

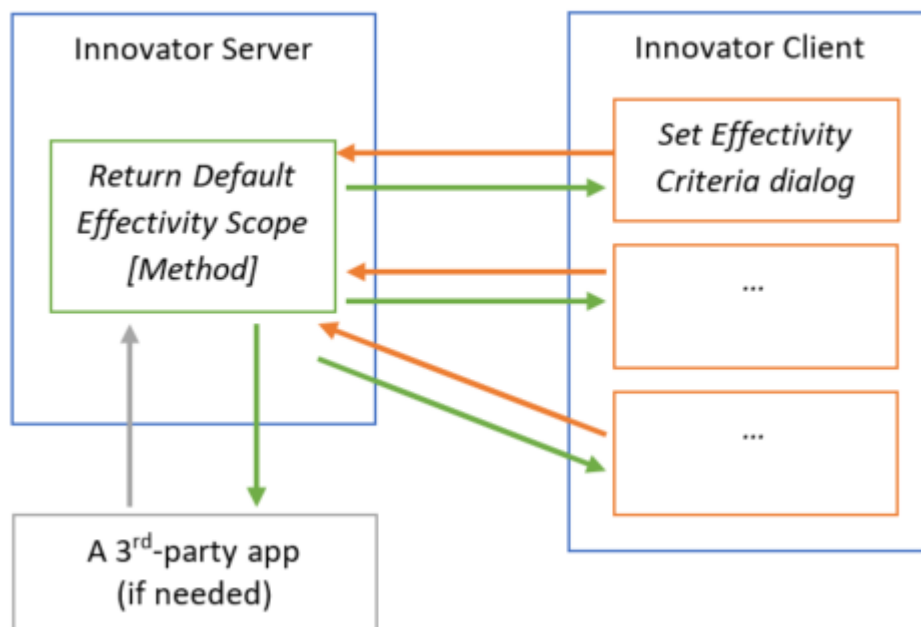
Appendix

How-to Example - Setting a default Effectivity Scope in “Effectivity Criteria Filter” dialog

This section discusses an approach to set a default Effectivity Scope on the Effectivity Criteria dialog with the corresponding variables based on the default scope.

There can be a variety of business rules to determine the default effectivity scope, such as item classification, item state, user identities, and teams. It may be necessary to pre-populate the Effectivity scope only if a single effectivity scope can be determined. In addition, business-specific logic may change over time.

It is ideal to have a single place where all default rule logic is defined/implemented and retrieved. A server-side method – with parameters as needed – will be an entry point for querying a default effectivity scope. The results can be cached across many requests. This server method can be called from either the client or server-side as shown in the following figure.



For the new server-side method to identify a default effectivity scope:

- A new server-side method is created and given a name such as `effs_getScopeDefaults` . This name is abstract enough to expand the method's functionality as needed. For example, the method can return not only a default scope, but also default values for variables in that scope.
- The method can return a JSON string. The format may look like:

```
{
  "scope": {
    "id": "32-hex-id"
  }
}
```

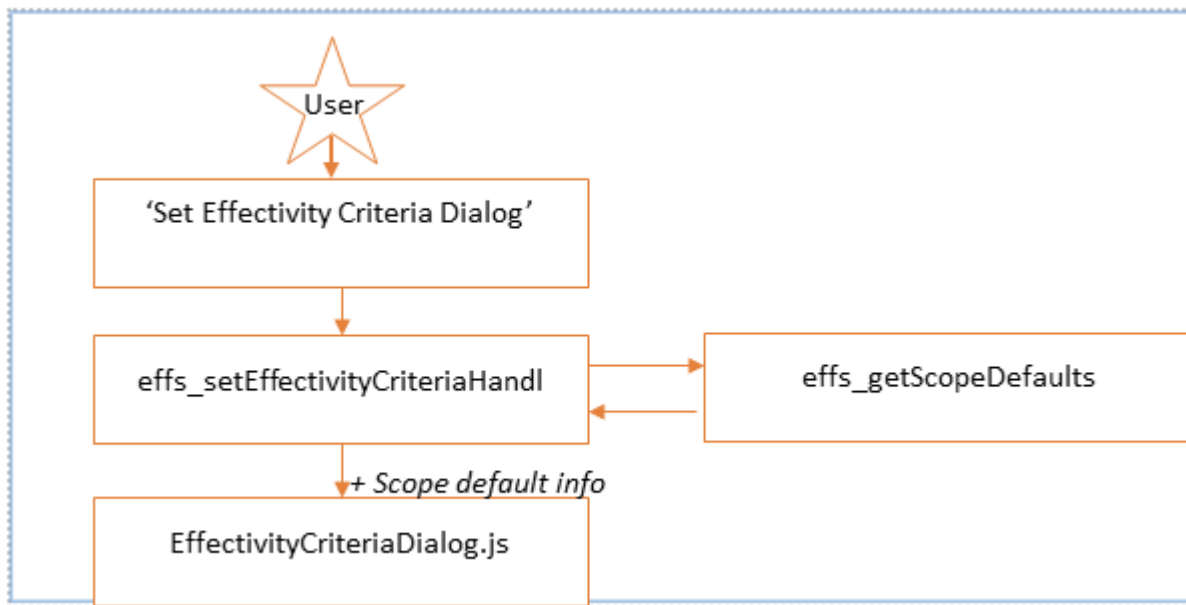
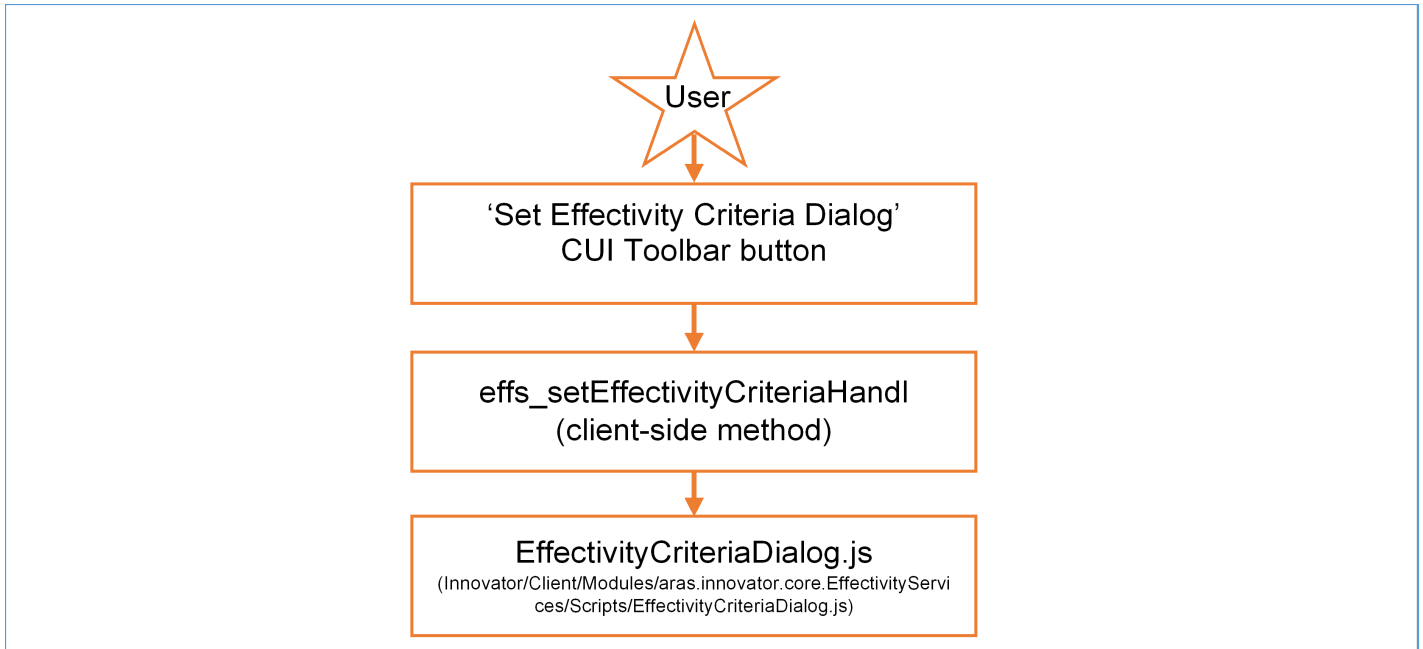
The value of the `scope.id` property is the item ID of the effectivity scope item (`effs_scope ItemType`).

This format provides flexibility to consider future improvements with minimal to no changes.

- The method may accept parameters, as needed, for processing of the business logic.

As for the client-side changes to the Effectivity Criteria dialog, the current order of events is outlined as shown below.





Requesting the default scope from the server and passing that information along to the Effectivity Criteria dialog to be displayed requires the following modification:

The changes outlined in the previous figure can be accomplished with the following:



- In the `effs_setEffectivityCriteriaHandle` client-side method, make a request to the new server-side method to get default scope details. A client-side request to a server-side method can be achieved using `aras.applyMethod('effs_getScopeDefaults','');`
- Construct an object from the response string that would be expected in the dialog `EffectivityCriteriaDialog`'s constructor (modifications in the constructor would be a next step). You can create an object from a string using `JSON.parse()`.
- Take the `scope` property from the just constructed object and pass it as a new additional argument to `EffectivityCriteriaDialog`'s object creation in the `effs_setEffectivityCriteriaHandle` client-side method.
- After the changes from the steps outlined above, the `effs_setEffectivityCriteriaHandle` client-side method will look similar to the following:

```

42 → if (!contextParameters.effectivityCriteriaDialog) {
43 →     /* START: CHANGES FOR DEFAULTING SCOPE */
44 →     const scopeDefaultsStringResponse = aras.applyMethod('effs_getScopeDefaults','');
45 →     const scopeDefaults = JSON.parse(scopeDefaultsStringResponse);
46 →     contextParameters.effectivityCriteriaDialog = new EffectivityCriteriaDialog(window.aras, scopeDefaults.scope);
47 →     /* END: CHANGES FOR DEFAULTING SCOPE */
48 → }
49 →
50 → contextParameters.effectivityResolutionConfigurationQueryDefinitionId = aras.evalMethod('effs_getResolutionCfgQdIdByTgvId','',{
51 →     treeGridViewDefinitionId: contextParameters.tgvContext.getViewDefinitionId()
52 → });

```

- Open the following file:
Innovator/Client/Modules/aras.innovator.core.EffectivityServices/Scripts/EffectivityCriteriaDialog
- Update the `EffectivityCriteriaDialog`'s constructor to process the new parameter. To pre-populate the scope field on the dialog form, add the `this.scope.id` property in the constructor. The resulting constructor may look like the following:

```

5 →     scope: null,
6 →
7 →     constructor: function (aras, scopeDefaults) {
8 →         this.aras = aras;
9 →         this.scope = scopeDefaults;
10 →     },

```

How-to Example - Executing Query Definitions with Effectivity Resolution



This section outlines executing a Query Definition and applying Effectivity Resolution. This involves:

1. Retrieving the Query Definition AML
2. Specifying values for the query parameters of the Query Definition
3. Executing the Query Definition with Effectivity Resolution

Retrieve Query Definition AML

You can retrieve the Query Definition AML using the following AML query:

```
<Item type="qry_QueryDefinition" id="%ID%" action="get" levels="2" />
```

Important

%ID% should be replaced with the ID of the Query Definition that will be executed with Effectivity Resolution.

Specifying Query Parameter Values for the Query Definition

To execute a Query Definition with Effectivity Resolution, values for the appropriate query parameters must be specified. This can be done using the Query Definition's AML as the starting point.

1. Open the Query Definition AML retrieved in the previous section.
2. Find the “ **EffectivityScopeId** ” **qry_QueryParameter** item. Add the following XML node:
%value% with the Effectivity Scope Item ID as the value of the node. In the following example, the ID of the “Aras Part BOM Scope” Effectivity Scope is used.

```
1 <Item type="qry_QueryParameter" typeId="1F23B91C84264C6D9DF614DF2833D99D" id="A87B2A2E5C06499EBA565741219C511C">
2   <behavior>fixed</behavior>
3   <config_id keyed_name="A87B2A2E5C06499EBA565741219C511C" type="qry_QueryParameter">A87B2A2E5C06499EBA565741219C!
4   <created_by_id keyed_name="Super User" type="User">AD30A6D8D3B642F5A2AFED1A4B02BEFA</created_by_id>
5   <created_on>2019-04-25T06:16:42</created_on>
6   <generation>1</generation>
7   <id keyed_name="A87B2A2E5C06499EBA565741219C511C" type="qry_QueryParameter">A87B2A2E5C06499EBA565741219C511C</id>
8   <is_current>1</is_current>
9   <is_released>0</is_released>
10  <keyed_name>A87B2A2E5C06499EBA565741219C511C</keyed_name>
11  <major_rev>A</major_rev>
12  <modified_by_id keyed_name="Super User" type="User">AD30A6D8D3B642F5A2AFED1A4B02BEFA</modified_by_id>
13  <modified_on>2019-04-25T06:16:42</modified_on>
14  <name>EffectivityScopeId</name>
15  <new_version>1</new_version>
16  <not_lockable>0</not_lockable>
17  <permission_id keyed_name="Query Definition - In Work" type="Permission" origPermission="08862820B2384F2D946029!
18  <sort_order>384</sort_order>
19  <source_id keyed_name="PE_BomStructure" type="qry_QueryDefinition">F40469675A0C4137B5504B591995D04A</source_id>
20  <value>D1916568FD3C4E5C92FFE42A57B60CC8</value> E.g., id of the 'Aras Part BOM Scope' Effectivity Scope item
21 </Item>
```



3. Prepare the Effectivity Criteria defined in free-form Boolean language in the following format:

```
<EQ>...</EQ><EQ>...</EQ>...<EQ>...</EQ>
```

For example:

```
<EQ><variable id="ID_OF_MODEL_VARIABLE"/><named-constant id="ID_OF_MODEL_IT
```

Important

To understand the Scope structure required and to retrieve the correct IDs for each criterion, it is best to run a request to the `cfg_GetScopeStructure` method. Refer to the Section 5.2.1 of the Aras Innovator 40 - Configurator Services Programmers Guide for further details.

4. Escape the XML generated in the previous step to make it XML-safe. This can be done using a text editor or an online XML escape tool. Save the resulting escaped string.

5. Find the `EffectivityCriteriaDefinition` `qry_QueryParameter` item. Add the following XML node: `%value%` with the escaped string defining the prepared Effectivity Criteria as the value of the node, similar to the following example:

```
1 <Item type="qry_QueryParameter" typeId="1F23B91C84264C6D9DF614DF2833D99D" id="968FB53DD6AB404B8073377650458BDD">
2   <behavior>fixed</behavior>
3   <config_id keyed_name="968FB53DD6AB404B8073377650458BDD" type="qry_QueryParameter">968FB53DD6AB404B8073377650458BDD
4   </config_id>
5   <created_by_id keyed_name="Super User" type="User">AD30A6D8D3B642F5A2AFED1A4B02BEFA</created_by_id>
6   <created_on>2019-04-25T06:16:42</created_on>
7   <generation>1</generation>
8   <id keyed_name="968FB53DD6AB404B8073377650458BDD" type="qry_QueryParameter">968FB53DD6AB404B8073377650458BDD</id>
9   <is_current>1</is_current>
10  <is_released>0</is_released>
11  <keyed_name>968FB53DD6AB404B8073377650458BDD</keyed_name>
12  <major_rev>A</major_rev>
13  <modified_by_id keyed_name="Super User" type="User">AD30A6D8D3B642F5A2AFED1A4B02BEFA</modified_by_id>
14  <modified_on>2019-04-25T06:16:42</modified_on>
15  <name>EffectivityCriteriaDefinition</name>
16  <new_version>1</new_version>
17  <not_lockable>0</not_lockable>
18  <permission_id keyed_name="Query Definition - In Work" type="Permission" origPermission=
19  "08862820B2384F2D9460295A5B4CB0F1">08862820B2384F2D9460295A5B4CB0F1</permission_id>
20  <sort_order>512</sort_order>
21  <source_id keyed_name="PE_BomStructure" type="qry_QueryDefinition">F40469675A0C4137B5504B591995D04A</source_id>
22  <value>&lt;eq&gt;&lt;variable id=&quot;E0A261A8F65649F0B87DE2B8DB23CC20&quot;/&gt;&lt;constant type=&quot;int
23  &quot;&gt;&lt;11&lt;/constant&gt;&lt;/eq&gt;&lt;/value> E.g., Unit = 11
24 </Item>
```

6. Find the `qry_QueryReference` item with no `parent_ref_id` node inside it. This item represents the entry point, or top node, for traversing a structure. Add the following XML node: `...` to provide a start condition.



If the Query Definition's top level (root) Item is an ItemType Item (e.g. Part), use:

```
<filter_xml>&lt;condition>

&lt;eq>

&lt;property name='id' />

&lt;constant>%ITEM-ID%&lt;/constant>

&lt;/eq>

&lt;/condition></filter_xml>
```

If the Query Definition's top level (root) Item is a RelationshipType Item (e.g. Part BOM), use:

```
<filter_xml>&lt;condition>

&lt;eq>

&lt;property name='source_id' />

&lt;constant>%ITEM-ID%&lt;/constant>

&lt;/eq>

&lt;/condition></filter_xml>
```

Important

%ITEM-ID% should be replaced with the ID of the Item that is the root item in the structure where effectivity will be applied.



```

<Item type="qry_QueryReference" typeId="F96B52A885B04917A1C34251248E47D6" id="799DC78332E741B3BD9CC114E4E66D8C">
  <behavior>float</behavior>
  <child_ref_id>C367FA04BD74494FA9182CC1BE6031CC</child_ref_id>
  <config_id keyed_name="799DC78332E741B3BD9CC114E4E66D8C" type="qry_QueryReference">799DC78332E741B3BD9CC114E4E66D8C
  </config_id>
  <created_by_id keyed_name="Super User" type="User">AD30A6D8D3B642F5A2AFED1A4B02BEFA</created_by_id>
  <created_on>2020-03-24T05:48:48</created_on>
  <generation>1</generation>
  <id keyed_name="799DC78332E741B3BD9CC114E4E66D8C" type="qry_QueryReference">799DC78332E741B3BD9CC114E4E66D8C</id>
  <is_current>1</is_current>
  <is_released>0</is_released>
  <keyed_name>799DC78332E741B3BD9CC114E4E66D8C</keyed_name>
  <major_rev>A</major_rev>
  <modified_by_id keyed_name="Super User" type="User">AD30A6D8D3B642F5A2AFED1A4B02BEFA</modified_by_id>
  <modified_on>2020-03-24T05:48:48</modified_on>
  <new_version>1</new_version>
  <not_lockable>0</not_lockable>
  <permission_id keyed_name="Query Definition - In Work" type="Permission" origPermission=
  "08862820B2384F2D9460295A5B4CB0F1">08862820B2384F2D9460295A5B4CB0F1</permission_id>
  <sort_order>128</sort_order>
  <source_id keyed_name="PE BomStructure" type="qry_QueryDefinition">F40469675A0C4137B5504B591995D04A</source_id>
  <filter_xml><lt;condition>
    <eq>
      <property name="source_id" />
      <constant>5957740D88BA420FA40CE2B98A610F9E</constant>
    </eq>
  </condition></filter_xml>
</Item>

```

E.g., [Part BOM].source_id = '5957740D88BA420FA40CE2B98A610F9E'

7. Save the AML response.

Query Definition Execution

1. Open the saved AML response from step 7 from the previous section.
2. Find the root `qry_QueryDefinition` item and set the value of its `action` attribute to `qry_ExecuteQueryDefinition`.

```

1 <Item type="qry_QueryDefinition" typeId="0421B558114B4B22886C0F4206C923EE" id="F40469675A0C4137B5504B591995D04A"
2   action="qry_ExecuteQueryDefinition">
3     <config_id keyed_name="PE_BomStructure" type="qry_QueryDefinition">F40469675A0C4137B5504B591995D04A
4     </config_id>
5     <created_by_id keyed_name="Super User" type="User">AD30A6D8D3B642F5A2AFED1A4B02BEFA</created_by_id>
6     <created_on>2019-04-25T06:16:41</created_on>
7     <current_state name="Start" keyed_name="Start" type="Life Cycle State">A19F8DFC829349B79928EE674977ED5F
8     </current_state>
9     <generation>1</generation>
10    <id keyed_name="PE_BomStructure" type="qry_QueryDefinition">F40469675A0C4137B5504B591995D04A</id>
11    <is_current>1</is_current>
12    <is_released>0</is_released>
13    <keyed_name>PE_BomStructure</keyed_name>
14    <major_rev>A</major_rev>
15    <modified_by_id keyed_name="Super User" type="User">AD30A6D8D3B642F5A2AFED1A4B02BEFA</modified_by_id>
16    <modified_on>2019-04-25T06:16:42</modified_on>
17    <new_version>1</new_version>
18    <not_lockable>0</not_lockable>
19    <permission_id keyed_name="Query Definition - In Work" type="Permission" origPermission=
20    "08862820B2384F2D9460295A5B4CB0F1">08862820B2384F2D9460295A5B4CB0F1</permission_id>
21    <state>Start</state>
22    <name>PE_BomStructure</name>

```

3. The resulting AML can now be run to execute the Query Definition with Effectivity Resolution applied using the set parameters for the resolution.

