Aras Innovator 33

AML Security Settings

Document #: *D-008086*

Last Modified: *8/22/2024*

Copyright Information

Aras Corporation

100 Brickstone Square

Suite 100

Andover, MA 01810

Phone: 978-691-8900

E-mail: support@aras.com

Website: https://www.aras.com/

Table of Contents

Send Us Your Comments

Aras Corporation welcomes your comments and suggestions on the quality and usefulness of this document. Your input is an important part of the information used for future revisions.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where and what level of detail?
- Are the examples correct? Do you need more examples?
- What features did you like most?

If you find any errors or have any other suggestions for improvement, indicate the document title, and the chapter, section, and page number (if available).

You can send comments to us in the following ways:

Email:
TechDocs@aras.com
Subject: Aras Product Documentation

Or,

Postal service:
Aras Corporation
100 Brickstone Square
Suite 100
Andover, MA 01810
Attention: Aras Technical Documentation

If you would like a reply, provide your name, email address, address, and telephone number.

If you have usage issues with the software, visit https://www.aras.com/support/

1 Overview

Starting with the 11.0 SP9 release, Aras Innovator introduced additional security features to the AML which is parsed by the Aras Innovator Server upon receipt from client machines. These changes include:

- Verification of all values in Where attributes, Between condition, and In condition clauses executed by a client or client-side Methods.
- The inclusion of a Suppression List in the core product that allows for exceptions to be added on a case-by-case basis.
- The addition of a new global Operating Parameter in the InnovatorServerConfig.xml file to control this feature.

Note: Values of *where/in/between* attributes of AMLs executed from server-side methods are never validated.

All Where Attributes and Properties with the conditions In or Between present in the AML that are sent from the client to the server, prior to executing OnBefore* Server Events, will now be parsed, and verified to ensure that they are not using SELECT, UPDATE, EXISTS, or any SQL statements which could potentially be used to retrieve or update data on SQL Tables unrelated to the overall AML query. When an invalid criterion is passed in AML from a client, an Item Analysis error is now returned, with additional details available for Administrators. A full list of valid SQL Tokens is included in Appendix A – Allowed SQL.

Suppression of invalidating specific SQL statements can be introduced into Aras Innovator by creating new ItemSuppression files, in an XML format, in the Innovator Server's App_Data. This allows for specific use cases with SQL statements to pass validation.

A new Operating Parameter in the InnovatorServerConfig.xml file has also been introduced that allows disabling validation of values of *where/in/between* attributes of AMLs passed from a client if existing code relied heavily on this feature prior to upgrading to Aras Innovator 11.0 SP9 or higher. The Operating Parameter will ignore all Item Analysis on the AML and treat Where Attributes and Properties with the conditions In or Between as in pre-11.0 SP9 environments. Disabling the validation is introduced for legacy purposes; however, this parameter will not be available in future major versions of Aras Innovator. We recommend either adding permanent suppressions in the ItemSuppression files or altering any affected code to eliminate the reliance on these queries.

2 AML Analysis

Prior to Aras Innovator 11.0 SP9, Aras Innovator supported the use of any SQL to qualify the data that needed to be retrieved, in the form of the `Where` attribute on `<Item>` tags, and the `condition` attribute's `In` and `Between` on Properties. As SQL operates below the permissions model, as opposed to the restrictions already in place on AML, Aras has introduced an Item Analysis functionality to ensure that all queries respect the permissions model and prevent SQL injections.

To this effect, when an AML sent from a client contains *where/in/between* attributes, Aras Innovator will validate values of these attributes and generate an Item Analysis Error if the validation failed. The error generated depends on the User's Identity List. A non-administrator, upon executing an invalid AML query, will see an error that recommends they contact their Administrator. For example:

User query:

```
<AML>
    <Item action="get" type="Part" where="[Part].id IN (SELECT id FROM [PART].id WHERE
    id IN ('EBA30FA710AC4753B76E43DED126CF76', 'FE8D32A113714C0498C271191210F19E'))"/>
</AML>
```

Response:

```
<SOAP-ENV:Fault xmlns:af="http://www.aras.com/InnovatorFault">
    <faultcode>SOAP-ENV:Server.ItemAnalysisException</faultcode>
    <faultstring><![CDATA[Item Analysis Error. Some Items have incorrect
    attribute/property values syntax. Please contact your system administrator for
    more details.]]> </faultstring>
    …
</SOAP-ENV:Fault>
```

When a User who is a Member of the Administrator Identity runs the same query, they will see a breakdown of the error, and what specific section of the code is incorrect. For example:

Administrator query:

```
<AML>
    <Item action="get" type="Part" where="[Part].id IN (SELECT id FROM [PART].id WHERE
    id IN ('EBA30FA710AC4753B76E43DED126CF76', 'FE8D32A113714C0498C271191210F19E'))"/>
</AML>
```

Response:

```
<SOAP-ENV:Fault xmlns:af="http://www.aras.com/InnovatorFault">
    <faultcode>SOAP-ENV:Server.ItemAnalysisException</faultcode>
    <faultstring>
        <![CDATA[Item Analysis Error. Some Items have incorrect attribute/property values
        syntax. See details for more information.
        Details:
        Incorrect value: "where="[Part].id IN (SELECT id FROM [PART].id WHERE id IN
        ('EBA30FA710AC4753B76E43DED126CF76', 'FE8D32A113714C0498C271191210F19E'))"".
        Incorrect value explanation: "SELECT" is forbidden in where attribute.
        ]]>
    </faultstring>
    …
</SOAP-ENV:Fault>
```

Note that the validation mechanism will still allow conditions on properties of the requested ItemType including cases when the condition contains standard SQL qualifiers. For example, using an attribute such as `<Item type="Part" ... where="[Part].id = '0123456789abcdef'" ... />` will execute the same as before.

Aras Support recommends, if you encounter an Item Analysis Error, that the AML query be adjusted to conform to the new rules.

3 Suppression Examples

In the use case that a query must be executed with an open-ended SQL clause to obtain the correct values, Aras Innovator provides the option to explicitly suppress the Item Analysis for such queries.

To generate a Suppression, an Administrator must create an ItemAnalysis.Suppressions.*.xml file in the \Innovator\Server\App_Data folder, such as \Innovator\Server\App_Data\ItemAnalysis.Suppressions.MyCustom.xml. There can be multiple ItemAnalysis.Suppressions.*.xml files, each with their own unique set of Suppressions.

The format of the ItemAnalysis.Suppessions.*.xml file is as follows:

```
<itemAnalysis>
    <suppressions>
        <!-- Suppression Parameters -->
    </suppressions>
</itemAnalysis>
```

The Suppressions support `@Parameter` and `@ParametersList` as variables, which correspond with a SQL Constant and a SQL Constants List, respectively.

Examples of Suppression declarations are listed in Sections 3.1, 3.2, and 3.3 below.

3.1 Attribute Examples

3.1.1 Where Attribute Suppression Example

For each Where Attribute that needs to be suppressed, you must add a `<whereAttribute>` node.

The following is an example of a `<whereAttribute>` node in the ItemAnalysis.Suppressions file:

```
<whereAttribute>
    <template><![CDATA[[Part].id IN (SELECT id FROM [PART] WHERE cost =
    @Parameter)]]></template>
</whereAttribute>
```

With this node, the following AML sent from a client will successfully pass the validation:

```
<AML>
    <Item action="get" type="Part" where="[Part].id IN (SELECT id FROM [PART] WHERE
    cost = 5)"/>
</AML>
```

3.1.2 Condition In Attribute Suppression Example

For each Condition In Attribute that needs to be suppressed, you must add a `<conditionInProperty>` node.

The following is an example of a `<conditionInProperty>` node in the ItemAnalysis.Suppressions file:

```
<conditionInProperty>
    <template><![CDATA[(SELECT cost FROM [innovator].[PART] WHERE [item_number] IN
    (@ParametersList))]]></template>
</conditionInProperty>
```

With this node, the following AML sent from a client will successfully pass the validation:

```
<Item action="get" type="Part">
    <cost condition="in">(SELECT cost FROM [innovator].[PART] WHERE [item_number] IN
    ('A', 'C', 'D'))</cost>
</Item>
```

3.1.3 Condition Between Attribute Suppression

For each Condition Between Attribute that needs to be suppressed, you must add a `<conditionBetweenProperty>` node.

The following is an example of a `<conditionBetweenProperty>` node in the ItemAnalysis.Suppressions file:

```
<conditionBetweenProperty>
    <template><![CDATA[(SELECT cost FROM [innovator].[PART] WHERE [item_number] =
    @Parameter) and (SELECT cost FROM [innovator].[PART] WHERE [item_number] =
    @Parameter)]]></template>
</conditionBetweenProperty>
```

With this node, the following AML sent from a client will successfully pass the validation:

```
<Item action="get" type="Part">
    <cost condition="between">(SELECT cost FROM [innovator].[PART] WHERE [item_number]
    = 'B') and (SELECT cost FROM [innovator].[PART] WHERE [item_number] = 'D')</cost>
</Item>
```

3.2 Specificity of Suppressions and Multiple Suppressions

As many Suppressions as required can be added to a single ItemAnalysis.Suppressions file. To add additional Suppressions, a second node of the required values must be added. An example of multiple Suppressions in a single file is included below:

```
<itemAnalysis>
  <suppressions>
    <whereAttribute>
      <template><![CDATA[[Part].id IN (SELECT id FROM [PART] WHERE description =
@Parameter)]]></template>
    </whereAttribute>
    <whereAttribute>
      <template><![CDATA[[Part].id IN (SELECT id FROM [PART] WHERE cost >
@Parameter)]]></template>
    </whereAttribute>
    <conditionInProperty>
```

```
      <template><![CDATA[(SELECT cost FROM [innovator].[PART] WHERE [item_number] IN
(@ParametersList))]]></template>
    </conditionInProperty>
    ...
  </suppressions>
</itemAnalysis>
```

As demonstrated, however, the Suppressions that are defined in the ItemAnalysis.Suppressions file are explicit in their interpretation. If an AML sent from a client does not pass the ItemAnalysis or match the Suppression precisely, the AML will be rejected.

Note: Suppressions are also case-sensitive.

As an example, given the original `<whereAttribute>` Suppression node:

```
<whereAttribute>
    <template><![CDATA[[Part].id IN (SELECT id FROM [PART] WHERE cost =
    @Parameter)]]></template>
</whereAttribute>
```

The following query will succeed:

```
<AML>
    <Item action="get" type="Part" where="[Part].id IN (SELECT id FROM [PART] WHERE
    cost = 5)"/>
</AML>
```

While the following query will throw an Item Analysis Error:

```
<AML>
    <Item action="get" type="Part" where="[Part].id IN (SELECT id FROM [PART] WHERE
    cost > 5)"/>
</AML>
```

In order for the second query to succeed, an additional `<whereAttribute>` node must be added to an ItemAnalysis.Suppressions file.

4 Disabling Item Analysis Globally

In certain scenarios, for example working with legacy systems undergoing upgrades or containing older solutions, an Administrator may find the need to disable the Item Analysis functionality in order to return the system to production-readiness quickly. If this behavior is a requirement, and individual Suppression events do not meet this requirement adequately, it is currently possible to disable the Item Analysis functionality.

To disable the Item Analysis functionality, you must add the `<operating_parameter …/>` `parse_item` to the `InnovatorServerConfig.xml` file, located in the root folder of the Aras Innovator installation, and set the value to false. See the following example:

```
<Innovator>
  ...
  <operating_parameter key="parse_item" value="false"/>
  ...
</Innovator>
```

This will disable all Item Analysis functionality. Queries using functions like the examples listed in Sections 3.1, 3.2, and 3.3 will function as expected.

Note: **The `<operating_parameter key="parse_item" …/>` is for legacy purposes only. This parameter will not exist in any future major version of Aras Innovator. As such, Aras Support recommends either updating any Client-side Methods or Actions that require this option so they no longer rely on this code or adding in a Suppression rule to ensure that the functionality will continue to work beyond Aras Innovator 11.0.**

5 Appendix A – Allowed SQL

This Appendix contains a list of all allowed SQL tokens and functions which can be used in a Where attribute, In condition, or Between condition.

5.1 Allowed SQL Tokens

SQL Token	Comments
Add	
And	
As	
Between	
Case	
Coalesce	
Collate	
Convert	
CurrentDate	See Reserved Keywords (https://msdn.microsoft.com/en-us/library/ms189822.aspx)
CurrentTime	
CurrentTimestamp	
Double	
Else	
End	
Escape	
In	

SQL Token	Comments
Is	
Like	
Not	
Null	
NullIf	
Of	
On	
Or	
Some	
Then	
When	
Precision	
TryConvert	
Bang	Symbol "!"
PercentSign	
Ampersand	
LeftParenthesis	
RightParenthesis	
LeftCurly	
RightCurly	
Star	
MultiplyEquals	
Plus	
Comma	
Minus	

SQL Token	Comments
Dot	
Divide	
Colon	
DoubleColon	
Semicolon	
LessThan	
EqualsSign	
GreaterThan	
Circumflex	
VerticalLine	
Tilde	
AddEquals	
SubtractEquals	
DivideEquals	
ModEquals	
BitwiseAndEquals	
BitwiseOrEquals	
BitwiseXorEquals	
Integer	
Numeric	
Real	
HexLiteral	
Money	
DollarPartition	
AsciiStringOrQuotedIdentifier	

SQL Token	Comments
AsciiStringLiteral	
UnicodeStringLiteral	
Identifier	
QuotedIdentifier	
SingleLineComment	
MultilineComment	
WhiteSpace	

5.2 Allowed SQL Functions

SQL Function Name	Comments
GETDATE	
GETUTCDATE	