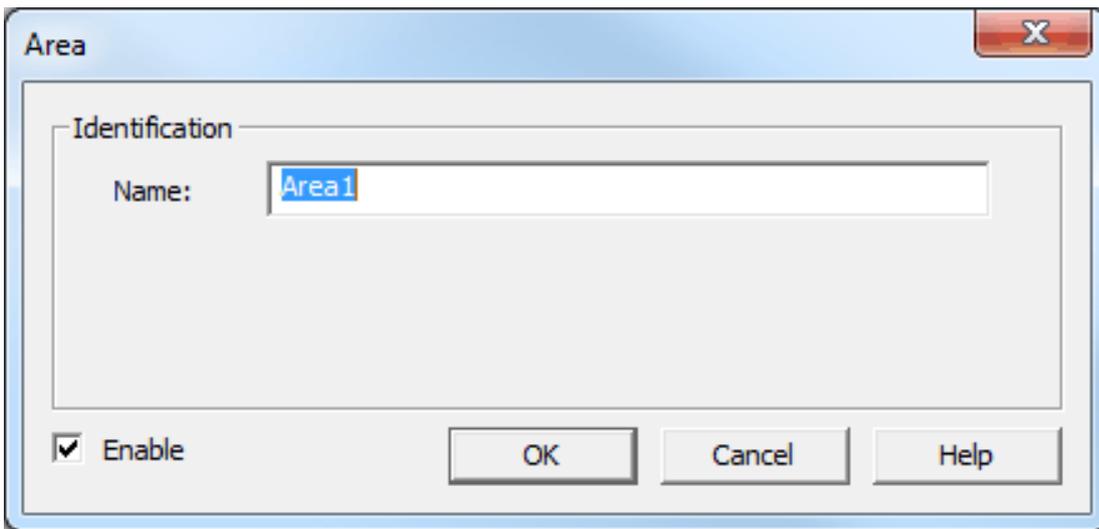
The Detail view displays the conditions that belong to the source selected in the Project view. Users can create a new condition, edit an existing condition, toggle, copy, cut, paste, delete, and drag and drop a condition to a source in the Project view.

## Toolbar

The Toolbar provides convenient access to all of the OPC 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 OPC AE Plug-In.



Description is as follows:

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

## Source Interface

Each source is linked to an OPC server tag.

Descriptions are as follows:

**Tag:** This parameter specifies the address of the OPC server tag that is linked to the source.

**Tag Browser:** When clicked, this button (...) launches the server Tag Browser. Once a tag is selected, its address is placed in the Tag parameter.

**Use the tag name as the source name:** This parameter controls whether or not the specified Name parameter is used. When checked, the Name edit control is grayed out and the source is named after the tag address. When disabled, the user must supply a valid source name.

● **Note:** When enabled, the tag address is altered in the following manner to create a valid source name:

Leading underscores are changed to 0.

Periods are converted to '\_'.

Double underscores are converted to single underscores.

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

**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 is grayed out.

**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 milliseconds. The default setting is 1000 milliseconds.

**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 are 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:

**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 affects 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 [OPC AE Plug-In Conditions](#).

**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:

- **None:** No deadband is used.
- **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.
- **Percent:** This option is similar to Absolute except 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  $\pm 1$ .

**Deadband:** This parameter specifies the deadband value. Any positive number is supported. If the Deadband Type is set to None, this parameter is grayed out. The default value is 0.

**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.

**Enable:** This parameter turns the condition on or off. When enabled, the condition is on and can produce Conditional events. When disabled, the condition is off and cannot produce events.

## Sub-Condition General

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

Descriptions are as follows:

**Description:** The description of what the sub-condition is for. This description is sent to OPC AE clients when they query for the state of a condition. The maximum length is 1024.

**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 OPC 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. Refer to the table below for ranges of severity.

Client Severity	OPC Severity
High	801-1000
Medium High	601-800
Medium	401-600

Client Severity	OPC Severity
Medium Low	201-400
Low	1-200

**Enable:** This parameter enables/disables the sub-condition. If a sub-condition is disabled, it cannot become active.

**Auto-Acknowledge:** This parameter automatically acknowledges the transition into this sub-condition. Sub-condition transitions can be acknowledged by the server or by OPC AE clients.

**In-To:** The string sent to OPC AE clients when the condition transitions in-to the selected sub-condition. The maximum character length is 1024.

**Out-Of:** The string sent to OPC AE clients when the condition transitions out-of the selected sub-condition. The maximum character length is 1024.

**ACK:** The string sent to OPC AE clients when the condition's transition into this sub-condition is acknowledged. The maximum character length is 1024.

### 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 becomes active.

**Threshold:** A static value or tag which is compared to the data updates from the parent source. If the threshold is a tag, any OPC server tag is available aside from array tags and complex tags.

**Value/Tag:** This parameter specifies what a threshold is. The default setting is Value.

**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 >= 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.

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

**Update Rate:** This parameter specifies the tag's update rate. This is the fastest rate at which updates are received from the tag.

**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:

- **None:** No deadband is used.
- **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.
- **Percent:** This option is similar to Absolute except 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  $\pm 1$ .

**Deadband:** This parameter specifies the deadband value. Any positive number is supported. If the Deadband Type is set to None, this parameter is unavailable. The default value is 0.

**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.

The screenshot shows the 'Sub-Condition Triggers' dialog box. On the left, under 'Sub-Conditions', there is a list with 'ROC\_HI' selected and a checked checkbox. On the right, under 'Rate of Change', the 'Threshold Type' is set to 'Value' (radio button selected), 'Threshold' is '0', 'Interval' is 'ms', and 'Comparison' is 'Source > Threshold'. At the bottom of the dialog are four buttons: '< Back', 'Finish', 'Cancel', and 'Help'.

**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.

**Value/Tag:** This parameter specifies the interpretation of the Threshold as either a value or tag. The default setting is Value.

**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 MULTI\_LEVEL, HI\_HI, HI, LO, or LO\_LO is selected as a condition with a threshold type of tag, Update Rate is used. This field holds the update rate for the tag in milliseconds. The range is 100-60000 ms. The default setting is 1000 ms.

**Comparison:** This parameter specifies the comparison type between the ROC of the Threshold and the ROC of the Source. The default setting is "Source > Threshold." 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.

 **Important:** The parameters described below only apply to tags.

**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:

- **None:** No deadband is used.
- **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.
- **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  $\pm 1$ .

**Deadband:** This parameter specifies the deadband value. Any positive number is supported. If the Deadband Type is set to None, this parameter is grayed out. The default value is 0.

**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 dialog box titled "Sub-Condition Triggers". On the left side, there is a list box labeled "Sub-Conditions" containing one item, "DEV\_HI", which is checked. On the right side, there is a "Deviation" section with two input fields for "Value" (both containing "0") and a dropdown menu for "Type" set to "Absolute". At the bottom of the dialog are four buttons: "< Back", "Finish", "Cancel", and "Help".

**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 maximum deviation is 50, the source must exceed 50.

**Type:** This parameter specifies how to interpret the '+/-' value. Options include Absolute and Percent. The default setting is Absolute.

---

## CSV Import/Export

---

The Alarms & Events Plug-In supports the import and export of data in a Comma Separated Variable (CSV) file. CSV import and export efficiently configures a large number of event sources, conditions, and sub-conditions. CSV functions are only available when an OPC AE Area is selected. Furthermore, parameters that are not defined in the CSV file are assigned appropriate default values. For more information on a specific aspect of CSV Import/Export, select a link from the list below.

### [Creating a Template](#)

### [Exporting an OPC AE Area](#)

### [Importing a CSV File into the Server](#)

### [Using Other Characters as the Delimiter](#)

● **Note:** For information on specifying which character to use as the variable (comma or semicolon), refer to "Options - General" in the server help file.

---

## Creating a Template

---

The easiest way to create an import CSV file is to create a template. For more information, refer to the instructions below.

1. To start, create an OPC AE area.
2. Under the area, define a source.
3. Under the source, define a condition of each type to be used in the project.
4. Export the area as a CSV file.
5. Use this template in a spreadsheet application that supports CSV files, and then modify the file as desired.

● **Note:** Users can save the CSV file to disk, and then re-import it into the server in the same area or a new area.

● **Note:** Microsoft Excel is an excellent tool for editing large groups of tags outside the server. Once a template CSV file has been exported, it can be loaded into Excel for editing.

---

## Exporting an OPC AE Area

---

Exporting an OPC AE area generates a .CSV text file that contains a section for sources, conditions, and sub-conditions. The sub-conditions section is further subdivided into sections for value sub-conditions (HI\_HI, HI, LO, LO\_LO), ROC sub-conditions (ROC\_HI, ROC\_LO), and deviation sub-conditions (DEV\_HI, DEV\_LO). Each section contains a heading record followed by a record for each item defined under the selected area. Column names must be exactly the same as those listed; however, columns may be in any order.

● **Important:** The OPC AE CSV source section must precede the conditions section, which must precede the sub-conditions section. Conditions must reference the source to which it belongs, and sub-conditions must reference both the source and condition to which it belongs. However, the items that are referenced need not exist as CSV records, as long as they already exist in the area to which the CSV file is being imported. Note that condition records have no required fields and can therefore be omitted completely if their existing or default fields are desired.

### Source Fields

The required rows are listed in **bold**.

Column Name	Values
<b>Record Type</b>	"Source".
<b>Tag</b>	The tag monitored as the source of alarms.
Name	The source name (with path relative to the area being imported).
Update Rate (ms)	The rate at which the source tag is updated.
Enabled	1 to enable, and 0 to disable.

• See Also: [Source Interface](#)

## Condition Fields

The required rows are listed in **bold**.

Column Name	Values
<b>Record Type</b>	"Condition".
<b>Relative Address</b>	The address of the condition, relative to the area being imported.
Enabled	1 to enable, and 0 to disable.
Deadband Type	The type of deadband used to filter updates to the source tag for the condition. Options include "None," "Absolute," and "Percent".
Deadband	The deadband value. The value must be a positive, real number.
Range Low	The lower range of a percent deadband. The value may be any real number, but must be less than the high range value.
Range High	The higher range of a percent deadband. The value may be any real number, but must be greater than the low range value.

• **Note:** Conditions are created automatically when sub-condition records are imported, so the conditions section (or individual conditions records) can be omitted from the CSV file if the default values are desired.

• See Also: [Condition Interface](#)

## Value Sub-Condition Fields

The required rows are listed in **bold**.

Column Name	Values
<b>Record Type</b>	"ValueSubCondition".
<b>Relative Address</b>	The address of the sub-condition, relative to the area being imported.
Description	The description of the sub-condition. The maximum length of the field is 1024.
Severity	The severity of the sub-condition. The range is 1 to 1000.
Enabled	1 to enable sub-condition, and 0 to disable.
Auto-Acknowledge	1 to enable auto-acknowledgment, and 0 to disable.
In-to	To sub-condition's In-to message. The maximum length of the field is 1024.
Out-of	To sub-condition's Out-of message. The maximum length of the field is 1024.
ACK	To sub-condition's ACK message. The maximum length of the field is 1024.
Threshold is Value	1 if the threshold field contains a value, and 0 if it contains a tag. The default

Column Name	Values
	value is 1.
<b>Threshold</b>	A static value or tag to which the data updates from the parent source are compared.
Comparison	The comparison between the source and the threshold. Options include >, >=, <, <=, ==, and !=. The comparison ">" would make the comparison Source > Threshold.
Update Rate (ms)	The rate at which a tag threshold is updated.
Deadband Type	The type of deadband used to filter updates to a tag threshold. Options include "None," "Absolute," and "Percent".
Deadband	The deadband value. The value must be a positive, real number.
Range Low	The lower range of a percent deadband. The value may be any real number, but must be less than the high range value.
Range High	The higher range of a percent deadband. The value may be any real number, but must be greater than the low range value.

### Deviation Sub-Condition Fields

The required rows are listed in **bold**.

Column Name	Value
<b>Record Type</b>	"DeviationSubCondition".
<b>Relative Address</b>	The address of the sub-condition, relative to the area being imported.
Description	The description of the sub-condition. The maximum length of the field is 1024.
Severity	The severity of the sub-condition. The range is 1 to 1000.
Enabled	1 to enable sub-condition, and 0 to disable.
Auto-Acknowledge	1 to enable auto-acknowledgment, and 0 to disable.
In-to	To sub-condition's In-to message. The maximum length of the field is 1024.
Out-of	To sub-condition's Out-of message. The maximum length of the field is 1024.
ACK	To sub-condition's ACK message. The maximum length of the field is 1024.
<b>Threshold</b>	The fixed point on which deviation is based. The value may be any real number.
<b>Deviation</b>	The value that the source can deviate from the threshold before triggering the sub-condition.
Deviation Type	Determines how the Deviation value is interpreted. Options include "Absolute" and "Percent".

### ROC Sub-Condition Fields

The required rows are listed in **bold**.

Column Name	Value
<b>Record Type</b>	"ROCSUBCondition".
<b>Relative Address</b>	The address of the sub-condition, relative to the area being imported.
Description	The description of the sub-condition. The maximum length of the field is 1024.

Column Name	Value
Severity	The severity of the sub-condition. The range is 1 to 1000.
Enabled	1 to enable sub-condition, and 0 to disable.
Auto-Acknowledge	1 to enable auto-acknowledgment, and 0 to disable.
In-to	To sub-condition's In-to message. The maximum length of the field is 1024.
Out-of	To sub-condition's Out-of message. The maximum length of the field is 1024.
ACK	To sub-condition's ACK message. The maximum length of the field is 1024.
Threshold is Value	1 if the threshold field contains a value, and 0 if it contains a tag. The default value is 1.
<b>Threshold</b>	A static value or tag, representing a rate of change in units/time, to which the parent source are compared.
Comparison	The comparison between the source and the threshold. Options include >, >=, <, <=, ==, and !=. The comparison ">" would make the comparison Source > Threshold.
Interval	This field is used when the threshold is a value. The field specifies the ROC interval. Options include "ms", "sec", and "min". The default value is "ms".
Update Rate (ms)	This field is used when the threshold is a tag. The field specifies the update rate of the threshold tag in milliseconds. The range is 100 to 60000 ms.
Deadband Type	The type of deadband used to filter updates to a tag threshold. Options include "None," "Absolute," and "Percent".
Deadband	The deadband value. The value must be a positive, real number.
Range Low	The lower range of a percent deadband. The value may be any real number, but must be less than the high range value.
Range High	The higher range of a percent deadband. The value may be any real number, but must be greater than the low range value.

• See Also: [Condition Interface](#)

## Importing a CSV File into the Server

Once the CSV file has been edited, it can be re-imported into the server by selecting **File | Import CSV**.

• **Note:** This option is only available when an area is selected.

## Using Other Characters as the Delimiter

When utilizing a CSV file that does not use a comma or semi-colon delimiter, do one of the following:

1. Save the project in XML, then perform mass configuration on the XML file instead of using CSV.
2. Perform a search-and-replace on the delimiter in the CSV file and then replace the delimiter with a comma or semicolon. The delimiter being used by the server (either comma or semi-colon) must be set to the replacement character.

• For more information, refer to "Options - General" in the server help file.

## 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 occur. Click on the link for a description of the message.

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

[Failed to reference server tag <tag name> on object <source name.tag name <condition>>.](#)

[Invalid reference to a tag with an unsupported or array data type \(<tag name>\).](#)

[Please enter a valid name.](#)

[Please enter a valid tag reference.](#)

[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.](#)

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

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

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

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

**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.

**Failed to reference server tag <tag name> on object <source name.tag name <condition>> .**

---

### Error Type:

Error

### Possible Cause:

The plug-in attempted to add an invalid item (e.g., the item does not exist in the project, has an invalid address, or is an unsupported data type.)

### Solution:

Locate and correct the item that could not be added.

---

**Invalid reference to a tag with an unsupported or array data type (<tag name>).**

---

**Error Type:**

Error

**Possible Cause:**

A source or sub-condition references a tag with an array or other unsupported type.

**Solution:**

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

---

**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.

---

**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 reference which is not allowed.**

---

**Error Type:**

Error

**Possible Cause:**

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

**Solution:**

Remove any referenced dynamic tags and choose only static tags.  
Locate the tag in the Tag Browser to verify that it exists.

**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 Alarms and Events 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.

**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.

**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.

# Index

## A

Area Interface 10

## C

Condition Interface 11

Creating a Template 19

CSV Import/Export 19

## D

Data Types Description 23

## E

Error Descriptions 24

Exporting an OPC AE Area 19

## F

Failed to add the condition to source. The source already has this condition. 24

Failed to reference server tag <tag name> on object <source name.tag name <condition> >. 24

## H

Help Contents 4

## I

Importing a CSV File into the Server 22

Invalid reference to a tag with an unsupported or array data type (<tag name>). 25

**O**

OPC AE Plug-In Conditions 5

OPC AE Plug-In Terminology 4

Overview 4

**P**

Please enter a valid name. 25

Please enter a valid tag reference. 25

Plug-In Interface 9

**R**

Real World Examples 6

References to complex tags are not supported <tag name>. 25

**S**

Source Interface 10

**T**

Tag does not exist or you are trying to add a dynamic reference which is not allowed. 25

The condition <condition> is not supported. 26

The deadband range is invalid. The high value must be higher than the low value. 26

The name <name> already exists. 26

The name <name> is invalid. 26

The OPC AE Plug-In from the OPC AE Client's Perspective 5

The severity must range from <value> to <value>. 27

**U**

User Interfaces 9

Using Other Characters as the Delimiter 22