

SQL Schema Compare Entity Filters customization

An entity filter is a new feature introduced in the version 4 of the command line that replaces the schema filters supported by the previous versions. It allows you to include or exclude database objects based on some predefined criteria.

i Schema filters supported by the previous versions of the command line are depreciated and will be removed in a future release.

An entity filter is composed by groups combined with an operator: AND/OR. Each group contains conditions, which act as the filter criteria. When the filter is applied, the conditions are evaluated against the name or the schema of a database object. If the object meets the condition criteria, it is included in the comparison, otherwise it is excluded.

The following xml fragment defines a filter that selects views the name of which starts with the prefix v_emp:

```
<EntityFilters>
  <filter entitytype="View">
    <condition>
      <conditiontype>StartingWith</conditiontype>
      <expression>v_emp</expression>
    </condition>
  </filter>
</EntityFilters>
```

The following xml fragment defines a filter that contains two conditions. It selects tables the name of which does not start with "#" and "##":

```
<entityfilters>
  <filter entitytype="Table">
    <group conditionoperator="And">
      <condition>
        <conditiontype>NotStartingWith</conditiontype>
        <expression>#</expression>
      </condition>
      <condition>
        <conditiontype>NotStartingWith</conditiontype>
        <expression>##</expression>
      </condition>
    </group>
  </filter>
</entityfilters>
```

i The command line provides samples for the most common entity filter scenarios. You will find them in the command line sample folder under **\Program Files\Idera\SQL comparison toolset\Command Line Samples\Schema Compare\EntityFilters**.

The structure of the entity filters supported by the command line xml config file is as follows:

```
<entityfilters exclusionrule="ExcludeAlways|ExcludeIfPossible">
  <filter entitytype="object_type" groupoperator="And|Or">
    <group conditionoperator="And|Or" includematches="
true|false">
      <condition casesensitive="true|false" namecondition="
true|false">
        <conditiontype>condition_type</conditiontype>
        <expression>expression</expression>
      </condition>
    </group>
  </filter>
</entityfilters>
```

xml Element /Attribute	Description	Usage
EntityFilters	The starting element of the entity filters.	
ExclusionRule: <i>ExcludeAlways</i> <i> ExcludeIfPossible</i>	<p>Determines the rule that should be used to exclude objects that don't meet the filter criteria with respect to dependencies:</p> <ul style="list-style-type: none"> • ExcludeAlways indicates that objects should be excluded regardless of dependencies. • ExcludeIfPossible indicates that objects should be excluded, but database dependencies might include them if needed. 	Optional. Default value is <i>ExcludeAlways</i> .
Filter	Starts a filter element. Multiple filters are supported, each for a specific object type.	
EntityType	<p>Indicates the type of the database objects associated with the filter. Nearly all database are supported, except for the following:</p> <ul style="list-style-type: none"> • Filegroups • Full-Text Indexes • Permissions • Extended Properties 	Required.
GroupOperator : <i>And Or</i>	Determines the operator that combines the groups that participate in the filter.	Optional. Default value is <i>And</i> .

Group	Starts a filter group. Multiple groups can be included in a single filter.	Optional when the filter contains only one condition, otherwise required.
ConditionOperator: <i>And/Or</i>	Determines the operator that combines filter conditions associated with a group.	Optional. Default value is <i>Or</i> .
IncludeMatches: <i>true/false</i>	Indicates whether the database objects that meet the group criteria should be included in the comparison.	Optional. Default value is <i>true</i> .
Condition	Starts a filter condition.	Required.
CaseSensitive: <i>true/false</i>	Indicates whether the condition should honor the character case.	Optional. Default value is <i>false</i> .
NameCondition: <i>true/false</i>	Determines whether the filter condition should be evaluated against the name of a database object. If false, the condition is checked against the schema.	Optional. Default value is <i>true</i> .
ConditionType	One of the following condition types: <ul style="list-style-type: none"> • StartingWith • NotStartingWith • EndingWith • NotEndingWith • Containing • NotContaining • EqualsTo • DifferentFrom • RegExMatching • RegExNotMatching 	Required.
Expression	A text expression that defines the condition criteria.	Required.

The following xml fragment is a more complete example that defines three entity filters:

- The first filter creates two conditions that select database views starting with v_emp or v_cust. Both conditions are case-sensitive.
- The second filter selects stored procedures that belong to HumanResource or Personal schema. In addition to schema, the procedure name should contain at least one digit.
- The third filter selects functions that do not end with _temp and _tmp

```

<entityfilters exclusionrule="ExcludeAlways">
  <filter entitytype="View">
    <group conditionoperator="Or">
      <condition casesensitive="true">
        <conditiontype>StartingWith</conditiontype>
        <expression>v_emp</expression>
      </condition>
      <condition casesensitive="true">
        <conditiontype>StartingWith</conditiontype>
        <expression>v_cust</expression>
      </condition>
    </group>
  </filter>
  <filter entitytype="StoredProcedure" groupoperator="And">
    <group conditionoperator="Or">
      <condition namecondition="false">
        <conditiontype>EqualsTo</conditiontype>
        <expression>HumanResource</expression>
      </condition>
      <condition namecondition="false">
        <conditiontype>EqualsTo</conditiontype>
        <expression>Personal</expression>
      </condition>
    </group>
    <group>
      <condition namecondition="true">
        <conditiontype>RegExMatching</conditiontype>
        <expression>\d+</expression>
      </condition>
    </group>
  </filter>
  <filter entitytype="UserDefinedFunction">
    <group conditionoperator="And">
      <condition>
        <conditiontype>NotEndingWith</conditiontype>
        <expression>_temp</expression>
      </condition>
      <condition>
        <conditiontype>NotEndingWith</conditiontype>
        <expression>_tmp</expression>
      </condition>
    </group>
  </filter>
</entityfilters>

```

IDERA Website	Products	Purchase	Support	Community	About Us	Resources	Legal
-------------------------------	--------------------------	--------------------------	-------------------------	---------------------------	--------------------------	---------------------------	-----------------------