

Custom scripts on OLAP objects in timeXtender

Contents

- Custom scripts on OLAP objects in timeXtender 1
- What is XMLA? 3
- Supported XMLA scripting options in timeXtender 3
 - Pre- or Post steps 3
 - Inline Scripts 3
- Inline Script syntax 3
 - Replace 3
 - Sample: Change the setting of the unknownmember property on a dimension 3
 - Insert Before 4
 - Sample: Insert description before the ID node of a cube 4
 - Insert After 4
 - Sample: Insert description after the ID node of a cube 4
 - Insert End 4
 - Sample: Add drillthrough actions to a cube 4
- Multiple commands in same script 5

What is XMLA?

XML for Analysis (XMLA) is the industry standard for discovering and manipulating data in analytical systems, such as SQL Server Analysis Services (SSAS). XMLA can i.e. be used to create, modify and delete objects during deployment and to process a dimension or cube during execution.

The XMLA language doesn't natively allow manipulation of detailed objects such as the attributes of a dimension or the actions of a cube. It only allows manipulation of entire major objects such as cubes or dimensions.

In addition to XMLA scripting on major objects, timeXtender supports adding or replacing parts of a script for more flexible XMLA scripting.

Supported XMLA scripting options in timeXtender

In timeXtender you can add **Script Actions** on the Cube tab. These script actions can be considered as "script snippets" that can be used either pre- or post the normal process of deploying or executing against Analysis Services.

Pre- or Post steps

This is the equivalent of scripting in SQL Server Management Studio and applies to major objects only.

This can be applied to both deployment and execution on OLAP server, cube and dimension objects.

Inline Scripts

The inline scripts allows you to manipulate the XMLA script that is the the result after the normal deployment process against Analysis Services.

This can be applied to deployment only on OLAP server, cube and dimension objects.

Inline Script syntax

The following commands are available to allow replacement and additions to be made to XMLA scripts.

Replace

Replaces an entire node in the XMLA script with the script snippet

```
<Replace ReplaceNode="Node">
  [Code Snippet Here]
</Replace>
```

Sample: Change the setting of the unknownmember property on a dimension

```
<Replace ReplaceNode="UnknownMember">
  <UnknownMember>Hidden</UnknownMember>
</Replace>
```

Insert Before

Inserts the script snippet immediately before the specified node (same level)

```
<InsertBefore BeforeNode="Node">
  [Code Snippet Here]
</InsertBefore>
```

Sample: Insert description before the ID node of a cube

```
<InsertBefore BeforeNode="ID">
  <Description>Inserted before description</Description>
</InsertBefore>
```

Insert After

Inserts the script snippet immediately after the specified node (same level)

```
<InsertAfter AfterNode="Node">
  [Code Snippet Here]
</InsertAfter>
```

Sample: Insert description after the ID node of a cube

```
<InsertAfter AfterNode="ID">
  <Description>Inserted description After ID</Description>
</InsertAfter>
```

Insert End

Inserts the script snippet at the end (as the last child) of the node given as parameter.

```
<InsertEnd Node="Node">
  [Code Snippet Here]
</InsertEnd>
```

Sample: Add drillthrough actions to a cube

```
<InsertEnd Node="Cube">
  <Actions xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <Action xsi:type="DrillThroughAction">
      <ID>Drillthrough Action</ID>
      <Name>Drillthrough Action</Name>
      <TargetType>Cells</TargetType>
      <Type>DrillThrough</Type>
      <Columns>
        <Column xsi:type="MeasureBinding">
          <MeasureID>61d8fc97-db40-4444-82af-9bb48bc98aa9</MeasureID>
        </Column>
        <Column xsi:type="MeasureBinding">
          <MeasureID>ade4605f-35df-44fd-b5e4-9bb48bd068e3</MeasureID>
        </Column>
        <Column xsi:type="CubeAttributeBinding">
          <CubeID>Sales</CubeID>
          <CubeDimensionID>Product</CubeDimensionID>
          <AttributeID>Product ID</AttributeID>
          <Type>All</Type>
        </Column>
      </Columns>
    </Action>
  </Actions>
</InsertEnd>
```

```

    </Column>
    <Column xsi:type="CubeAttributeBinding">
      <CubeID>Sales</CubeID>
      <CubeDimensionID>Product</CubeDimensionID>
      <AttributeID>Product Type</AttributeID>
      <Type>All</Type>
    </Column>
  </Columns>
</Action>
</Actions>
</InsertEnd>

```

Multiple commands in same script

It is possible to include multiple commands in the same script by wrapping the commands in a script node as seen in the sample below, which includes both a **Replace** and **InsertEnd** command

```

<Script>
  <Replace ReplaceNode="UnknownMember">
    <UnknownMember>Hidden</UnknownMember>
  </Replace>
  <InsertEnd Node="Attribute">
    <DefaultMember>[Time].[YMD].[Year].&[2005].&[1].&[2005-01-01T00:00:00]</DefaultMember>
  </InsertEnd>
</Script>

```