SQL Schema Compare Schema Filters

Schema filters allow you to include or exclude database objects based on matching criteria. You can exclude, for example, tables the name of which start with "tmp_" or include views that belong to the schema HumanResources.

> Schema filters are depreciated and will be removed in a future version of the command line utility. Use entity filters instead.

A schema filter affects only the objects that belong to the class of objects for which the filter is defined. A table filter, for example, affects tables only. When a schema filter is defined, the command line performs one of the following actions:

- If the filter is defined as an "Include" filter, objects that pass the filter criteria are included; all other objects that do not pass the filter criteria are excluded.
- If the filter is defined as an "Exclude" filter, objects that do NOT pass the filter criteria are included; all other objects that pass the filter criteria are excluded.

The command line provides two elements for defining schema filters:

- Simple filters that contain just a few criteria can be defined under the element <SchemaFilters>
- Complex filter with multiple criteria, different validation types and specific case-sensitivity can be defined under the element <SchemaFilterExpressions>

Simple filters are defined as <SchemaFilter> under <SchemaFilters> elements with the following attributes:

- **Criteria** Contains the filter matching criteria.
- **ObjectType** Indicates the type of object on which the filter will be applied.
- **Action="Exclude|Include"** Indicates whether the object that satisfies the filter criteria should be included or excluded.
- **FilterType** Indicates the filter type which can be one of the following:
  - StartsWith
  - EndsWith
  - Contains
  - EqualsTo
  - Schema
  - RegularExpression

The following xml fragment adds a filter that excludes all tables whose name ends with "_tmp":

```
<SchemaFilters>
     <SchemaFilter Criteria="_tmp" FilterType="EndsWith"
ObjectType="Table" Action="Exclude"/>
</SchemaFilters>
```

Schema filters defined under <SchemaFilters> element are sufficient in most comparison scenarios. It you need more control over the filter criteria, you can define the filters as <SchemaFilterExpression> under the <SchemaFilterExpressios> element. The <SchemaFilterExpression> element contains two attributes:

- **ObjectType** Indicates the type of object on which the filter should be applied
- **FilterValidationType="AllCriteria|AtLeastOneCriteria"** Indicates the type of validation that should be performed when the filter contains multiple criteria. If AllCriteria is specified, an object must pass all criteria in order to satisfy the filter; otherwise one criteria is sufficient. If not specified the validation is set by default to AtLeastOneCriteria.

Under the <SchemaFilterExpression> element, you can defined one or multiple filter criteria. Each criteria is a <SchemaFilterCriteria> element with the following attributes:

- **Criteria** Contains the filter criteria
- **Action="Exclude|Include"** Indicates whether the object that satisfies the filter criteria should be included or excluded.
- **FilterType** Indicates the filter type and can be one of the followings:
  - **StartsWith**
  - **EndsWith**
  - **Contains**
  - **EqualsTo**
  - **Schema**
  - **RegularExpression**
  - **IgnoreCase="true|false"** Indicates whether the character case should be ignored when the criteria is applied. By default the filter ignores the case.

The following xml fragment defines two filters: one for views and one for stored procedures. The view filter has three criteria. It selects only views that start with v1_, v2_ or v3_. The procedure filter has two criteria. It excludes procedures the name of which contain _p1_ AND _p2_. The first criteria of the procedure filter is case-sensitive:

```
<SchemaFilterExpressions>
     <SchemaFilterExpression ObjectType="View">
          <SchemaFilterCriteria Action="Include" FilterType="
StartsWith" Criteria="v1_" />
          <SchemaFilterCriteria Action="Include" FilterType="
StartsWith" Criteria="v2_" />
          <SchemaFilterCriteria Action="Include" FilterType="
StartsWith" Criteria="v3_" />
     <SchemaFilterExpression>
          <SchemaFilterExpression ObjectType="StoredProcedure"
FilterValidationType="AllCriteria">
          <SchemaFilterCriteria Action="Exclude" FilterType="
Contains" Criteria="_p1_" IgnoreCase="false" />
          <SchemaFilterCriteria Action="Exclude" FilterType="
Contains" Criteria="_p2_" />
     </SchemaFilterExpression>
</SchemaFilterExpressions>
```

Total compare from data to schema with IDERA SQL Comparison Toolset. Learn more > >

IDERA Website	Products	Purchase	Support	Community	About Us	Resources	Legal