

Table of Contents

[Biml Documentation](#)

[Biml API Reference](#)

[Overview](#)

[Biml Language Reference](#)

[Biml Language Reference](#)

[Biml Release Notes](#)

[Release Notes](#)

[Biml Snippets](#)

[BimlScripts](#)

[Callable BimlScript \(Callee\)](#)

[Callable BimlScript \(Caller\)](#)

[Conditionals](#)

[No Op](#)

[Extract All Tables](#)

[Get Live Database Schema](#)

[Import Database Assets](#)

[Includes](#)

[Setting Variable Value](#)

[Connections](#)

[ADO.NET](#)

[Analysis Services](#)

[Cache](#)

[Excel OleDb](#)

[File](#)

[Flat File](#)

[FTP](#)

[HTTP](#)

[MSMQ](#)

[OleDb](#)

[SMTP](#)

[SQL Server Management](#)

[WMI](#)

[File Formats](#)

Flat File Format

Raw File Format

Cubes

Drillthrough Action

Report Action

Standard Action

Dimension / Hierarchy

Dimension Binding

KPI

MDX Script

Measure Group / Measure

Perspective

Partition - Incremental Notification

Partition - Table Notification

Partition - Query Source

Aggregation Design

Calculated Member

Named Set

Script Command

Miscellaneous

Log Event

Log Provider

Principal

Script Projects

Script Component Project

Script Task Project

SSIS Tasks

Bulk Insert

Data Profiling

Execute Package

Execute SQL

Execute Process

File System

Execute SQL Parameters

Foreach File Loop

FTP

Message Queue

Precedence Constraints

WMI Event Watcher

Script (Simple)

Send Mail

Dataflow

WMI Data Reader

Transformations

Aggregate

Audit

Character Map

Conditional Split

Copy Column

Data Conversion

Derived Columns

Export Column

Fuzzy Lookup

Fuzzy Grouping

Import Column

Lookup

Merge

Merge Join

Multicast

Percentage Sampling

Pivot

Row Count

Row Sampling

Script Component Source

Sort

Term Extraction

Term Lookup

Union All

Unpivot

Tables

Dimension

Fact

Clone Table

Columns

[Multiple Column Table Reference](#)

[Keys](#)

[Foreign Key](#)

[Foreign Key - Simple](#)

[Indexes](#)

[Schema](#)

[Static Source](#)

[Custom Extension](#)

Biml Documentation

Biml API Reference

[Biml API Reference](#)

Biml Language Reference

[Biml Language Reference](#)

Biml Release Notes

[Biml Release Notes](#)

Biml Snippets

The Biml Snippets collection includes a variety of Biml snippets for implementing assets in your Biml-based BI solution.

Biml Forum

If you have questions about a snippet or are looking for help with a particular Biml implementation, you can search and post in the [Biml forum](#) for help.

Email Support

While the forums are the best way to get answers, if your question contains proprietary, or other, information that you would prefer not to post publicly, private email support is available at support@varigence.com

Resources

Learn Biml on [BimlScript.com](#)

[Biml Language Reference](#)

[Biml API Reference](#)

Business Intelligence Markup Language (Biml) API

Business Intelligence Markup Language (Biml) API official reference documentation

If you have questions about a snippet or are looking for help with a particular Biml implementation, you can search and post in the [Biml forum](#) for help.

Email Support

While the forums are the best way to get answers, if your question contains proprietary, or other, information that you would prefer not to post publicly, private email support is available at support@varigence.com

Resources

Learn Biml on [BimlScript.com](#)

[Biml Language Reference](#)

Business Intelligence Markup Language (Biml) API

Business Intelligence Markup Language (Biml) API official reference documentation

If you have questions about a snippet or are looking for help with a particular Biml implementation, you can search and post in the [Biml forum](#) for help.

Email Support

While the forums are the best way to get answers, if your question contains proprietary, or other, information that you would prefer not to post publicly, private email support is available at support@varigence.com

Resources

Learn Biml on [BimlScript.com](#)

[Biml Language Reference](#)

Business Intelligence Markup Language (Biml) Reference

Business Intelligence Markup Language (Biml) official Language reference documentation

If you have questions about a snippet or are looking for help with a particular Biml implementation, you can search and post in the [Biml forum](#) for help.

Email Support

While the forums are the best way to get answers, if your question contains proprietary, or other, information that you would prefer not to post publicly, private email support is available at support@varigence.com

Resources

Learn Biml on [BimlScript.com](#)

[Biml API Reference](#)

Business Intelligence Markup Language (Biml) Reference

Business Intelligence Markup Language (Biml) official Language reference documentation

If you have questions about a snippet or are looking for help with a particular Biml implementation, you can search and post in the [Biml forum](#) for help.

Email Support

While the forums are the best way to get answers, if your question contains proprietary, or other, information that you would prefer not to post publicly, private email support is available at support@varigence.com

Resources

Learn Biml on BimlScript.com

[Biml API Reference](#)

Biml Release Notes

If you have questions about a snippet or are looking for help with a particular Biml implementation, you can search and post in the [Biml forum](#) for help.

Email Support

While the forums are the best way to get answers, if your question contains proprietary, or other, information that you would prefer not to post publicly, private email support is available at support@varigence.com

Resources

Learn Biml on [BimlScript.com](#)

[Biml Language Reference](#)

[Biml API Reference](#)

Biml Release Notes

If you have questions about a snippet or are looking for help with a particular Biml implementation, you can search and post in the [Biml forum](#) for help.

Email Support

While the forums are the best way to get answers, if your question contains proprietary, or other, information that you would prefer not to post publicly, private email support is available at support@varigence.com

Resources

Learn Biml on [BimlScript.com](#)

[Biml Language Reference](#)

[Biml API Reference](#)

Biml Documentation

Biml Snippets

The Biml Snippets collection includes a variety of Biml snippets for implementing assets in your Biml-based BI solution.

Biml Forum

If you have questions about a snippet or are looking for help with a particular Biml implementation, you can search and post in the [Biml forum](#) for help.

Email Support

While the forums are the best way to get answers, if your question contains proprietary, or other, information that you would prefer not to post publicly, private email support is available at support@varigence.com

Resources

Learn Biml on [BimlScript.com](#)

[Biml Language Reference](#)

[Biml API Reference](#)

BimlScripts

Biml is our XML based language for modeling a BI solution. BimlScript allows you to embed C# or VB.NET code into Biml, in the same way that ASP.NET works with HTML.

The BimlScript snippets demonstrate scenarios where BimlScript can solve otherwise cumbersome tasks. To aid in writing your own BimlScripts, check out the [Biml API reference](#), which documents every property, method, and event available to each .NET class.

Callable BimlScript (Callee)

```
<#@ property name="tableName" type="String" #>
<#@ property name="lateArriving" type="System.Boolean" #>
<#
    string lateArrivingString = "(LA)";
    string notLateArrivingString = "(not LA)";
#>
<Package Name="Building from <#=tableName#> <#= lateArriving ? lateArrivingString : notLateArrivingString #>"
ConstraintMode="Linear" />
```

Callable BimlScript (Caller)

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Packages>
    <#foreach (var table in RootNode.Tables) { #>
      <#=CallBimlScript("../BimlScripts\\Callee.biml", table.Name, table.LateArriving) #>
      <# } #>
    </Packages>
  </Biml>
```


Conditionals

```
<#@ template language="C#" hostspecific="True" mergemode="LocalMerge" #>
<#@ import namespace="Varigence.Hadron.Extensions.DataTransformation" #>

<Container>
  <Tasks>
    <ExecuteSQL Name="Count Rows Inserted" ConnectionName="MyStagingDb" BypassPrepare="false"
ResultSet="SingleRow">
      <Expressions>
        <#if (TargetNode.GetTag("TableType") == "Fact") || TargetNode.GetTag("TableType") == "Dim") {#>
          <Expression PropertyName="SqlStatementSource">"SELECT DISTINCT COUNT ('WarehouseId') As
RowsInserted FROM MyTable</Expression>
          <#} else {#>
            <Expression PropertyName="SqlStatementSource">"SELECT COUNT(1) AS RowsInserted FROM
MyTable</Expression>
            <#}#>
          </Expressions>
        </ExecuteSQL>
      </Tasks>
    </Container>
```

No Op

```
<#@ template language="C#" hostspecific="True" mergemode="LocalMerge" #>
<#@ import namespace="Varigence.Hadron.Extensions.DataTransformation" #>

<!-- If Framework Transformer returns null or string.Empty then the transformer has no
      effect on the target node. This snippet shows an example of doing a check to see if
      the framework transformer should be applied to this target node. -->

<#
if (TargetNode.Annotations["Exclude"] != null)
{
    return null;
}
#>

<!-- The rest of your framework transformer code here -->
```

Extract All Tables

```
<#@ template language="C#" hostspecific="true" #>
<#@ import namespace="System.Data" #>

<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Packages>
    <!-- Creates a new package for each table in the project -->
    <# foreach (var table in RootNode.Tables) { #>
      <Package Name="Extract <#=table.Name#" ConstraintMode="Linear"
AutoCreateConfigurationsType="None">
        <Tasks>
          <Dataflow Name="Copy Data">
            <Transformations>
              <OleDbSource Name="Retrieve Data" ConnectionName="Source">
                <DirectInput>SELECT * FROM <#=table.Name#></DirectInput>
              </OleDbSource>
              <OleDbDestination Name="Insert Data" ConnectionName="Target">
                <ExternalTableOutput Table="<#=table.Name#" />
              </OleDbDestination>
            </Transformations>
          </Dataflow>
        </Tasks>
      </Package>
    <# } #>

    <!-- Creates a new package that executes each of the packages created above -->
    <Package Name="Driver - Extract All" ConstraintMode="Linear" AutoCreateConfigurationsType="None">
      <Tasks>
        <# foreach (var table in RootNode.Tables) { #>
          <ExecutePackage Name="Extract <#=table.Name#">
            <Package PackageName="Extract <#=table.Name#" />
          </ExecutePackage>
        <# } #>
      </Tasks>
    </Package>
  </Packages>
</Biml>
```

Get Live Database Schema

```
<#@ template language="C#" hostspecific="true"#>
<#@ import namespace="System.Data" #>

<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Packages>
    <!-- Connects to a live database, using standard ADO.NET, to create a set of packages
         that correspond to the tables in that database. -->
    <# var connectionStringBuilder = new
System.Data.SqlClient.SqlConnectionStringBuilder("Server=localhost;Initial Catalog=Master;Integrated
Security=SSPI;");

        var con = new System.Data.SqlClient.SqlConnection(connectionStringBuilder.ConnectionString);
        con.Open();
        var tblDatabases = con.GetSchema("Databases", new string[] { "SportsData" });

        foreach (System.Data.DataRow database in tblDatabases.Rows)
        {
            connectionStringBuilder.InitialCatalog = database["database_name"].ToString();

            var conTbl = new
System.Data.SqlClient.SqlConnection(connectionStringBuilder.ConnectionString);
            conTbl.Open();

            var tblTables =
conTbl.GetSchema(System.Data.SqlClient.SqlClientMetaDataCollectionNames.Tables);

            foreach (System.Data.DataRow table in tblTables.Rows)
            { #>
                <Package Name="<#=table["table_name"]#>">
                    <!-- Your logic here -->
                </Package>
            <# }

            conTbl.Close();
        }

        con.Close(); #>
    </Packages>
</Biml>
```

Import Database Assets

```
<!-- This sample demonstrates BimlScript that imports schemas and tables using a connection to
AdventureWorksLT. -->

<#@ template language="C#" hostspecific="True"#>
<#@ import namespace="Varigence.Languages.Biml.Connection" #>
<#@ import namespace="Varigence.Hadron.Extensions" #>
<#@ import namespace="Varigence.Hadron.Extensions.SchemaManagement" #>

<#+ public ImportResults Results
{
    get
    {
        return ((AstOleDbConnectionNode)RootNode.Connections["AdventureWorksLT"]).ImportDB();
    }
}
#>

<Biml xmlns="http://schemas.varigence.com/biml.xsd">
    <Schemas>
        <#=Results.SchemaNodes.GetBiml()#>
    </Schemas>
    <Tables>
        <#=Results.TableNodes.GetBiml()#>
    </Tables>
</Biml>
```

Includes

```
<#@ template language="C#" hostspecific="True" mergemode="LocalMerge" #>
<#@ import namespace="System.Data" #>
<#@ import namespace="Varigence.Languages.Biml.Transformation.Destination" #>
<!-- This includes the OurUtility.biml file to this Biml file.
      Including a file does not result in code generation, but brings in references.
      More akin to C# references than C #includes -->
<#@ include file="OurUtility.biml" #>

<!-- Biml/BimlScript code here -->
```

Setting Variable Value

```
<#@ template language="C#" hostspecific="True" mergemode="LocalMerge" addtoheadofcollections="True"#>
<#@ import namespace="Varigence.Languages.Biml.Task" #>

<Container>
  <Variables>
    <!-- Set the value of a variable by using the value from various tags on the TargetNode -->
    <Variable Name="SchemaName" DataType="String" ><#=TargetNode.GetTag("SchemaName")#></Variable>
    <Variable Name="TableName" DataType="String" ><#=TargetNode.GetTag("TableName")#></Variable>
    <Variable Name="ContainerName" DataType="String" ><#=TargetNode.Name#></Variable>
  </Variables>
</Containers>
```

Connections

Biml replaces connection managers with named connection objects that can be referenced by other elements. This enables easy reuse. Furthermore, if a connection needs to change, you only need to change it in one place.

Biml includes connection objects for standard connection types, including ADO.NET, Excel, FTP, HTTP, MSMQ, OLE DB, WMI, and others.

ADO.NET

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Connections>
    <AdoNetConnection Name="AdoNetConnection" Provider="System.Data.SqlClient.SqlConnection, System.Data,
Version=2.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089" ConnectionString="Data
Source=localhost;Initial Catalog=SportsData;Integrated Security=True;" />
  </Connections>
</Biml>
```

Analysis Services

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Connections>
    <!-- This is a sample AnalysisServices connection that connects to a server running an Analysis
Services database. -->
    <AnalysisServicesConnection
      Name="AdvWorksCube"
      Provider="MSOLAP"
      Database="AdventureWorksCubeVulcan"
      Server="localhost"
      ConnectionString="Data Source=localhost;Initial Catalog=Adventure Works DW
2008;Provider=MSOLAP.4;Impersonation Level=Impersonate;"
    />
  </Connections>
</Biml>
```

Cache

```
<Biml xmlns:Pamlico="http://schemas.mariner.com/pamlico" xmlns="http://schemas.varigence.com/biml.xsd">
  <!-- Cache connection file that uses MyRawFileFormat as its file format -->
  <Connections>
    <CacheConnection Name="MyCacheConnection" RawFileFormatName="MyRawFileFormat" />
  </Connections>
</Biml>
```

Excel OleDb

```
<Biml xmlns:Pamlico="http://schemas.mariner.com/pamlico" xmlns="http://schemas.varigence.com/biml.xsd">
  <!-- Creates a connection to an Excel 2007 xlsx file. -->
  <Connections>
    <ExcelConnection Name="ExcelOleDbConnection" ConnectionString="Provider=Microsoft.ACE.OLEDB.12.0;Data
Source=c:\MyExcel2007file.xlsx;Extended Properties='Excel 12.0 Xml;HDR=YES';" />
  </Connections>
</Biml>
```

File

```
<Biml xmlns:Pamlico="http://schemas.mariner.com/pamlico" xmlns="http://schemas.varigence.com/biml.xsd">
  <!-- Creates a connection to a file with the absolute path c:\myfile.txt. -->
  <Connections>
    <FileConnection Name="MyFileConnection" FilePath="C:\myfile.txt" RelativePath="true"
FileUsageType="CreateFile" />
  </Connections>
</Biml>
```

Flat File

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Connections>
    <FlatFileConnection Name="FlatFileConnection" FileFormat="FlatFileFormatSurvey"
FilePath="c:\users\me\documents\myFile.txt" />
  </Connections>
</Biml>
```

FTP

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <!-- Creates an FTP connection to an FTP server at theFtpServer.com. Note the password in this file is in
plaintext,
      so treat the file with care if placing the password in it. -->
  <Connections>
    <FtpConnection Name="MyFtpConnection" Password="p@ssw0rd" UserName="myUserName"
ServerName="theFtpServer.com" />
  </Connections>
</Biml>
```

HTTP

```
<Biml xmlns:Pamlico="http://schemas.mariner.com/pamlico" xmlns="http://schemas.varigence.com/biml.xsd">
  <!-- Creates an HTTP connection with a certificate. Note the password in this file is in plaintext. -->
  <Connections>
    <HttpConnection Name="MyHttpConnection" ClientCertificateFileName="CertConnection"
Domain="varigence.com" Password="p@ssw0rd" ServerUrl="https://server.varigence.com" UserName="varigence" />
  </Connections>
</Biml>
```


MSMQ

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Connections>
    <MsmqConnection Name="MsmqConnection" Path=".\private$\MyMessageQ" />
  </Connections>
</Biml>
```

OleDb

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <!-- Connection for a simple OleDb Connection to a SQL Server Database. Notice the use of the text for
the connection string. -->
  <Connections>
    <Connection Name="OleDbConnection" ConnectionString="Provider=SQLNCLI10;Server=localhost;Integrated
Security=SSPI;" />
  </Connections>
</Biml>
```

SMTP

```
<Biml xmlns:Pamlico="http://schemas.mariner.com/pamlico" xmlns="http://schemas.varigence.com/biml.xsd">
  <!-- Creates an SMTP connection with SSL -->
  <Connections>
    <SmtpConnection Name="MySmtpConnection" SmtpServer="smtp.varigence.com" EnableSsl="true" />
  </Connections>
</Biml>
```

SQL Server Management

```
<Biml xmlns:Pamlico="http://schemas.mariner.com/pamlico" xmlns="http://schemas.varigence.com/biml.xsd">
  <!-- Represents a connection to a SQL Server instance via SQL Server Management -->
  <Connections>
    <SqlServerManagementConnection Name="SqlServerManagementConnection" SqlServerName="localhost" />
  </Connections>
</Biml>
```

WMI

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Connections>
    <WmiConnection Name="WmiConnection" ServerName="\\localhost" UseWindowsAuthentication="true"
Namespace="\root\cimv2" />
  </Connections>
</Biml>
```

File Formts

Biml allows you to define file formats for storing data in files. Both raw and flat file formats are supported in Biml.

Flat File Format

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <FileFormats>
    <!-- Flat file format that specifies the columns and delimiter used to extract a table from a flat
file. -->
    <FlatFileFormat Name="MetadataFileFormat" RowDelimiter="LF" ColumnNamesInFirstDataRow="true"
IsUnicode="false">
      <Columns>
        <Column Name="Category" DataType="String" Length="128" Delimiter="Tab" CodePage="1252" />
        <Column Name="DisplayName" DataType="String" Length="256" Delimiter="Tab" />
        <Column Name="Maturity" DataType="String" Length="32" Delimiter="Tab" />
        <Column Name="Frequency" DataType="String" Length="32" Delimiter="Tab" />
        <Column Name="Status" DataType="String" Length="32" Delimiter="Tab" />
        <Column Name="CompactName" DataType="String" Length="32" Delimiter="Tab" />
        <Column Name="Url" DataType="String" Length="4000" Delimiter="Tab" />
        <Column Name="HeaderRowsToSkip" Delimiter="Tab" />
        <Column Name="DateGrain" DataType="String" Length="32" Delimiter="LF" />
      </Columns>
    </FlatFileFormat>
  </FileFormats>
</Biml>
```

Raw File Format

```
<Biml xmlns:Pamlico="http://schemas.mariner.com/pamlico" xmlns="http://schemas.varigence.com/biml.xsd">
  <FileFormats>
    <!-- Specifies a raw file format with 3 columns. -->
    <RawFileFormat Name="MyRawFileFormat">
      <Columns>
        <Column Name="ID" IndexPosition="0" />
        <Column Name="CacheSize" DataType="Int32" IndexPosition="1" />
        <Column Name="Description" Length="200" IndexPosition="2" />
      </Columns>
    </RawFileFormat>
  </FileFormats>
</Biml>
```


Cubes

Biml allows you to define cubes, that correspond to SSAS cubes, in a complete yet concise manner. Biml includes support for all cube objects, including actions, perspectives, partitions, and aggregations, and makes it easy to leverage defined project assets within your cube objects.

Drillthrough Action

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Cubes>
    <Cube Name="Adventure Works" ConnectionName="AdvWorksCube">
      <CubeDimensions>
        <CubeDimension Name="Scenario" DimensionName="DimScenario">
          <Attributes>
            <Attribute AttributeName="DimScenario.Scenario" />
          </Attributes>
        </CubeDimension>
      </CubeDimensions>
      <CubeMeasureGroups>
        <CubeMeasureGroup Name="Fact Finance" FactName="FactFinance"/>
      </CubeMeasureGroups>
      <Actions>
        <!--
        This is a sample DrillThrough Action. Its Target references the cube's Fact Finance measure
group.

        Its columns list references measure groups measures and cube dimension attributes.
        -->
        <DrillThrough
          Name="Finance Details"
          Target="Adventure Works.Fact Finance"
          Caption="Drillthrough..."
          Default="true"
          CaptionIsMdx="false"
          Invocation="Interactive"
          TargetType="Cells"
        >
          <Columns>
            <Column>Adventure Works.Date.Fiscal Year</Column>
            <Column>Adventure Works.Date.Date</Column>
            <Column>Adventure Works.Date.Calendar Year</Column>
            <Column>Adventure Works.Account.Accounts</Column>
            <Column>Adventure Works.Organization.Organizations</Column>
            <Column>Adventure Works.Scenario.Scenario</Column>
            <Column>Adventure Works.Department.Departments</Column>
            <Column>Adventure Works.Destination Currency.Destination Currency Code</Column>
          </Columns>
        </DrillThrough>
      </Actions>
    </Cube>
  </Cubes>
</Biml>
```

Report Action

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Cubes>
    <Cube Name="Adventure Works" ConnectionName="AdvWorksCube">
      <CubeDimensions>
        <CubeDimension Name="Product" DimensionName="DimProduct">
          <Attributes>
            <Attribute AttributeName="DimProduct.Category" />
            <Attribute AttributeName="DimProduct.Product" />
            <Attribute AttributeName="DimProduct.Class" />
            <Attribute AttributeName="DimProduct.Color" />
            <Attribute AttributeName="DimProduct.List Price" />
          </Attributes>
        </CubeDimension>
      </CubeDimensions>
      <Actions>
        <!--
This is a sample Report Action that uses report format parameters and report parameters, with
a MDX caption.
Its target is the cube's Product dimension's Category attribute.
-->
        <Report
          Name="Sales Reason Comparisons"
          CaptionIsMdx="true"
          Caption=""Sales Reason Comparisons for " + &#xD;&#xA;[Product].
[Category].CurrentMember.Member_Caption + "..."."
          Condition="// This action requires that both Reporting Services and the Reporting
Services&#xD;&#xA;// sample reports be installed on the local machine."
          ReportServer="Localhost"
          Path="ReportServer?/AdventureWorks Sample Reports/Sales Reason Comparisons"
          TargetType="AttributeMembers"
          Target="Adventure Works.Product.Category"
        >
          <ReportFormatParameters>
            <ReportFormatParameter Name="RSCommand">Render</ReportFormatParameter>
            <ReportFormatParameter Name="RSFormat">Html40</ReportFormatParameter>
          </ReportFormatParameters>
          <ReportParameters>
            <ReportParameter Name="ProductCategory">UrlEscapeFragment( [Product].
[Category].CurrentMember.UniqueName )</ReportParameter>
          </ReportParameters>
        </Report>
      </Actions>
    </Cube>
  </Cubes>
</Biml>
```

Standard Action

```

<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Cubes>
    <Cube Name="Adventure Works" ConnectionName="AdvWorksCube">
      <CubeDimensions>
        <CubeDimension Name="Geography" DimensionName="DimGeography">
          <Attributes>
            <Attribute AttributeName="DimGeography.Geography Key" />
            <Attribute AttributeName="DimGeography.City" />
            <Attribute AttributeName="DimGeography.Country" />
            <Attribute AttributeName="DimGeography.Postal Code" />
            <Attribute AttributeName="DimGeography.State-Province" />
          </Attributes>
          <Hierarchies>
            <Hierarchy HierarchyName="DimGeography.Geography" />
          </Hierarchies>
        </CubeDimension>
      </CubeDimensions>
      <Actions>
        <!--
        This is a sample Standard Action that has a MDX expression and caption.
        Its Target is the cube's Geography dimension's City attribute.
        -->
        <Standard
          Name="City Map"
          Caption=""View Map for " + &#xD;&#xA;[Geography].
[City].CurrentMember.Member_Caption + ";...""
          CaptionIsMdx="true"
          TargetType="AttributeMembers"
          Target="Adventure Works.Geography.City"
        >
          <Expression>// URL for linking to MSN Maps
            "http://maps.msn.com/home.aspx?plce1=" +

            // Retrieve the name of the current city
            [Geography].[City].CurrentMember.Name + "," +

            // Append state-province name
            [Geography].[State-Province].CurrentMember.Name + "," +

            // Append country name
            [Geography].[Country].CurrentMember.Name +

            // Append region paramter
            "&reg1=" +

            // Determine correct region paramter value
            Case
              When [Geography].[Country].CurrentMember Is
                [Geography].[Country].&[Australia]
                Then "3"
              When [Geography].[Country].CurrentMember Is
                [Geography].[Country].&[Canada]
                Or
                [Geography].[Country].CurrentMember Is
                [Geography].[Country].&[United States]
                Then "0"
              Else "1"
            End
          </Expression>
        </Standard>
      </Actions>
    </Cube>
  </Cubes>
</Biml>

```

Dimension / Hierarchy

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Cubes>
    <!-- This is an example cube definition. -->
    <Cube Name="FederalReserve" ConnectionName="FederalReserve">
      <!-- This is the measure group -->
      <CubeMeasureGroups>
        <CubeMeasureGroup Name="Rates" FactName="FactRate">
          <Measures>
            <Measure MeasureName="FactRate.Interest Rate" />
            <Measure MeasureName="FactRate.Instrument Count" />
          </Measures>

          <!-- This is where the dimension binding to the fact table IDs are defined -->
          <CubeDimensionBindings>
            <CubeDimensionBinding CubeDimensionName="Instrument"
FactColumnName="FactRate.InstrumentID" />
            <CubeDimensionBinding CubeDimensionName="Date" FactColumnName="FactRate.DateID" />
          </CubeDimensionBindings>
          <Partitions>
            <Partition Name="Rates">
              <DsvTableSource TableName="FactRate" />
            </Partition>
          </Partitions>
        </CubeMeasureGroup>
      </CubeMeasureGroups>

      <!-- This is the definition of the cube dimensions -->
      <CubeDimensions>
        <CubeDimension Name="Instrument" DimensionName="DimInstrument">
          <Attributes>
            <Attribute AttributeName="DimInstrument.DimInstrumentKey" />
            <Attribute AttributeName="DimInstrument.Instrument" />
            <Attribute AttributeName="DimInstrument.Category" />
            <Attribute AttributeName="DimInstrument.Maturity" />
            <Attribute AttributeName="DimInstrument.Frequency" />
            <Attribute AttributeName="DimInstrument.Status" />
          </Attributes>
          <Hierarchies>
            <Hierarchy HierarchyName="DimInstrument.Instrument Hierarchy" />
          </Hierarchies>
        </CubeDimension>
        <CubeDimension Name="Date" DimensionName="DimDate">
          <Attributes>
            <Attribute AttributeName="DimDate.Date" />
            <Attribute AttributeName="DimDate.CalendarYear" />
            <Attribute AttributeName="DimDate.CalendarSemester" />
            <Attribute AttributeName="DimDate.CalendarQuarter" />
            <Attribute AttributeName="DimDate.CalendarMonth" />
            <Attribute AttributeName="DimDate.CalendarWeek" />
            <Attribute AttributeName="DimDate.DayOfWeek" />
            <Attribute AttributeName="DimDate.IsWeekend" />
            <Attribute AttributeName="DimDate.IsHoliday" />
          </Attributes>
          <Hierarchies>
            <Hierarchy HierarchyName="DimDate.MonthHierarchy" />
            <Hierarchy HierarchyName="DimDate.WeekHierarchy" />
          </Hierarchies>
        </CubeDimension>
      </CubeDimensions>

      <!-- An MdxScript for the cube that defines a calculated member -->
```

```
    <MdxScripts>
      <MdxScript Name="MdxScript" Default="true">
        <MdxScriptItems>
          <CalculatedMember Name="Average Interest Rate" ParentDimensionName="Measures"
FormatString="Percent" AssociatedMeasureGroupName="Rates">
            <NonemptyBehaviors>
              <NonemptyBehavior MeasureName="FactRate.Interest Rate" />
            </NonemptyBehaviors>
            <MdxScript>([Measures].[Interest Rate] / [Measures].[Instrument Count]) /
100</MdxScript>
          </CalculatedMember>
        </MdxScriptItems>
      </MdxScript>
    </MdxScripts>
  </Cube>
</Cubes>
</Biml>
```

Dimension Binding

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Cubes>
    <!-- This is an example cube definition. -->
    <Cube Name="FederalReserve" ConnectionName="FederalReserve">
      <!-- This is the measure group -->
      <CubeMeasureGroups>
        <CubeMeasureGroup Name="Rates" FactName="FactRate">
          <Measures>
            <Measure MeasureName="FactRate.Interest Rate" />
            <Measure MeasureName="FactRate.Instrument Count" />
          </Measures>

          <!-- This is where the dimension binding to the fact table IDs are defined -->
          <CubeDimensionBindings>
            <CubeDimensionBinding CubeDimensionName="Instrument"
FactColumnName="FactRate.InstrumentID" />
            <CubeDimensionBinding CubeDimensionName="Date" FactColumnName="FactRate.DateID" />
          </CubeDimensionBindings>
          <Partitions>
            <Partition Name="Rates">
              <DsvTableSource TableName="FactRate" />
            </Partition>
          </Partitions>
        </CubeMeasureGroup>
      </CubeMeasureGroups>

      <!-- This is the definition of the cube dimensions -->
      <CubeDimensions>
        <CubeDimension Name="Instrument" DimensionName="DimInstrument">
          <Attributes>
            <Attribute AttributeName="DimInstrument.DimInstrumentKey" />
            <Attribute AttributeName="DimInstrument.Instrument" />
            <Attribute AttributeName="DimInstrument.Category" />
            <Attribute AttributeName="DimInstrument.Maturity" />
            <Attribute AttributeName="DimInstrument.Frequency" />
            <Attribute AttributeName="DimInstrument.Status" />
          </Attributes>
          <Hierarchies>
            <Hierarchy HierarchyName="DimInstrument.Instrument Hierarchy" />
          </Hierarchies>
        </CubeDimension>
        <CubeDimension Name="Date" DimensionName="DimDate">
          <Attributes>
            <Attribute AttributeName="DimDate.Date" />
            <Attribute AttributeName="DimDate.CalendarYear" />
            <Attribute AttributeName="DimDate.CalendarSemester" />
            <Attribute AttributeName="DimDate.CalendarQuarter" />
            <Attribute AttributeName="DimDate.CalendarMonth" />
            <Attribute AttributeName="DimDate.CalendarWeek" />
            <Attribute AttributeName="DimDate.DayOfWeek" />
            <Attribute AttributeName="DimDate.IsWeekend" />
            <Attribute AttributeName="DimDate.IsHoliday" />
          </Attributes>
          <Hierarchies>
            <Hierarchy HierarchyName="DimDate.MonthHierarchy" />
            <Hierarchy HierarchyName="DimDate.WeekHierarchy" />
          </Hierarchies>
        </CubeDimension>
      </CubeDimensions>

      <!-- An MdxScript for the cube that defines a calculated member -->
```



```
    <MdxScripts>
      <MdxScript Name="MdxScript" Default="true">
        <MdxScriptItems>
          <CalculatedMember Name="Average Interest Rate" ParentDimensionName="Measures"
FormatString="Percent" AssociatedMeasureGroupName="Rates">
            <NonemptyBehaviors>
              <NonemptyBehavior MeasureName="FactRate.Interest Rate" />
            </NonemptyBehaviors>
            <MdxScript>([Measures].[Interest Rate] / [Measures].[Instrument Count]) /
100</MdxScript>
          </CalculatedMember>
        </MdxScriptItems>
      </MdxScript>
    </MdxScripts>
  </Cube>
</Cubes>
</Biml>
```

KPI

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Cubes>
    <Cube Name="Adventure Works" ConnectionName="AdvWorksCube">
      <CubeMeasureGroups>
        <CubeMeasureGroup Name="Fact Finance" FactName="FactFinance"/>
      </CubeMeasureGroups>
      <Kpis>
        <!--
        This is a sample KPI that has MDX expressions for Goal, Status, Trend, and Value.
        Its AssociatedMeasureGroupName property references the cube's Fact Finance measure group.
        -->
        <Kpi
          Name="Operating Profit"
          ParentKpiName="Net Income"
          AssociatedMeasureGroupName="Fact Finance"
          TrendGraphic="StatusArrow"
          StatusGraphic="TrafficLight"
        >
          <Goal>
            // Account 48 = Operating Profit | Scenario 2 = Budget
            ( [Account].[Accounts].&[48], [Scenario].[Scenario].&[2], [Measures].[Amount]
          )

          </Goal>
          <Status>
            Case
            When KpiValue( "Operating Profit" ) / KpiGoal( "Operating Profit" ) >= .95
            Then 1
            When KpiValue( "Operating Profit" ) / KpiGoal( "Operating Profit" ) < .95
            And
            KpiValue( "Operating Profit" ) / KpiGoal( "Operating Profit" ) >= .85
            Then 0
            Else -1
            End
          </Status>
          <Trend>
            Case
            When (
              KpiValue( "Operating Profit" )
              -
              (
                KpiValue( "Operating Profit" ),
                ParallelPeriod
                (
                  [Date].[Fiscal].[Fiscal Year],
                  1,
                  [Date].[Fiscal].CurrentMember
                )
              )
              /
              (
                KpiValue( "Operating Profit" ),
                ParallelPeriod
                (
                  [Date].[Fiscal].[Fiscal Year],
                  1,
                  [Date].[Fiscal].CurrentMember
                )
              ) > .03
            Then 1
            Else -1
          </Trend>
        </Kpi>
      </Kpis>
    </Cube>
  </Cubes>
</Biml>
```

```
        End
    </Trend>
    <Value>
        // Account 48 = Operating Profit | Scenario 1 = Actual
        ( [Account].[Accounts].&[48], [Scenario].[Scenario].&[1], [Measures].[Amount]
    )
    </Value>
</Kpi>
</Kpis>
</Cube>
</Cubes>
</Biml>
```

MDX Script

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Cubes>
    <!-- This is an example cube definition. -->
    <Cube Name="FederalReserve" ConnectionName="FederalReserve">
      <!-- This is the measure group -->
      <CubeMeasureGroups>
        <CubeMeasureGroup Name="Rates" FactName="FactRate">
          <Measures>
            <Measure MeasureName="FactRate.Interest Rate" />
            <Measure MeasureName="FactRate.Instrument Count" />
          </Measures>

          <!-- This is where the dimension binding to the fact table IDs are defined -->
          <CubeDimensionBindings>
            <CubeDimensionBinding CubeDimensionName="Instrument"
FactColumnName="FactRate.InstrumentID" />
            <CubeDimensionBinding CubeDimensionName="Date" FactColumnName="FactRate.DateID" />
          </CubeDimensionBindings>
          <Partitions>
            <Partition Name="Rates">
              <DsvTableSource TableName="FactRate" />
            </Partition>
          </Partitions>
        </CubeMeasureGroup>
      </CubeMeasureGroups>

      <!-- This is the definition of the cube dimensions -->
      <CubeDimensions>
        <CubeDimension Name="Instrument" DimensionName="DimInstrument">
          <Attributes>
            <Attribute AttributeName="DimInstrument.DimInstrumentKey" />
            <Attribute AttributeName="DimInstrument.Instrument" />
            <Attribute AttributeName="DimInstrument.Category" />
            <Attribute AttributeName="DimInstrument.Maturity" />
            <Attribute AttributeName="DimInstrument.Frequency" />
            <Attribute AttributeName="DimInstrument.Status" />
          </Attributes>
          <Hierarchies>
            <Hierarchy HierarchyName="DimInstrument.Instrument Hierarchy" />
          </Hierarchies>
        </CubeDimension>
        <CubeDimension Name="Date" DimensionName="DimDate">
          <Attributes>
            <Attribute AttributeName="DimDate.Date" />
            <Attribute AttributeName="DimDate.CalendarYear" />
            <Attribute AttributeName="DimDate.CalendarSemester" />
            <Attribute AttributeName="DimDate.CalendarQuarter" />
            <Attribute AttributeName="DimDate.CalendarMonth" />
            <Attribute AttributeName="DimDate.CalendarWeek" />
            <Attribute AttributeName="DimDate.DayOfWeek" />
            <Attribute AttributeName="DimDate.IsWeekend" />
            <Attribute AttributeName="DimDate.IsHoliday" />
          </Attributes>
          <Hierarchies>
            <Hierarchy HierarchyName="DimDate.MonthHierarchy" />
            <Hierarchy HierarchyName="DimDate.WeekHierarchy" />
          </Hierarchies>
        </CubeDimension>
      </CubeDimensions>

      <!-- An MdxScript for the cube that defines a calculated member -->
```

```
    <MdxScripts>
      <MdxScript Name="MdxScript" Default="true">
        <MdxScriptItems>
          <CalculatedMember Name="Average Interest Rate" ParentDimensionName="Measures"
FormatString="Percent" AssociatedMeasureGroupName="Rates">
            <NonemptyBehaviors>
              <NonemptyBehavior MeasureName="FactRate.Interest Rate" />
            </NonemptyBehaviors>
            <MdxScript>([Measures].[Interest Rate] / [Measures].[Instrument Count]) /
100</MdxScript>
          </CalculatedMember>
        </MdxScriptItems>
      </MdxScript>
    </MdxScripts>
  </Cube>
</Cubes>
</Biml>
```

Measure Group / Measure

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Cubes>
    <!-- This is an example cube definition. -->
    <Cube Name="FederalReserve" ConnectionName="FederalReserve">
      <!-- This is the measure group -->
      <CubeMeasureGroups>
        <CubeMeasureGroup Name="Rates" FactName="FactRate">
          <Measures>
            <Measure MeasureName="FactRate.Interest Rate" />
            <Measure MeasureName="FactRate.Instrument Count" />
          </Measures>

          <!-- This is where the dimension binding to the fact table IDs are defined -->
          <CubeDimensionBindings>
            <CubeDimensionBinding CubeDimensionName="Instrument"
FactColumnName="FactRate.InstrumentID" />
            <CubeDimensionBinding CubeDimensionName="Date" FactColumnName="FactRate.DateID" />
          </CubeDimensionBindings>
          <Partitions>
            <Partition Name="Rates">
              <DsvTableSource TableName="FactRate" />
            </Partition>
          </Partitions>
        </CubeMeasureGroup>
      </CubeMeasureGroups>

      <!-- This is the definition of the cube dimensions -->
      <CubeDimensions>
        <CubeDimension Name="Instrument" DimensionName="DimInstrument">
          <Attributes>
            <Attribute AttributeName="DimInstrument.DimInstrumentKey" />
            <Attribute AttributeName="DimInstrument.Instrument" />
            <Attribute AttributeName="DimInstrument.Category" />
            <Attribute AttributeName="DimInstrument.Maturity" />
            <Attribute AttributeName="DimInstrument.Frequency" />
            <Attribute AttributeName="DimInstrument.Status" />
          </Attributes>
          <Hierarchies>
            <Hierarchy HierarchyName="DimInstrument.Instrument Hierarchy" />
          </Hierarchies>
        </CubeDimension>
        <CubeDimension Name="Date" DimensionName="DimDate">
          <Attributes>
            <Attribute AttributeName="DimDate.Date" />
            <Attribute AttributeName="DimDate.CalendarYear" />
            <Attribute AttributeName="DimDate.CalendarSemester" />
            <Attribute AttributeName="DimDate.CalendarQuarter" />
            <Attribute AttributeName="DimDate.CalendarMonth" />
            <Attribute AttributeName="DimDate.CalendarWeek" />
            <Attribute AttributeName="DimDate.DayOfWeek" />
            <Attribute AttributeName="DimDate.IsWeekend" />
            <Attribute AttributeName="DimDate.IsHoliday" />
          </Attributes>
          <Hierarchies>
            <Hierarchy HierarchyName="DimDate.MonthHierarchy" />
            <Hierarchy HierarchyName="DimDate.WeekHierarchy" />
          </Hierarchies>
        </CubeDimension>
      </CubeDimensions>

      <!-- An MdxScript for the cube that defines a calculated member -->
```

```
    <MdxScripts>
      <MdxScript Name="MdxScript" Default="true">
        <MdxScriptItems>
          <CalculatedMember Name="Average Interest Rate" ParentDimensionName="Measures"
FormatString="Percent" AssociatedMeasureGroupName="Rates">
            <NonemptyBehaviors>
              <NonemptyBehavior MeasureName="FactRate.Interest Rate" />
            </NonemptyBehaviors>
            <MdxScript>([Measures].[Interest Rate] / [Measures].[Instrument Count]) /
100</MdxScript>
          </CalculatedMember>
        </MdxScriptItems>
      </MdxScript>
    </MdxScripts>
  </Cube>
</Cubes>
</Biml>
```

Perspective

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Cubes>
    <Cube Name="Adventure Works" ConnectionName="AdvWorksCube">
      <Perspectives>
        <!--
          This is a sample perspective for this Adventure Works cube. With Biml, it's easy to reference
          cube objects within the perspective.
        -->
        <Perspective Name="Direct Sales" DefaultMeasureName="FactInternetSales.Internet Sales Amount">
          <Actions>
            <Action ActionName="Internet Details" />
          </Actions>
          <Calculations>
            <Calculation CalculationName="[Internet Gross Profit]" />
          </Calculations>
          <Kpis>
            <Kpi KpiName="Growth in Customer Base" />
          </Kpis>
          <MeasureGroups>
            <MeasureGroup CubeMeasureGroupName="Fact Internet Sales 1">
              <Measures>
                <Measure MeasureName="FactInternetSales.Internet Order Quantity" />
                <Measure MeasureName="FactInternetSales.Internet Tax Amount" />
                <Measure MeasureName="FactInternetSales.Internet Freight Cost" />
                <Measure MeasureName="FactInternetSales.Internet Standard Product Cost" />
                <Measure MeasureName="FactInternetSales.Internet Transaction Count" />
                <Measure MeasureName="FactInternetSales.Internet Extended Amount" />
              </Measures>
            </MeasureGroup>
          </MeasureGroups>
          <Dimensions>
            <Dimension CubeDimensionName="Sales Territory">
              <Attributes>
                <Attribute AttributeName="DimSalesTerritory.Sales Territory Region" />
                <Attribute AttributeName="DimSalesTerritory.Sales Territory Country" />
              </Attributes>
              <Hierarchies>
                <Hierarchy HierarchyName="DimSalesTerritory.Sales Territory" />
              </Hierarchies>
            </Dimension>
          </Dimensions>
        </Perspective>
      </Perspectives>
      <MdxScripts>
        <MdxScript Name="MdxScript1" Default="true">
          <MdxScriptItems>
            <CalculatedMember Name="[Internet Gross Profit]" FormatString="Currency"
ParentDimensionName="Measures" AssociatedMeasureGroupName="Fact Internet Sales 1">
              <MdxScript>[Measures].[Internet Sales Amount] - [Measures].[Internet Total Product
Cost]</MdxScript>
            <NonemptyBehaviors>
              <NonemptyBehavior MeasureName="FactInternetSales.Internet Sales Amount" />
              <NonemptyBehavior MeasureName="FactInternetSales.Internet Total Product Cost"
/>
            </NonemptyBehaviors>
          </CalculatedMember>
        </MdxScriptItems>
      </MdxScript>
    </MdxScripts>
  </CubeMeasureGroups>
  <CubeMeasureGroup Name="Fact Internet Sales 1" FactName="FactInternetSales">
```



```

    <Measures>
      <Measure MeasureName="FactInternetSales.Internet Sales Amount" />
      <Measure MeasureName="FactInternetSales.Internet Order Quantity" />
      <Measure MeasureName="FactInternetSales.Internet Extended Amount" />
      <Measure MeasureName="FactInternetSales.Internet Tax Amount" />
      <Measure MeasureName="FactInternetSales.Internet Freight Cost" />
      <Measure MeasureName="FactInternetSales.Internet Unit Price" />
      <Measure MeasureName="FactInternetSales.Internet Total Product Cost" />
      <Measure MeasureName="FactInternetSales.Internet Standard Product Cost" />
      <Measure MeasureName="FactInternetSales.Internet Transaction Count" />
    </Measures>
  </CubeMeasureGroup>
</CubeMeasureGroups>
<CubeDimensions>
  <CubeDimension Name="Sales Territory" DimensionName="DimSalesTerritory">
    <Attributes>
      <Attribute AttributeName="DimSalesTerritory.Sales Territory Region" />
      <Attribute AttributeName="DimSalesTerritory.Sales Territory Country" />
      <Attribute AttributeName="DimSalesTerritory.Sales Territory Group" />
    </Attributes>
    <Hierarchies>
      <Hierarchy HierarchyName="DimSalesTerritory.Sales Territory" />
    </Hierarchies>
  </CubeDimension>
</CubeDimensions>
<Actions>
  <DrillThrough Name="Internet Details" Caption="Drillthrough..." Default="false"
Target="Adventure Works.Fact Internet Sales 1" TargetType="Cells"/>
  <DrillThrough Name="Finance Details" Target="Adventure Works.Fact Finance"
Caption="Drillthrough..." Default="true" CaptionIsMdx="false" Invocation="Interactive" TargetType="Cells"/>
</Actions>
<Kpis>
  <Kpi Name="Growth in Customer Base" AssociatedMeasureGroupName="Fact Internet Sales 1"
StatusGraphic="RoadSigns"/>
</Kpis>
</Cube>
</Cubes>
</Biml>

```

Partition - Incremental Notification

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Cubes>
    <Cube Name="Adventure Works" ConnectionName="AdvWorksCube">
      <CubeMeasureGroups>
        <CubeMeasureGroup Name="Fact Finance" FactName="FactFinance">
          <Partitions>
            <!--
              This partition has proactive caching enabled.
              Incremental updates are also enabled, showing how a polling query or processing query
can be specified.
            -->
            <Partition Name="Finance" EstimatedRows="39409" StorageMode="Holap">
              <ProactiveCaching Enabled="true">
                <IncrementalNotificationProactiveCachingSource>
                  <IncrementalProcessingNotifications>
                    <IncrementalProcessingNotification>
                      <Query><!-- Insert SQL here --></Query>
                      <ProcessingQuery></ProcessingQuery>
                    </IncrementalProcessingNotification>
                  </IncrementalProcessingNotifications>
                </IncrementalNotificationProactiveCachingSource>
              </ProactiveCaching>
              <DsvTableSource TableName="FactFinance" />
            </Partition>
          </Partitions>
        </CubeMeasureGroup>
      </CubeMeasureGroups>
    </Cube>
  </Cubes>
</Biml>
```

Partition - Table Notification

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Cubes>
    <Cube Name="Adventure Works" ConnectionName="AdvWorksCube">
      <CubeMeasureGroups>
        <CubeMeasureGroup Name="Fact Currency Rate" FactName="FactCurrencyRate">
          <AggregationDesigns>
            <AggregationDesign Name="Currency Rate AggregationDesign"/>
          </AggregationDesigns>
          <Partitions>
            <!--
              This partition has proactive caching enabled, with a SilenceInterval set.
              Tracking tables are specified in the TableNotifications list.
            -->
            <Partition Name="Currency_Rates" EstimatedRows="14264" AggregationDesignName="Currency
Rate AggregationDesign">
              <ProactiveCaching Enabled="true" IsSilenceEnabled="true" SilenceInterval="PT10S">
                <SqlServerTableNotificationProactiveCachingSource>
                  <TableNotifications>
                    <TableReference TableName="FactCurrencyRate" />
                    <TableReference TableName="FactSalesQuota" />
                  </TableNotifications>
                </SqlServerTableNotificationProactiveCachingSource>
              </ProactiveCaching>
              <!-- The FactCurrencyRate table provides the source data for the partition. -->
              <DsvTableSource TableName="FactCurrencyRate" />
            </Partition>
          </Partitions>
        </CubeMeasureGroup>
      </CubeMeasureGroups>
    </Cube>
  </Cubes>
</Biml>
```

Partition - Query Source

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Connections>
    <Connection Name="AdventureWorksDW2008" ConnectionString="Data Source=(local);Initial
Catalog=AdventureWorksDW2008;Provider=SQLNCLI10.1;Integrated Security=SSPI;" />
  </Connections>
  <Cubes>
    <Cube Name="Adventure Works" ConnectionName="AdvWorksCube">
      <CubeMeasureGroups>
        <CubeMeasureGroup Name="Fact Internet Sales 1" FactName="FactInternetSales">
          <Partitions>
            <!-- This Biml partition uses a Query Source to specify the query and SQL connection
that provide data for the partition. -->
            <Partition Name="Internet_Sales_2004" EstimatedRows="32265">
              <ProactiveCaching />
              <QuerySource ConnectionName="AdventureWorksDW2008">
                <Query>SELECT [dbo].[FactInternetSales].[ProductKey],[dbo].
[FactInternetSales].[OrderDateKey],[dbo].[FactInternetSales].[DueDateKey]
                FROM [dbo].[FactInternetSales]
                WHERE OrderDateKey >= '20040101' AND OrderDateKey <=
'20041231'</Query>
              </QuerySource>
            </Partition>
          </Partitions>
        </CubeMeasureGroup>
      </CubeMeasureGroups>
    </Cube>
  </Cubes>
</Biml>
```

Aggregation Design

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Cubes>
    <Cube Name="Adventure Works" ConnectionName="AdvWorksCube">
      <CubeMeasureGroups>
        <CubeMeasureGroup Name="Fact Currency Rate" FactName="FactCurrencyRate">
          <!--
            In Biml, Aggregations are defined within an AggregationDesign.
            Along with generating aggregations in Mist, you can define your own directly in Biml.
          -->
          <AggregationDesigns>
            <AggregationDesign Name="Currency Rate AggregationDesign">
              <Aggregations>
                <Aggregation Name="UsageBased_1">
                  <Dimensions>
                    <Dimension CubeDimensionName="Adventure Works.Destination Currency">
                      <Attributes>
                        <Attribute AttributeName="DimDestinationCurrency.Destination
Currency" />

                      </Attributes>
                    </Dimension>
                    <Dimension CubeDimensionName="Adventure Works.Date">
                      <Attributes>
                        <Attribute AttributeName="DimDate.Calendar Quarter of Year" />
                      </Attributes>
                    </Dimension>
                  </Dimensions>
                </Aggregation>
                <Aggregation Name="UserAggregation_2">
                  <Dimensions>
                    <Dimension CubeDimensionName="Adventure Works.Date">
                      <Attributes>
                        <Attribute AttributeName="DimDate.Date" />
                        <Attribute AttributeName="DimDate.Day Name" />
                      </Attributes>
                    </Dimension>
                  </Dimensions>
                </Aggregation>
              </Aggregations>
            </AggregationDesign>
          </AggregationDesigns>
        </CubeMeasureGroup>
      </CubeMeasureGroups>
      <CubeDimensions>
        <CubeDimension Name="Destination Currency" DimensionName="DimDestinationCurrency">
          <Attributes>
            <Attribute AttributeName="DimDestinationCurrency.Destination Currency" />
            <Attribute AttributeName="DimDestinationCurrency.Destination Currency Code" />
            <Attribute AttributeName="DimDestinationCurrency.Locale" />
          </Attributes>
        </CubeDimension>
        <CubeDimension Name="Date" DimensionName="DimDate"/>
      </CubeDimensions>
    </Cube>
  </Cubes>
</Biml>
```

Calculated Member

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Cubes>
    <Cube Name="Adventure Works" ConnectionName="AdvWorksCube">
      <MdxScripts>
        <MdxScript Name="MdxScript1" Default="true">
          <MdxScriptItems>
            <!--
              This calculated member specifies a MdxScript and references a cube measure group.
            -->
            <CalculatedMember Name="[Internet Gross Profit]" FormatString="Currency"
ParentDimensionName="Measures" AssociatedMeasureGroupName="Fact Internet Sales 1">
              <MdxScript>[Measures].[Internet Sales Amount] - [Measures].[Internet Total Product
Cost]</MdxScript>

              <NonemptyBehaviors>
                <NonemptyBehavior MeasureName="FactInternetSales.Internet Sales Amount" />
                <NonemptyBehavior MeasureName="FactInternetSales.Internet Total Product Cost"
/>

              </NonemptyBehaviors>
            </CalculatedMember>
          </MdxScriptItems>
        </MdxScript>
      </MdxScripts>
    <CubeMeasureGroups>
      <CubeMeasureGroup Name="Fact Internet Sales 1" FactName="FactInternetSales">
        <Measures>
          <Measure MeasureName="FactInternetSales.Internet Sales Amount" />
          <Measure MeasureName="FactInternetSales.Internet Order Quantity" />
          <Measure MeasureName="FactInternetSales.Internet Extended Amount" />
          <Measure MeasureName="FactInternetSales.Internet Total Product Cost" />
        </Measures>
      </CubeMeasureGroup>
    </CubeMeasureGroups>
  </Cube>
</Cubes>
</Biml>
```

Named Set

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Cubes>
    <Cube Name="Adventure Works" ConnectionName="AdvWorksCube">
      <MdxScripts>
        <MdxScript Name="MdxScript1" Default="true">
          <MdxScriptItems>
            <!-- Biml NamedSets allow you to specify a Type (dynamic or static) and a MdxScript.
-->

            <NamedSet Name="[Summary P&L]" DisplayFolder="Sets" Type="Static">
              <MdxScript>{
                [Account].[Accounts].[Account Level 01].&[47], -- Net Income
                [Account].[Accounts].[Account Level 02].&[48], -- Operating Profit
                [Account].[Accounts].[Account Level 03].&[49], -- Gross Margin
                [Account].[Accounts].[Account Level 04].&[50], -- Net Sales
                [Account].[Accounts].[Account Level 05].&[54], -- Discounts
                [Account].[Accounts].[Account Level 05].&[51], -- Gross Sales
                [Account].[Accounts].[Account Level 05].&[53], -- Returns
                [Account].[Accounts].[Account Level 04].&[55], -- Total Cost/Sales
                [Account].[Accounts].[Account Level 03].&[58], -- Operating Expense
                [Account].[Accounts].[Account Level 02].&[88], -- Other I/E
                [Account].[Accounts].[Account Level 03].&[93], -- Currency G/L
                [Account].[Accounts].[Account Level 03].&[91], -- Asset Sales G/L
                [Account].[Accounts].[Account Level 03].&[90], -- Interest Expense
                [Account].[Accounts].[Account Level 03].&[89], -- Interest Income
                [Account].[Accounts].[Account Level 03].&[92], -- Other Income
                [Account].[Accounts].[Account Level 02].&[94] -- Taxes
              }</MdxScript>
            </NamedSet>
          </MdxScriptItems>
        </MdxScript>
      </MdxScripts>
    <CubeDimensions>
      <CubeDimension Name="Account" DimensionName="DimAccount">
        <Attributes>
          <Attribute AttributeName="DimAccount.Account" />
          <Attribute AttributeName="DimAccount.Account Number" />
          <Attribute AttributeName="DimAccount.Account Type" />
          <Attribute AttributeName="DimAccount.Accounts" />
        </Attributes>
      </CubeDimension>
    </CubeDimensions>
  </Cube>
</Cubes>
</Biml>
```

Script Command

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Cubes>
    <Cube Name="Adventure Works" ConnectionName="AdvWorksCube">
      <MdxScripts>
        <MdxScript Name="MdxScript1" Default="true">
          <MdxScriptItems>
            <!--
              Biml ScriptCommands have names along with MdxScripts blocks.
              When multiple calculations are defined in a MdxScriptItems collection, their order
matters.

            -->
            <ScriptCommand Name="Calculate">
              <MdxScript>/*-- Aggregate leaf data -----*/

                  Calculate</MdxScript>
            </ScriptCommand>
            <ScriptCommand Name="Alter Cube">
              <MdxScript>
                /*-- Set default member for the Destination Currency cube dimension ----*/
                Alter Cube
                CurrentCube
                Update Dimension [Destination Currency].[Destination Currency],
                Default_Member = [Destination Currency].[Destination Currency].[US
Dollar]</MdxScript>
              </ScriptCommand>
              <ScriptCommand Name="Scope">
                <MdxScript>
                  /*-- Set language property for the Destination Currency cube dimension --*/
                  Scope
                  (

                      [Destination Currency].[Destination Currency Code].Members,
                      [Destination Currency].[Destination Currency].[Destination Currency].Members

                  )</MdxScript>
                </ScriptCommand>
              </MdxScriptItems>
            </MdxScript>
          </MdxScripts>
        <CubeDimensions>
          <CubeDimension Name="Destination Currency" DimensionName="DimDestinationCurrency">
            <Attributes>
              <Attribute AttributeName="DimDestinationCurrency.Destination Currency" />
              <Attribute AttributeName="DimDestinationCurrency.Destination Currency Code" />
              <Attribute AttributeName="DimDestinationCurrency.Locale" />
            </Attributes>
          </CubeDimension>
        </CubeDimensions>
      </Cube>
    </Cubes>
  </Biml>
```


Miscellaneous

This section has functionality that cuts across various feature sets of BI development.

Log Event

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Packages>
    <Package Name="My Package" ConstraintMode="Linear">
      <Tasks>
        <ExecuteSQL>

          </ExecuteSQL>
          <ExecuteSQL Name="ESQL_Enabled" ConnectionName="OleDbConnection" LoggingMode="Enabled">
            <!-- This defines two log providers. One that will output to a SQL Server database.
                  The other to the Windows Event Log. -->
            <LogProviders>
              <SqlServerLogProvider Name="MyOleDbConnection" ConnectionName="OleDbConnection" />
              <WindowsEventLogProvider Name="WindowsEventLog" />
            </LogProviders>
            <LogEvents>
              <LogEvent EventName="ExecuteSQLExecutingQuery">
                <EventColumns>
                  <EventColumn>Computer</EventColumn>
                  <EventColumn>Operator</EventColumn>
                </EventColumns>
              </LogEvent>
            </LogEvents>
            <DirectInput>SELECT * FROM INFORMATION_SCHEMA.TABLES</DirectInput>
          </ExecuteSQL>
        </Tasks>
      </Package>
    </Packages>
  </Biml>
```

Log Provider

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Packages>
    <Package Name="My Package" ConstraintMode="Linear">
      <Tasks>
        <ExecuteSQL>

          </ExecuteSQL>
          <ExecuteSQL Name="ESQL_Enabled" ConnectionName="OleDbConnection" LoggingMode="Enabled">
            <!-- This defines two log providers. One that will output to a SQL Server database.
                  The other to the Windows Event Log. -->
            <LogProviders>
              <SqlServerLogProvider Name="MyOleDbConnection" ConnectionName="OleDbConnection" />
              <WindowsEventLogProvider Name="WindowsEventLog" />
            </LogProviders>
            <LogEvents>
              <LogEvent EventName="ExecuteSQLExecutingQuery">
                <EventColumns>
                  <EventColumn>Computer</EventColumn>
                  <EventColumn>Operator</EventColumn>
                </EventColumns>
              </LogEvent>
            </LogEvents>
            <DirectInput>SELECT * FROM INFORMATION_SCHEMA.TABLES</DirectInput>
          </ExecuteSQL>
        </Tasks>
      </Package>
    </Packages>
  </Biml>
```

Principal

```
<Biml xmlns:Pamlico="http://schemas.mariner.com/pamlico" xmlns="http://schemas.varigence.com/biml.xsd">
  <Principals>
    <!-- Creates a principal that has control access to the SurveyResponses table an R/W access to the
         Federal Research cube -->
    <Principal Name="Admin" Type="SqlUser" ConnectionName="OleDbConnection">
      <Permissions>
        <TablePermission Action="Grant" Target="Control" TableName="dbo.SurveyResponses" />
        <CubePermission Process="true" CubeName="FederalReserve" Access="ReadWrite"
DrillThroughAccess="DrillThroughAndLocalCube" />
      </Permissions>
    </Principal>
  </Principals>
</Biml>
```

Script Projects

Biml places script projects in their own clean Biml file, making it easy to reuse and easier to see the big picture related to a script.

Script Component Project

```
<!-- This is Script Component Project that accesses Azure DataMarket data, pulls the data down
and stores it as new rows in the table. -->
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <ScriptProjects>
    <ScriptComponentProject ProjectCoreName="SC_2bca370105ff4883a705860bac68cfba"
Name="LoadCrimeDataFeed">
      <AssemblyReferences>
        <AssemblyReference AssemblyPath="Microsoft.SqlServer.DTSPipelineWrap.dll" />
        <AssemblyReference AssemblyPath="Microsoft.SqlServer.DTSRuntimeWrap.dll" />
        <AssemblyReference AssemblyPath="Microsoft.SqlServer.PipelineHost.dll" />
        <AssemblyReference AssemblyPath="Microsoft.SqlServer.TxScript.dll" />
        <AssemblyReference AssemblyPath="System.dll" />
        <AssemblyReference AssemblyPath="System.Data.dll" />
        <AssemblyReference AssemblyPath="System.Windows.Forms.dll" />
        <AssemblyReference AssemblyPath="System.Xml.dll" />
        <AssemblyReference AssemblyPath="System.Core, Version=3.5.0.0, Culture=neutral,
PublicKeyToken=b77a5c561934e089, processorArchitecture=MSIL, Custom=null" />
        <AssemblyReference AssemblyPath="System.Data.Services, Version=3.5.0.0, Culture=neutral,
PublicKeyToken=b77a5c561934e089, processorArchitecture=MSIL, Custom=null" />
        <AssemblyReference AssemblyPath="System.Data.DataSetExtensions, Version=3.5.0.0, Culture=neutral,
PublicKeyToken=b77a5c561934e089, processorArchitecture=MSIL, Custom=null" />
        <AssemblyReference AssemblyPath="System.Data.Services.Client, Version=3.5.0.0, Culture=neutral,
PublicKeyToken=b77a5c561934e089, processorArchitecture=MSIL, Custom=null" />
      </AssemblyReferences>
      <OutputBuffers>
        <OutputBuffer Name="ODataOutput" IsSynchronous="false">
          <Columns>
            <Column Name="ROWID" DataType="Int32" />
            <Column Name="State" DataType="String" Length="20" />
            <Column Name="City" DataType="String" Length="50" />
            <Column Name="Year" DataType="Int32" />
            <Column Name="Population" DataType="Int32" />
            <Column Name="PropertyCrime" DataType="Int32" />
            <Column Name="Burglary" DataType="Int32" />
            <Column Name="LarcenyTheft" DataType="Int32" />
            <Column Name="MotorVehicleTheft" DataType="Int32" />
            <Column Name="Arson" DataType="Int32" />
          </Columns>
        </OutputBuffer>
      </OutputBuffers>
    </ScriptComponentProject>
  </ScriptProjects>
  <Files>
    <File Path="AssemblyInfo.cs">using System.Reflection;
using System.Runtime.CompilerServices;

//
// General Information about an assembly is controlled through the following
// set of attributes. Change these attribute values to modify the information
// associated with an assembly.
//
[assembly: AssemblyTitle("SC_2bca370105ff4883a705860bac68cfba.csproj")]
[assembly: AssemblyDescription("")]
[assembly: AssemblyConfiguration("")]
[assembly: AssemblyCompany("Varigence")]
[assembly: AssemblyProduct("SC_2bca370105ff4883a705860bac68cfba.csproj")]
[assembly: AssemblyCopyright("Copyright @ Varigence 2011")]
[assembly: AssemblyTrademark("")]
[assembly: AssemblyCulture("")]
//
// Version information for an assembly consists of the following four values:
//
//      Major Version
```

```
//      Minor Version
//      Build Number
//      Revision
//
// You can specify all the values or you can default the Revision and Build Numbers
// by using the '*' as shown below:

[assembly: AssemblyVersion("1.0.*")]</File>
    <File Path="main.cs"> /* Microsoft SQL Server Integration Services Script Component
* Write scripts using Microsoft Visual C# 2008.
* ScriptMain is the entry point class of the script.*/

using System;
using System.Data;
using System.Linq;
using System.Net;
using Microsoft.SqlServer.Dts.Pipeline Wrapper;
using Microsoft.SqlServer.Dts.Runtime Wrapper;

[Microsoft.SqlServer.Dts.Pipeline.SSISScriptComponentEntryPointAttribute]
public class ScriptMain : UserComponent
{
    public override void PreExecute()
    {
        base.PreExecute();
        /*
        Add your code here for preprocessing or remove if not needed
        */
    }

    public override void PostExecute()
    {
        base.PostExecute();

        /*
        Add your code here for postprocessing or remove if not needed
        You can set read/write variables here, for example:
        Variables.MyIntVar = 100
        */
    }

    public override void CreateNewOutputRows()
    {
        var context = InitializeDataSource();

        int skip = 0;
        const int take = 100;

        while (true)
        {
            var rows = context.CityCrime.Skip(skip).Take(take);

            if (rows.Count() == 0)
            {
                break;
            }

            foreach (var row in rows)
            {
                this.ODataOutputBuffer.AddRow();
                this.ODataOutputBuffer.Arson = row.Arson;
                this.ODataOutputBuffer.Burglary = row.Burglary;
                this.ODataOutputBuffer.City = row.City;
                this.ODataOutputBuffer.LarcenyTheft = row.LarcenyTheft;
                this.ODataOutputBuffer.MotorVehicleTheft = row.MotorVehicleTheft;
            }
        }
    }
}
```

```

        this.ODataOutputBuffer.Population = row.Population;
        this.ODataOutputBuffer.PropertyCrime = row.PropertyCrime;
        this.ODataOutputBuffer.ROWID = row.ROWID;
        this.ODataOutputBuffer.State = row.State;
        this.ODataOutputBuffer.Year = row.Year;
    }

    skip += take;
}

private datagovCrimesContainer InitializeDataSource()
{
    var context =
        new datagovCrimesContainer(
            new Uri("https://api.datamarket.azure.com/Data.ashx/data.gov/Crimes/"));

    context.Credentials = new NetworkCredential(Variables.UserName, Variables.AccountKey);

    return context;
}
}

</File>
</Files>
<ReadOnlyVariables>
    <Variable Namespace="User" DataType="String" VariableName="UserName" />
    <Variable Namespace="User" DataType="String" VariableName="AccountKey" />
</ReadOnlyVariables>
</ScriptComponentProject>
</ScriptProjects>
</Biml>

```


Script Task Project

```

<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <ScriptProjects>
    <ScriptTaskProject ProjectCoreName="ST_19338c2d3c5d43d1b77d01b996ae1485" Name="Simple Task Script
Project">
      <AssemblyReferences>
        <AssemblyReference AssemblyPath="Microsoft.SqlServer.ManagedDTS.dll" />
        <AssemblyReference AssemblyPath="Microsoft.SqlServer.ScriptTask.dll" />
        <AssemblyReference AssemblyPath="System.dll" />
        <AssemblyReference AssemblyPath="System.AddIn.dll" />
        <AssemblyReference AssemblyPath="System.Data.dll" />
        <AssemblyReference AssemblyPath="System.Windows.Forms.dll" />
        <AssemblyReference AssemblyPath="System.Xml.dll" />
      </AssemblyReferences>
      <Files>
        <File Path="AssemblyInfo.cs">using System.Reflection;
using System.Runtime.CompilerServices;

[assembly: AssemblyTitle("ST_19338c2d3c5d43d1b77d01b996ae1485.csproj")]
[assembly: AssemblyDescription("")]
[assembly: AssemblyConfiguration("")]
[assembly: AssemblyCompany("Varigence")]
[assembly: AssemblyProduct("ST_19338c2d3c5d43d1b77d01b996ae1485.csproj")]
[assembly: AssemblyCopyright("Copyright @ Varigence 2011")]
[assembly: AssemblyTrademark("")]
[assembly: AssemblyCulture("")]

[assembly: AssemblyVersion("1.0.*")]</File>
        <File Path="ScriptMain.cs">

using System;
using System.Data;
using Microsoft.SqlServer.Dts.Runtime;
using System.Windows.Forms;

[System.AddIn.AddIn("ScriptMain", Version = "1.0", Publisher = "", Description = "")]
public partial class ScriptMain : Microsoft.SqlServer.Dts.Tasks.ScriptTask.VSTARTScriptObjectModelBase
{
    enum ScriptResults
    {
        Success = Microsoft.SqlServer.Dts.Runtime.DTSExecResult.Success,
        Failure = Microsoft.SqlServer.Dts.Runtime.DTSExecResult.Failure
    };

    public void Main()
    {
        if (Dts.Variables["User::TestVariable"].ReadOnly)
        {
            Dts.Events.FireError(0, "Simple Task Script Project", "Variable is readonly", string.Empty,
0);
        }

        Dts.TaskResult = (int)ScriptResults.Success;
    }
}

        </File>
      </Files>
      <ReadOnlyVariables>
        <Variable Namespace="User" DataType="Int32" VariableName="TestVariable" />
      </ReadOnlyVariables>

      <ReadWriteVariables />
    </ScriptTaskProject>
  </ScriptProjects>
</Biml>

```

SSIS Tasks

Biml includes support for all current SSIS tasks, but with a simpler interface and the ability to reuse defined assets across numerous tasks and packages. You can also combine task tags with BimlScript to generate components and tasks based on project metadata.

Keep an eye out for advanced features that simplify common behaviors such as Merge, IsNullPatcher, and the ExternalFile property on the Execute SQL task.

Bulk Insert

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Packages>
    <Package Name="My Package" ConstraintMode="Linear">
      <Tasks>
        <!-- Performs a bulk insert from the file specified by BulkFileConnection into the
SurveyResponses DB -->
        <BulkInsert Name="Bulk Insert Task" ConnectionName="SportsData" BatchSize="1"
FieldTerminator="Comma">
          <DestinationTable TableName="dbo.SurveyResponses" />
          <SourceFile ConnectionName="BulkFileConnection" />
        </BulkInsert>
      </Tasks>
    </Package>
  </Packages>
</Biml>
```

Data Profiling

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Packages>
    <Package Name="My Package" ConstraintMode="Linear">
      <Tasks>
        <!-- Get column statistics on the Response column from the SurveyResponses DB -->
        <DataProfiling Name="Data Profiling Task" OverwriteDestination="true" >
          <FileOutput ConnectionName="ProfileFile" />
          <ProfileRequests>
            <ColumnPatternProfileRequest ConnectionName="AdoNetConnection" Name="Column Stats"
Column="Response" TableName="dbo.SurveyResponses" />
          </ProfileRequests>
        </DataProfiling>
      </Tasks>
    </Package>
  </Packages>
</Biml>
```

Execute Package

```
<Packages>
  <Package Name="Rebuild Warehouse Schema" ConstraintMode="Parallel">
    <!-- Executes a series of four packages -->
    <Tasks>
      <ExecutePackage Name="Create Database">
        <Package PackageName="Create Database" />
      </ExecutePackage>
      <ExecutePackage Name="Create Tables">
        <PrecedenceConstraints>
          <Inputs>
            <Input OutputPathName="Create Database.Output" />
          </Inputs>
        </PrecedenceConstraints>
        <Package PackageName="Create Tables" />
      </ExecutePackage>
      <ExecutePackage Name="Create Scalar-valued Functions">
        <Package PackageName="Create Scalar-valued Functions" />
        <PrecedenceConstraints>
          <Inputs>
            <Input OutputPathName="Create Tables.Output" />
          </Inputs>
        </PrecedenceConstraints>
      </ExecutePackage>
      <ExecutePackage Name="Create Database Triggers">
        <Package PackageName="Create Database Triggers" />
        <PrecedenceConstraints>
          <Inputs>
            <Input OutputPathName="Create Scalar-valued Functions.Output" />
          </Inputs>
        </PrecedenceConstraints>
      </ExecutePackage>
    </Tasks>
  </Package>
</Packages>
```

Execute SQL

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Packages>
    <Package Name="Package1" ConstraintMode="Parallel">
      <Tasks>
        <!-- This Execute SQL task processes a query in the SqlQuery variable.
        It also returns a full result set and stores it in the Result variable. -->
        <ExecuteSQL Name="Execute SQL Task 1" ConnectionName="AdventureWorks" ResultSet="Full">
          <VariableInput VariableName="User.SqlQuery" />
          <Variables>
            <Variable Name="SqlQuery" DataType="String">select ProductID, LocationID, Shelf,
BINARY_CHECKSUM, Quantity
                        from AdventureWorks.Production.ProductInventory
                        where Quantity < 100</Variable>
            <Variable Name="Result" DataType="Object"></Variable>
          </Variables>
          <Results>
            <Result Name="Result" VariableName="User.Result" />
          </Results>
        </ExecuteSQL>
      </Tasks>
    </Package>
  </Packages>
</Biml>
```

Execute Process

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Packages>
    <Package Name="My Package" ConstraintMode="Linear">
      <Tasks>
        <!-- Executes the unzip process on a zipped document -->
        <ExecuteProcess Name="Execute Process Task" Executable="C:\BIN\unzip.exe"
WorkingDirectory="C:\Public" StandardInputVariableName="User::CompressedInput" >
          <Variables>
            <Variable Name="CompressedInput" DataType="String">documents.zip</Variable>
          </Variables>
        </ExecuteProcess>
      </Tasks>
    </Package>
  </Packages>
</Biml>
```


File System

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Packages>
    <Package Name="My Package" ConstraintMode="Linear">
      <Tasks>
        <!-- Copies a file from the local file system, from c:\public to c:\public\copy -->
        <FileSystem Name="File System Task" Operation="CopyFile" OverwriteDestination="true">
          <ExternalFileInput ExternalFilePath="c:\public\document.txt" />
          <ExternalFileOutput ExternalFilePath="c:\public\copy\document.txt" />
        </FileSystem>
      </Tasks>
    </Package>
  </Packages>
</Biml>
```

Execute SQL Parameters

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Packages>
    <Package Name="Package1" ConstraintMode="Parallel">
      <Variables>
        <Variable Name="etlp_BatchId" DataType="Int32" EvaluateAsExpression="false"
InheritFromPackageParentConfigurationString="User::etlp_BatchId" >0</Variable>
        <Variable Name="etlp_PackageLogId" DataType="Int32" >0</Variable>
        <Variable Name="etlp_BatchName" DataType="String" >Test Batch</Variable>
      </Variables>
      <Tasks>
        <!--
        In this Execute SQL task, the query uses the ? character to indicate parameters.
        The actual parameters are listed in the Parameters list.
        -->
        <ExecuteSQL Name="SQL_LogPackageEnd">
          <DirectInput>EXEC etlp.LogPackageEnd ?,?,?,</DirectInput>
          <Parameters>
            <Parameter Name="0" VariableName="User.etlp_PackageLogId" />
            <Parameter Name="1" VariableName="User.etlp_BatchId" />
            <Parameter Name="2" VariableName="User.etlp_EndBatchAudit" />
          </Parameters>
        </ExecuteSQL>
      </Tasks>
    </Package>
  </Packages>
</Biml>
```

Foreach File Loop

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Packages>
    <Package Name="Parent" ConstraintMode="Parallel">
      <Tasks>
        <!-- This ForEachFileLoop task uses a variable to communicate fully qualified paths to its
FileSystem task. -->
        <ForEachFileLoop Name="Foreach File Loop 1" ConstraintMode="Parallel" Folder="C:\Orders"
FileSpecification="*.*" ProcessSubfolders="true">
          <Tasks>
            <FileSystem Name="File System Task 1" Operation="MoveFile"
OverwriteDestination="true">
              <VariableInput VariableName="User.ForeachFilePath" />
              <FileOutput ConnectionName="ArchiveConnection" />
            </FileSystem>
          </Tasks>
          <Variables>
            <Variable Name="ForeachFilePath" DataType="String"></Variable>
          </Variables>
          <VariableMappings>
            <VariableMapping Name="Mapping" VariableName="User.ForeachFilePath" />
          </VariableMappings>
        </ForEachFileLoop>
      </Tasks>
    </Package>
  </Packages>
</Biml>
```

FTP

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Packages>
    <Package Name="My Package" ConstraintMode="Linear">
      <Tasks>
        <!-- FTP a remote file from an FTP server to the local machine -->
        <Ftp Name="Ftp Task" Operation="Receive" ConnectionName="MyFtpConnection" >
          <ExternalFileInput ExternalFilePath="c:\public" FileUsageType="ExistingFolder" />
          <RemotePath>/logs/staging.varigence.com.log</RemotePath>
        </Ftp>
      </Tasks>
    </Package>
  </Packages>
</Biml>
```

Message Queue

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Packages>
    <Package Name="My Package" ConstraintMode="Linear">
      <Tasks>
        <!-- Sending a message to the message queue specified in MsmqConnection -->
        <MessageQueue Name="Message Queue Task" MsmqConnectionName="MsmqConnection">
          <FileInput ConnectionName="MyMessageFile" />
        </MessageQueue>
      </Tasks>
    </Package>
  </Packages>
</Biml>
```

Precedence Constraints

```
<Packages>
  <Package Name="Rebuild Warehouse Schema" ConstraintMode="Parallel">
    <!-- A series of ExecutePackage tasks that are constrained with precedence constraints
         specified in each task -->
    <Tasks>
      <ExecutePackage Name="Create Database">
        <Package PackageName="Create Database" />
      </ExecutePackage>
      <ExecutePackage Name="Create Tables">
        <PrecedenceConstraints>
          <Inputs>
            <Input OutputPathName="Create Database.Output" />
          </Inputs>
        </PrecedenceConstraints>
        <Package PackageName="Create Tables" />
      </ExecutePackage>
      <ExecutePackage Name="Create Scalar-valued Functions">
        <Package PackageName="Create Scalar-valued Functions" />
        <PrecedenceConstraints>
          <Inputs>
            <Input OutputPathName="Create Tables.Output" />
          </Inputs>
        </PrecedenceConstraints>
      </ExecutePackage>
      <ExecutePackage Name="Create Database Triggers">
        <Package PackageName="Create Database Triggers" />
        <PrecedenceConstraints>
          <Inputs>
            <Input OutputPathName="Create Scalar-valued Functions.Output" />
          </Inputs>
        </PrecedenceConstraints>
      </ExecutePackage>
    </Tasks>
  </Package>
</Packages>
```

WMI Event Watcher

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Packages>
    <Package Name="Package1" ConstraintMode="Parallel">
      <Tasks>
        <!-- This WMI Event Watcher task listens for when the freespace on C:\ falls beneath 100 MB. -
->
          <WmiEventWatcher Name="WMI Event Watcher Task 1" ConnectionName="WmiConnection1">
            <Expressions>
              <Expression PropertyName="WqlQuerySource">"SELECT * FROM __InstanceModificationEvent
WITHIN 5 WHERE TargetInstance ISA \"Win32_LogicalDisk\" AND TargetInstance.Name=\"C:\" AND
TargetInstance.FreeSpace &lt; " + (DT_WSTR, 50)@[User::FreeSpaceThreshold]</Expression>
            </Expressions>
            <DirectInput> </DirectInput>
          </WmiEventWatcher>
        </Tasks>
      <Variables>
        <Variable Name="FreeSpaceThreshold" DataType="Int64">104857600</Variable>
      </Variables>
    </Package>
  </Packages>
</Biml>
```

Script (Simple)

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Packages>
    <Package Name="My Package" ConstraintMode="Linear">
      <Tasks>
        <!-- A simple script project that runs a script which is specified in the Script Project named
              "Simple Task Script Project" -->
        <Script Name="Script Task" LoggingMode="Enabled">
          <ScriptTaskProjectReference ScriptTaskProjectName="Simple Task Script Project" />
        </Script>
      </Tasks>
    </Package>
  </Packages>
</Biml>
```


Send Mail

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Packages>
    <Package Name="FailureNotifier" ConstraintMode="Parallel">
      <Tasks>
        <!-- This Send Mail task notifies admin@varigence.com of a problem. -->
        <SendMail Name="Send Mail Task 2" ConnectionName="SmtpConnection2"
FromLine="user@varigence.com" ToLine="admin@varigence.com" Subject="There is a problem" Priority="High">
          <DirectInput>A package has unexpectedly failed! Please investigate.</DirectInput>
        </SendMail>
      </Tasks>
    </Package>
  </Packages>
</Biml>
```

Dataflow (Simple)

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Packages>
    <Package Name="Rebuild Warehouse Data" ConstraintMode="Linear" AutoCreateConfigurationsType="None">
      <Tasks>
        <#
        foreach (var table in OrderTablesByDependency())
        {
          if (!table.Name.StartsWith("v"))
          {
            #>
            <!-- Dataflow task that copies data from a source table to a destination table -->
            <Dataflow Name="Copy Data <#=table.Name#>">
              <Transformations>
                <OleDbSource Name="Retrieve Data" ConnectionName="Source">
                  <DirectInput>SELECT * FROM <#=table.Name#></DirectInput>
                </OleDbSource>
                <OleDbDestination Name="Insert Data" ConnectionName="Target" KeepIdentity="true">
                  <ExternalTableOutput Table="<#=table.Name#>" />
                </OleDbDestination>
              </Transformations>
            </Dataflow>
            <# }
          } #>
        </Tasks>
      </Package>
    </Packages>
  </Biml>
```

WMI Data Reader

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Packages>
    <Package Name="My Package" ConstraintMode="Linear">
      <Tasks>
        <!-- Gets WMI information about the local logical disk and write it to a file specified by
WmiFileConnection -->
        <WmiDataReader Name="WMI Data Reader Task" ConnectionName="WmiConnection"
OverwriteDestination="OverwriteDestination">
          <DirectInput>SELECT FreeSpace, DeviceId, Size, SystemName, Description FROM
Win32_LogicalDisk</DirectInput>
          <FileOutput ConnectionName="WmiFileConnection" />
        </WmiDataReader>
      </Tasks>
    </Package>
  </Packages>
</Biml>
```

Transformations

Biml includes support for all current SSIS dataflow transformations, but with a simpler interface and the ability to reuse defined assets across numerous tasks and packages. You can also combine with BimlScript to generate components and transformations based on project metadata.

Aggregate

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Packages>
    <Package Name="Package" ConstraintMode="Parallel">
      <Tasks>
        <Dataflow Name="Dataflow Task 1">
          <Transformations>
            <!-- Outputs distinct count information for the DisplayName column
                  using the Aggregate transformation. -->
            <Aggregate Name="Aggregate Transform">
              <InputPath OutputPathName="FedReserve.Output" />
              <OutputPaths>
                <OutputPath Name="Aggregate Transform Output">
                  <Columns>
                    <Column SourceColumn="DisplayName" Operation="CountDistinct" />
                  </Columns>
                </OutputPath>
              </OutputPaths>
            </Aggregate>
            <OleDbSource Name="FedReserve" ConnectionName="FederalReserveInstruments">
              <Columns>
                <Column SourceColumn="DisplayName" TargetColumn="DisplayName" />
              </Columns>
              <DirectInput>select * from DimFrequency</DirectInput>
            </OleDbSource>
          </Transformations>
        </Dataflow>
      </Tasks>
    </Package>
  </Packages>
</Biml>
```

Audit

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Packages>
    <Package Name="My Package" ConstraintMode="Parallel">
      <Tasks>
        <Dataflow Name="My Dataflow Task">
          <Transformations>
            <OleDbSource Name="FedReserve" ConnectionName="FederalReserveInstruments">
              <DirectInput>select * from DimFrequency</DirectInput>
            </OleDbSource>

            <!-- Add a version ID audit column to the DimFrequency table -->
            <Audit Name="Audit Transformation">
              <Audits>
                <Audit AuditType="VersionId" Name="VersionAudit" />
              </Audits>
            </Audit>
          </Transformations>
        </Dataflow>
      </Tasks>
    </Package>
  </Packages>
</Biml>
```

Character Map

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Packages>
    <Package Name="My Package" ConstraintMode="Parallel">
      <Tasks>
        <Dataflow Name="My Dataflow Task">
          <Transformations>
            <OleDbSource Name="FedReserve" ConnectionName="FederalReserveInstruments">
              <DirectInput>select * from DimFrequency</DirectInput>
            </OleDbSource>

            <!-- Makes characters uppercase for the DisplayName column -->
            <CharacterMap Name="Character Map Transformation">
              <Columns>
                <Column ReplaceExisting="true" SourceColumn="DisplayName">
                  <MappingOptions>
                    <MappingOption>Uppercase</MappingOption>
                  </MappingOptions>
                </Column>
              </Columns>
            </CharacterMap>
          </Transformations>
        </Dataflow>
      </Tasks>
    </Package>
  </Packages>
</Biml>
```

Conditional Split

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Packages>
    <Package Name="My Package" ConstraintMode="Parallel">
      <Tasks>
        <Dataflow Name="My Dataflow Task">
          <Transformations>
            <OleDbSource Name="FedReserve" ConnectionName="FederalReserveInstruments" >
              <DirectInput>select * from DimInstrument</DirectInput>
            </OleDbSource>

            <!-- Outputs the row to the CategoryOut output if the first three letters from the
              Category row are "PRE" -->
            <ConditionalSplit Name="Conditional Split Transformation">
              <InputPath OutputPathName="FedReserve.Output" />
              <OutputPaths>
                <OutputPath Name="CategoryOut">
                  <Expression>SUBSTRING(Category, 1, 3) == "PRE"</Expression>
                </OutputPath>
              </OutputPaths>
            </ConditionalSplit>
          </Transformations>
        </Dataflow>
      </Tasks>
    </Package>
  </Packages>
</Biml>
```


Copy Columns

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Packages>
    <Package Name="My Package" ConstraintMode="Linear" >
      <Tasks>
        <Dataflow Name="My Dataflow Task">
          <Transformations>
            <OleDbSource Name="FedReserve" ConnectionName="FederalReserveInstruments">
              <DirectInput>select * from DimInstrument</DirectInput>
            </OleDbSource>
            <!-- Copy the Category column to a new column called SubCategory -->
            <CopyColumn Name="Copy Column Transformation">
              <Columns>
                <Column SourceColumn="Category" TargetColumn="SubCategory" />

                </Columns>
              </CopyColumn>
              <FlatFileDestination Name="OutputFile" ConnectionName="FlatFileConnection" >

                </FlatFileDestination>
              </Transformations>
            </Dataflow>
          </Tasks>
        </Package>
      </Packages>
    </Biml>
```

Data Conversion

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Packages>
    <Package Name="My Package" ConstraintMode="Parallel">
      <Tasks>
        <Dataflow Name="My Dataflow Task">
          <Transformations>
            <OleDbSource Name="FedReserve" ConnectionName="FederalReserveInstruments" >
              <DirectInput>select * from DimInstrument</DirectInput>
            </OleDbSource>

            <!-- Converts the datatype of the Category column to StringFixedLength -->
            <DataConversion Name="Data Conversion Transformation">
              <Columns>
                <Column SourceColumn="Category" DataType="StringFixedLength" />
              </Columns>
            </DataConversion>
          </Transformations>
        </Dataflow>
      </Tasks>
    </Package>
  </Packages>
</Biml>
```

Derived Columns

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Packages>
    <Package Name="My Package" ConstraintMode="Parallel">
      <Tasks>
        <Dataflow Name="My Dataflow Task">
          <Transformations>
            <OleDbSource Name="FedReserve" ConnectionName="FederalReserveInstruments">
              <DirectInput>select * from DimInstrument</DirectInput>
            </OleDbSource>

            <!-- Derives a new column, Domain, by stripping out the domain from email addresses
specified in the Url column -->
            <DerivedColumns Name="Derived Columns Transformation">
              <Columns>
                <Column Name="Domain" DataType="AnsiString" Length="50">SUBSTRING(Url, 1,
FINDSTRING(Url, "/", 2))</Column>
              </Columns>
            </DerivedColumns>
          </Transformations>
        </Dataflow>
      </Tasks>
    </Package>
  </Packages>
</Biml>
```

Export Column

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Packages>
    <Package Name="My Package" ConstraintMode="Parallel">
      <Tasks>
        <Dataflow Name="My Dataflow Task">
          <Transformations>
            <OleDbSource Name="FedReserve" ConnectionName="FederalReserveInstruments">
              <DirectInput>select * from DimInstrument</DirectInput>
            </OleDbSource>
            <DerivedColumns Name="Derived Columns">
              <InputPath OutputPathName="FedReserve.Output" />
              <Columns>
                <Column DataType="String" Length="50"
Name="FileLocation">"C:\\Users\\MyName\\Documents\\urls.txt"</Column>
              </Columns>
            </DerivedColumns>

            <!-- Exports the Url column to the file specified by FileLocation -->
            <ExportColumn Name="Export Column Transformation">
              <Columns>
                <Column FilePathColumn="FileLocation" ExtractColumn="Url" AllowAppend="true"
/>
              </Columns>
              <InputPath OutputPathName="Derived Columns.Output" />
            </ExportColumn>
          </Transformations>
        </Dataflow>
      </Tasks>
    </Package>
  </Packages>
</Biml>
```

Fuzzy Lookup

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Packages>
    <Package Name="My Package" ConstraintMode="Linear">
      <Tasks>
        <Dataflow Name="My Dataflow Task">
          <Transformations>
            <OleDbSource Name="SurveyResponses" ConnectionName="SportsData">
              <DirectInput>select * from SurveyResponses</DirectInput>
            </OleDbSource>

            <!-- Performs a fuzzy lookup on the Attribute column against the JuniorSurveyResponse
DB, and outputs the corresponding Response column to NewResponse. -->
            <FuzzyLookup Name="Fuzzy Lookup Transformation" ConnectionName="SportsData"
Exhaustive="true" >
              <ReferenceTableInput TableName="dbo.JuniorSurveyResponses" />
              <Inputs>
                <Column SourceColumn="Attribute" TargetColumn="Attribute" />
              </Inputs>
              <Outputs>
                <Column SourceColumn="Response" TargetColumn="NewReponse" />
              </Outputs>
              <InputPath OutputPathName="SurveyResponses.Output" />
            </FuzzyLookup>

            <FlatFileDestination Name="OutputFile" ConnectionName="FlatFileConnection"
Overwrite="true" />
          </Transformations>
        </Dataflow>
      </Tasks>
    </Package>
  </Packages>
</Biml>
```

Fuzzy Grouping

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Packages>
    <Package Name="My Package" ConstraintMode="Linear">
      <Tasks>
        <Dataflow Name="My Dataflow Task">
          <Transformations>
            <OleDbSource Name="SurveyResponses" ConnectionName="SportsData">
              <DirectInput>select * from SurveyResponses</DirectInput>
            </OleDbSource>

            <!-- Does fuzzy grouping of the Response column. -->
            <FuzzyGrouping Name="Fuzzy Grouping Transformation" ConnectionName="SportsData"
Exhaustive="true" SimilarityScoreColumnName="Similarity" OutputKeyColumnName="OutputKey"
InputKeyColumnName="InputKey" >
              <Columns>
                <Column MatchType="Fuzzy" SourceColumn="Response"
GroupOutputAlias="GroupedOutput" TargetColumn="Response" SimilarityOutputAlias="SimilarityOutput" />
              </Columns>
            </FuzzyGrouping>

            <FlatFileDestination Name="OutputFile" ConnectionName="FlatFileConnection"
Overwrite="true" />

          </Transformations>
        </Dataflow>
      </Tasks>
    </Package>
  </Packages>
</Biml>
```

Import Column

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Packages>
    <Package Name="My New Package" ConstraintMode="Linear">
      <Tasks>
        <Dataflow Name="My Dataflow Task">
          <Transformations>
            <ScriptComponentSource ProjectCoreName="SC_eb1debcd2374468ebccbbfad4fbe5976"
Name="Script Component Source">
              <ScriptComponentProjectReference ScriptComponentProjectName="Component Script
Project2" />

              </ScriptComponentSource>

              <!-- Import column from the above Script Component and create the column named
FileName -->

              <ImportColumn Name="Import Column Transformation">
                <InputPath OutputPathName="Script Component Source.Output0" />
                <Columns>
                  <Column SourceColumn="FileName" TargetColumn="FileName" />

                  </Columns>
                </ImportColumn>
                <FlatFileDestination Name="OutputFile" ConnectionName="FlatFileConnection">

                  <InputPath OutputPathName="Import Column Transformation.Output" />

                  </FlatFileDestination>
                </Transformations>
              </Dataflow>
            </Tasks>
          </Package>
        </Packages>
      </Biml>
```

Lookup

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Packages>
    <Package Name="My Package" ConstraintMode="Linear">
      <Tasks>
        <Dataflow Name="My Dataflow Task">
          <Transformations>
            <OleDbSource Name="SurveyResponses" ConnectionName="SportsData">
              <Columns>
                <Column SourceColumn="SurveyId" SortKeyPosition="1" />
                <Column SourceColumn="Attribute" SortKeyPosition="2" />
                <Column SourceColumn="Response" SortKeyPosition="3" />
                <Column SourceColumn="NewResponse" SortKeyPosition="0"
TargetColumn="NewReponse" />
              </Columns>
              <DirectInput>select * from SurveyResponses order by surveyid, attribute,
response</DirectInput>
            </OleDbSource>

            <!-- Performs a lookup on JuniorSurveyResponses over the AttributeColumn, mapping the
resulting Response column to NewResponse -->
            <Lookup Name="Lookup Transformation" OleDbConnectionName="SportsData"
NoMatchBehavior="IgnoreFailure" CacheMode="Partial">
              <DirectInput>
                select * from JuniorSurveyResponses
              </DirectInput>
              <Inputs>
                <Column SourceColumn="Attribute" TargetColumn="Attribute" />
              </Inputs>
              <Outputs>
                <Column SourceColumn="Response" TargetColumn="NewReponse" />
              </Outputs>
              <InputPath OutputPathName="SurveyResponses.Output" />

            </Lookup>
            <FlatFileDestination Name="OutputFile" ConnectionName="FlatFileConnection"
Overwrite="true">
              <Columns>
                <Column SourceColumn="SurveyId" TargetColumn="SurveyId" />
                <Column SourceColumn="Attribute" TargetColumn="Attribute" />
                <Column SourceColumn="Response" TargetColumn="Response" />
                <Column SourceColumn="NewResponse" TargetColumn="NewResponse" IsUsed="true" />
              </Columns>
            </FlatFileDestination>

          </Transformations>
        </Dataflow>
      </Tasks>
    </Package>
  </Packages>
</Biml>
```


Merge

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Packages>
    <Package Name="My Package" ConstraintMode="Linear">
      <Tasks>
        <Dataflow Name="My Dataflow Task">
          <Transformations>
            <OleDbSource Name="SurveyResponses" ConnectionName="SportsData">
              <Columns>
                <Column SourceColumn="SurveyId" SortKeyPosition="1" />
                <Column SourceColumn="Attribute" SortKeyPosition="2" />
                <Column SourceColumn="Response" SortKeyPosition="3" />
              </Columns>
              <DirectInput>select * from SurveyResponses order by surveyid, attribute,
response</DirectInput>
            </OleDbSource>

            <OleDbSource Name="JuniorSurveyResponses" ConnectionName="SportsData">
              <Columns>
                <Column SourceColumn="SurveyId" SortKeyPosition="1" />
                <Column SourceColumn="Attribute" SortKeyPosition="2" />
                <Column SourceColumn="Response" SortKeyPosition="3" />
              </Columns>
              <DirectInput>select * from JuniorSurveyResponses order by surveyid, attribute,
response</DirectInput>
            </OleDbSource>

            <!-- Merges SurveyResponses and JuniorSurveyResponses outputs -->
            <Merge Name="Merge Transformation">
              <LeftInputPath OutputPathName="SurveyResponses.Output" />
              <RightInputPath OutputPathName="JuniorSurveyResponses.Output" />
            </Merge>
            <FlatFileDestination Name="OutputFile" ConnectionName="FlatFileConnection" />
          </Transformations>
        </Dataflow>
      </Tasks>
    </Package>
  </Packages>
</Biml>
```

Merge Join

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Packages>
    <Package Name="My Package" ConstraintMode="Linear">
      <Tasks>
        <Dataflow Name="My Dataflow Task">
          <Transformations>
            <OleDbSource Name="SurveyResponses" ConnectionName="SportsData">
              <DirectInput>select * from SurveyResponses order by response, attribute,
surveyid</DirectInput>
            </OleDbSource>

            <OleDbSource Name="SportDescription" ConnectionName="SportsData">
              <DirectInput>select * from SportDescription order by sport</DirectInput>
            </OleDbSource>

            <!-- Performs a merge joining on SurveyResponses and JuniorSurveyResponses, using
Response and Sport as the Left/Right join keys respectively. -->
            <MergeJoin Name="Merge Join Transformation" JoinType="InnerJoin"
TreatNullsAsEqual="false" MaxBuffersPerInput="10" >
              <LeftInputPath OutputPathName="SurveyResponses.Output" >
                <Columns>
                  <Column SourceColumn="Response" SortKeyPosition="1"
TargetColumn="Response" />
                  <Column SourceColumn="Attribute" SortKeyPosition="2" />
                  <Column SourceColumn="SuveryId" SortKeyPosition="3" />
                </Columns>
              </LeftInputPath>
              <RightInputPath OutputPathName="SportDescription.Output" />
              <JoinKeys>
                <JoinKey LeftColumn="Response" RightColumn="Sport" />
              </JoinKeys>
            </MergeJoin>

            <FlatFileDestination Name="OutputFile" ConnectionName="FlatFileConnection"
Overwrite="true" />
          </Transformations>
        </Dataflow>
      </Tasks>
    </Package>
  </Packages>
</Biml>
```

Multicast

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Packages>
    <Package Name="My Package" ConstraintMode="Linear">
      <Tasks>
        <Dataflow Name="My Dataflow Task">
          <Transformations>
            <OleDbSource Name="FedReserve" ConnectionName="FederalReserveInstruments">
              <DirectInput>select * from DimInstrument</DirectInput>
            </OleDbSource>

            <!-- Distributes the input to two outputs, MultiOut0 and MultiOut1 -->
            <Multicast Name="Multicast Transformation">
              <OutputPaths>
                <OutputPath Name="MultOut0" />
                <OutputPath Name="MultOut1" />
              </OutputPaths>
            </Multicast>

            <FlatFileDestination Name="OutputFile" ConnectionName="FlatFileConnection">
              <InputPath OutputPathName="Multicast Transformation.MultOut0" />
            </FlatFileDestination>
            <FlatFileDestination Name="OutputFile2" ConnectionName="FlatFileConnection2">
              <InputPath OutputPathName="Multicast Transformation.MultOut1" />
            </FlatFileDestination>
          </Transformations>
        </Dataflow>
      </Tasks>
    </Package>
  </Packages>
</Biml>
```

Percentage Sampling

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Packages>
    <Package Name="My Package" ConstraintMode="Linear" >
      <Tasks>
        <Dataflow Name="My Dataflow Task">
          <Transformations>
            <OleDbSource Name="FedReserve" ConnectionName="FederalReserveInstruments">
              <DirectInput>select * from DimInstrument</DirectInput>
            </OleDbSource>
            <!-- Outputs 20% of the rows, randomly, from the input -->
            <PercentageSampling Name="Percentage Sampling Transformation" PercentageOfRows="20"
/>
              <FlatFileDestination Name="OutputFile" ConnectionName="FlatFileConnection" >
                </FlatFileDestination>
              </Transformations>
            </Dataflow>
          </Tasks>
        </Package>
      </Packages>
    </Biml>
```

Pivot

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Packages>
    <Package Name="My Package" ConstraintMode="Linear">
      <Tasks>
        <Dataflow Name="My Dataflow Task">
          <Transformations>
            <OleDbSource Name="SurveyResponses" ConnectionName="SportsData">
              <DirectInput>select * from SurveyResponses order by surveyid, attribute,
response</DirectInput>
            </OleDbSource>

            <!-- Performs a pivot transformation on the SurveyResponse database -->
            <Pivot Name="Pivot Transformation">
              <InputPath OutputPathName="SurveyResponses.Output" />
              <Inputs>
                <Column SourceColumn="Response" PivotUsage="Output" />
                <Column SourceColumn="Attribute" PivotUsage="Pivot" />
                <Column SourceColumn="SurveyId" PivotUsage="Key" />
              </Inputs>

              <Outputs>
                <Column SourceColumn="Response" PivotKeyValue="Name" TargetColumn="Name" />
                <Column SourceColumn="Response" PivotKeyValue="Weight" TargetColumn="Weight"
/>
                <Column SourceColumn="Response" PivotKeyValue="Sport" TargetColumn="Sport" />
              </Outputs>
            </Pivot>

            <FlatFileDestination Name="OutputFile" ConnectionName="FlatFileConnection"
Overwrite="true" />
          </Transformations>
        </Dataflow>
      </Tasks>
    </Package>
  </Packages>
</Biml>
```

Row Count

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Packages>
    <Package Name="My Package" ConstraintMode="Linear">
      <Tasks>
        <Dataflow Name="My Dataflow Task">
          <Transformations>
            <OleDbSource Name="SurveyResponses" ConnectionName="SportsData">
              <DirectInput>select * from SurveyResponses</DirectInput>
            </OleDbSource>

            <!-- Outputs the row count to the User.RowCount variable -->
            <RowCount Name="Row Count Transformation" VariableName="User.RowCount">
              <InputPath OutputPathName="SurveyResponses.Output" />
            </RowCount>

            <FlatFileDestination Name="OutputFile" ConnectionName="FlatFileConnection"
Overwrite="true" />

          </Transformations>
        </Dataflow>
      </Tasks>
      <Variables>
        <Variable Name="RowCount" DataType="Int32">0</Variable>
      </Variables>
    </Package>
  </Packages>
</Biml>
```

Row Sampling

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Packages>
    <Package Name="My Package" ConstraintMode="Linear" >
      <Tasks>
        <Dataflow Name="My Dataflow Task">
          <Transformations>
            <OleDbSource Name="FedReserve" ConnectionName="FederalReserveInstruments">
              <DirectInput>select * from DimInstrument</DirectInput>
            </OleDbSource>
            <!-- Gets a random sample of 25 rows from DimInstrument -->
            <RowSampling Name="Percentage Sampling Transformation" NumberOfRows="25" />
            <FlatFileDestination Name="OutputFile" ConnectionName="FlatFileConnection" >

              </FlatFileDestination>
            </Transformations>
          </Dataflow>
        </Tasks>
      </Package>
    </Packages>
  </Biml>
```

Script Component Source

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Packages>
    <Package Name="My New Package" ConstraintMode="Linear">
      <Tasks>
        <Dataflow Name="My Dataflow Task">
          <Transformations>
            <!-- Creates an input based on a script code defined in "My Script Component Project"
-->
              <ScriptComponentSource ProjectCoreName="SC_eb1debcd2374468ebccbbfad4fbe5976"
Name="Script Component Source Transformation">
                <ScriptComponentProjectReference ScriptComponentProjectName="My Script Component
Project" />

                </ScriptComponentSource>
                <ImportColumn Name="Import Column Transformation">
                  <InputPath OutputPathName="Script Component Source.Output0" />
                  <Columns>
                    <Column SourceColumn="FileName" TargetColumn="FileName" />

                    </Columns>
                  </ImportColumn>
                  <FlatFileDestination Name="OutputFile" ConnectionName="FlatFileConnection">
                    <InputPath OutputPathName="Import Column Transformation.Output" />

                    </FlatFileDestination>
                  </Transformations>
                </Dataflow>
              </Tasks>
            </Package>
          </Packages>
        </Biml>
```


Sort

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Packages>
    <Package Name="My Package" ConstraintMode="Linear" >
      <Tasks>
        <Dataflow Name="My Dataflow Task">
          <Transformations>
            <OleDbSource Name="FedReserve" ConnectionName="FederalReserveInstruments">
              <DirectInput>select * from DimInstrument</DirectInput>
            </OleDbSource>
            <!-- Sorts the rows based using the Category column as the sort key -->
            <Sort Name="Sort Transformation" EliminateDuplicates="true" >
              <Columns>
                <Column Sort="true" SourceColumn="Category" />
              </Columns>
            </Sort>
            <FlatFileDestination Name="OutputFile" ConnectionName="FlatFileConnection" >
              </FlatFileDestination>
            </Transformations>
          </Dataflow>
        </Tasks>
      </Package>
    </Packages>
  </Biml>
```

Term Extraction

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Packages>
    <Package Name="My Package" ConstraintMode="Linear">
      <Tasks>
        <Dataflow Name="My Dataflow Task">
          <Transformations>
            <OleDbSource Name="SurveyResponses" ConnectionName="SportsData">
              <DirectInput>select * from SurveyResponses</DirectInput>
            </OleDbSource>

            <!-- Extracts terms (noun/noun phrases) from the Response column that occur at least
thrice. Outputs the terms found for each row to the new ExtractedTerms column. -->
            <TermExtraction Name="Term Extraction Transformation" TermFrequencyThreshold="3"
CaseSensitiveTermExtraction="false" TermExtractionColumn="Response" TermOutputColumn="ExtractedTerms" >
              <InputPath OutputPathName="SurveyResponses.Output" />
            </TermExtraction>

            <FlatFileDestination Name="OutputFile" ConnectionName="FlatFileConnection"
Overwrite="true" />

          </Transformations>
        </Dataflow>
      </Tasks>
    </Package>
  </Packages>
</Biml>
```

Term Lookup

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Packages>
    <Package Name="My Package" ConstraintMode="Linear">
      <Tasks>
        <Dataflow Name="My Dataflow Task">
          <Transformations>
            <OleDbSource Name="SurveyResponses" ConnectionName="SportsData">
              <DirectInput>select * from SurveyResponses</DirectInput>
            </OleDbSource>

            <!-- Does a term lookup on SurveyResponses using the Response column in the
JuniorSurveyResponses table as the external table where the terms are taken from -->
            <TermLookup Name="Term Lookup Transformation" AutoPassThrough="true"
CaseSensitiveTermLookup="false" ConnectionName="SportsData">
              <InputPath OutputPathName="SurveyResponses.Output" />
              <Columns>
                <Column SourceColumn="Response" InputColumnType="Lookup"
TargetColumn="Response" />
              </Columns>
              <ExternalReferenceTermTableColumnInput Table="JuniorSurveyResponses"
Column="Response" />
            </TermLookup>

            <FlatFileDestination Name="OutputFile" ConnectionName="FlatFileConnection"
Overwrite="true" />
          </Transformations>
        </Dataflow>
      </Tasks>
    </Package>
  </Packages>
</Biml>
```

Union All

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Packages>
    <Package Name="My Package" ConstraintMode="Linear">
      <Tasks>
        <Dataflow Name="My Dataflow Task">
          <Transformations>
            <OleDbSource Name="SurveyResponses" ConnectionName="SportsData">
              <DirectInput>select * from SurveyResponses</DirectInput>
            </OleDbSource>

            <OleDbSource Name="JuniorSurveyResponses" ConnectionName="SportsData">
              <DirectInput>select * from JuniorSurveyResponses</DirectInput>
            </OleDbSource>

            <!-- Does a union of the SurveyResponses and JuniorSurveyResponses to the flat file
specified by FlatFileConnection. -->
            <UnionAll Name="Union All Transformation">
              <InputPaths>
                <InputPath OutputPathName="SurveyResponses.Output" />
                <InputPath OutputPathName="JuniorSurveyResponses.Output" />
              </InputPaths>
            </UnionAll>

            <FlatFileDestination Name="OutputFile" ConnectionName="FlatFileConnection" />

          </Transformations>
        </Dataflow>
      </Tasks>
    </Package>
  </Packages>
</Biml>
```

Unpivot

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Packages>
    <Package Name="My Package" ConstraintMode="Linear">
      <Tasks>
        <Dataflow Name="My Dataflow Task">
          <Transformations>
            <OleDbSource Name="SurveyResponses" ConnectionName="SportsData">
              <DirectInput>select * from SurveyResponses order by surveyid, attribute,
response</DirectInput>
            </OleDbSource>

            <Pivot Name="Pivot Transformation">
              <InputPath OutputPathName="SurveyResponses.Output" />
              <Inputs>
                <Column SourceColumn="Response" PivotUsage="Output" />
                <Column SourceColumn="Attribute" PivotUsage="Pivot" />
                <Column SourceColumn="SurveyId" PivotUsage="Key" />
              </Inputs>

              <Outputs>
                <Column SourceColumn="Response" PivotKeyValue="Name" TargetColumn="Name" />
                <Column SourceColumn="Response" PivotKeyValue="Weight" TargetColumn="Weight"
/>

                <Column SourceColumn="Response" PivotKeyValue="Sport" TargetColumn="Sport" />
              </Outputs>
            </Pivot>

            <!-- After pivoting, this code performs an unpivot, so that the resulting data will be
the same as the input
                to the pivot (although the order may have changed). -->
            <Unpivot Name="Univot Transformation" PivotKeyValueColumnDataType="String"
PivotKeyValueColumnName="Attribute" PivotKeyValueColumnLength="10" AutoPassThrough="false">
              <InputPath OutputPathName="Pivot Transformation.Output" />
              <Columns>
                <Column PivotKeyValue="Name" SourceColumn="Name" TargetColumn="Response" />
                <Column PivotKeyValue="Weight" SourceColumn="Weight" TargetColumn="Response"
/>

                <Column PivotKeyValue="Sport" SourceColumn="Sport" TargetColumn="Response" />
              </Columns>
            </Unpivot>

            <FlatFileDestination Name="OutputFile" ConnectionName="FlatFileConnection"
Overwrite="true" />
          </Transformations>
        </Dataflow>
      </Tasks>
    </Package>
  </Packages>
</Biml>
```

Tables

Tables in Biml allow you to define your cube's entire relational schema. Besides including support for common table elements, such as columns, keys, and indexes, and table types, Biml adds several unique capabilities to tables.

Biml tables can have custom extensions, that execute SSIS tasks after table creation. Additionally, static sources let you specify starter data to fill into a table after creation. Biml also includes a clone table, which lets you specify a baseline table to be copied. The copied table can include or exclude the baseline table's keys, indexes, or columns, as well as add its own. Clone tables come in handy for staging and temporary storage.

Dimension

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Tables>
    <!-- This sample creates a dimension with attributes, relationships, and hierarchies. -->
    <Dimension Name="DimProduct">
      <Columns>
        <Column Name="ProductID"/>
        <Column Name="ProductName" DataType="String" Length="50"/>
        <Column Name="CategoryName" DataType="String" Length="50"/>
        <Column Name="SubcategoryName" DataType="String" Length="50"/>
      </Columns>
      <Attributes>
        <Attribute Usage="Key" Name="ProductName">
          <KeyColumns>
            <KeyColumn ColumnName="ProductID"/>
          </KeyColumns>
          <NameColumn ColumnName="ProductName"/>
        </Attribute>
        <Attribute Name="Category Name">
          <KeyColumns>
            <KeyColumn ColumnName="CategoryName"/>
          </KeyColumns>
        </Attribute>
        <Attribute Name="Subcategory Name">
          <KeyColumns>
            <KeyColumn ColumnName="SubcategoryName"/>
          </KeyColumns>
        </Attribute>
      </Attributes>
      <Relationships>
        <Relationship Name="Category Name" ParentAttributeName="Subcategory Name"
ChildAttributeName="Category Name"/>
        <Relationship Name="Subcategory Name" ParentAttributeName="ProductName"
ChildAttributeName="Subcategory Name"/>
      </Relationships>
      <AttributeHierarchies>
        <Hierarchy Name="Categories">
          <Levels>
            <Level Name="Category Name" AttributeName="DimProduct.Category Name"/>
            <Level Name="Subcategory Name" AttributeName="DimProduct.Subcategory Name"/>
            <Level Name="ProductName" AttributeName="DimProduct.ProductName"/>
          </Levels>
        </Hierarchy>
      </AttributeHierarchies>
    </Dimension>
  </Tables>
</Biml>
```

Fact

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Tables>
    <!-- This sample illustrates a fact table with columns and measures. -->
    <Fact Name="FactSales" ConnectionName="AdventureWorksLTDataMart"
xmlns="http://schemas.varigence.com/biml.xsd">
      <Columns>
        <Column Name="SalesID" />
        <Column Name="SalesOrderNumber" DataType="String" Length="25" />
        <Column Name="TaxAmount" DataType="Currency" />
        <Column Name="Freight" DataType="Currency" />
        <Column Name="OrderQty" DataType="Int16" />
        <Column Name="UnitPrice" DataType="Currency" />
        <Column Name="UnitPriceDiscount" DataType="Currency" />
      </Columns>
      <Measures>
        <Measure Name="Tax Amount" AggregateColumnName="TaxAmount" MeasureFormat="Currency" />
        <Measure Name="Freight" AggregateColumnName="Freight" MeasureFormat="Currency" />
        <Measure Name="Order Qty" AggregateColumnName="OrderQty" SsasDataType="Integer"/>
        <Measure Name="Unit Price" AggregateColumnName="UnitPrice" MeasureFormat="Currency"
AggregateFunction="Min" />
        <Measure Name="Unit Price Discount" AggregateColumnName="UnitPriceDiscount"
MeasureFormat="Currency" />
      </Measures>
    </Fact>
  </Tables>
</Biml>
```


Clone Table

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Tables>
    <Table Name="SampleTable" ConnectionName="TableConnection">
      <Columns>
        <Column Name="Simple" DataType="Int32" IsNullable="false"/>
      </Columns>
    </Table>
    <!-- This CloneTable copies the SampleTable table above and includes an additional column. -->
    <CloneTable
      Name="SampleTableClone"
      CloneIndexes="false"
      CloneKeys="false"
      CloneStaticSources="false"
      NullClonedColumns="true"
      TableName="SampleTable"
      ConnectionName="TableConnection"
    >
      <Columns>
        <Column Name="ExtraColumn" DataType="Int32" />
      </Columns>
    </CloneTable>
  </Tables>
</Biml>
```

Columns

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Tables>
    <Table Name="SampleTable" ConnectionName="TableConnection">
      <Columns>
        <Column Name="Simple" DataType="Int32" />
        <Column Name="IdentityColumn" DataType="Int64" IdentityIncrement="1" IdentitySeed="2" />
        <Column Name="String" DataType="String" Length="255" />
        <Column Name="DefaultValueColumn" DataType="Int32" Default="-1" />
        <Column Name="NotNull" DataType="Int32" IsNullable="false" />

        <!-- A Length of -1 will automatically be converted to nvarchar(max)/varchar(max) -->
        <Column Name="LongString" DataType="String" Length="-1" />

        <Column Name="Decimal" DataType="Decimal" Precision="5" Scale="1" />
        <!-- Computed columns - enter the TSQL expression you would normally enter after the "AS"
statement -->
        <Column Name="ComputedColumn" Computed="[Simple]*2" DataType="Int32"/>

        <!-- For Custom Types, supply the best guess for the data type, length, precision, and scale.
            Additionally set the custom SQL Server type.
            We use the DataType information as a guess when generating OLAP Cubes and other
metadata.
            -->
        <Column Name="Custom" DataType="Binary" Length="8" CustomType="rowversion" />
      </Columns>
    </Table>
  </Tables>
</Biml>
```

Multiple Column Table Reference

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Tables>
    <!-- MultipleColumnTableReference: Columns must be part of the table's unique or primary key.
         Use MultipleColumnTableReferenceGroupName to group a set of columns into a single foreign key-
    -->
    <Table Name="SampleChild" ConnectionName="TableConnection">
      <Columns>
        <MultipleColumnTableReference Name="Column1" ForeignColumnName="SampleParent.Column1"
MultipleColumnTableReferenceGroupName="FK_MyForeignKey" />
        <MultipleColumnTableReference Name="Column2" ForeignColumnName="SampleParent.Column2"
MultipleColumnTableReferenceGroupName="FK_MyForeignKey" />
      </Columns>
    </Table>
  </Tables>
</Biml>
```

Keys

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Tables>
    <Table Name="SalesOrderDetail" ConnectionName="AdventureWorksLT2008" SchemaName="SalesLT" >
      <Columns>
        <TableReference ForeignKeyNameOverride="FK_SalesOrderDetail_SalesOrderHeader_SalesOrderID"
        TableName="SalesLT.SalesOrderHeader" Name="SalesOrderID" />
        <Column Name="SalesOrderDetailID" />
        <Column Name="OrderQty" DataType="Int16" />
        <Column Name="UnitPrice" DataType="Currency" />
        <Column Name="UnitPriceDiscount" DataType="Currency" Default="((0.0))" />
        <Column Name="LineTotal" DataType="Decimal" Precision="38" Scale="6" />
        <Column Name="rowguid" DataType="Guid" Default="(newid())" />
        <Column Name="ModifiedDate" DataType="DateTime" Default="(getdate())" />
      </Columns>
      <Keys>
        <!-- This sample demonstrates using the three types of keys.
              Identity Keys combine a Primary Key and Identity Together in one combination -->
        <Identity Name="IK_SalesOrderDetail">
          <Columns>
            <Column ColumnName="SalesOrderDetailID" />
          </Columns>
        </Identity>
        <UniqueKey Name="AK_SalesOrderDetail_rowguid">
          <Columns>
            <Column ColumnName="rowguid" />
          </Columns>
        </UniqueKey>
        <PrimaryKey Name="PK_SalesOrderDetail_SalesOrderID_SalesOrderDetailID">
          <Columns>
            <Column ColumnName="SalesOrderID" />
            <Column ColumnName="SalesOrderDetailID" />
          </Columns>
        </PrimaryKey>
      </Keys>
    </Table>
  </Tables>
</Biml>
```

Foreign Key

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Tables>
    <Table Name="ProductModelProductDescription" ConnectionName="AdventureWorksLT2008"
SchemaName="SalesLT" >
      <Columns>
        <!-- This sample demonstrates the use of ForeignKeys via TableReference columns. -->
        <TableReference
ForeignKeyNameOverride="FK_ProductModelProductDescription_ProductModel_ProductModelID"
TableName="SalesLT.ProductModel" Name="ProductModelID" />
        <TableReference
ForeignKeyNameOverride="FK_ProductModelProductDescription_ProductDescription_ProductDescriptionID"
TableName="SalesLT.ProductDescription" Name="ProductDescriptionID" />
        <Column Name="Culture" DataType="StringFixedLength" Length="6" />
        <Column Name="rowguid" DataType="Guid" Default="(newid())" />
        <Column Name="ModifiedDate" DataType="DateTime" Default="(getdate())" />
      </Columns>
      <Keys>
        <PrimaryKey
Name="PK_ProductModelProductDescription_ProductModelID_ProductDescriptionID_Culture">
          <Columns>
            <Column ColumnName="ProductModelID" />
            <Column ColumnName="ProductDescriptionID" />
            <Column ColumnName="Culture" />
          </Columns>
        </PrimaryKey>
      </Keys>
    </Table>
    <Table Name="ProductDescription" ConnectionName="AdventureWorksLT2008" SchemaName="SalesLT" >
      <Columns>
        <Column Name="ProductDescriptionID" />
        <Column Name="Description" DataType="String" Length="400" />
        <Column Name="rowguid" DataType="Guid" Default="(newid())" />
        <Column Name="ModifiedDate" DataType="DateTime" Default="(getdate())" />
      </Columns>
    </Table>
    <Table Name="ProductModel" ConnectionName="AdventureWorksLT2008" SchemaName="SalesLT" >
      <Columns>
        <Column Name="ProductModelID" />
        <Column Name="Name" DataType="String" Length="50" />
        <Column Name="CatalogDescription" DataType="Xml" IsNullable="true" />
        <Column Name="rowguid" DataType="Guid" Default="(newid())" />
        <Column Name="ModifiedDate" DataType="DateTime" Default="(getdate())" />
      </Columns>
    </Table>
  </Tables>
</Biml>
```

Foreign Key (Simple)

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Tables>
    <!-- TableReference automatically detects the primary key for a table and includes the correct column
in your model
For multiple column primary keys, see the MultipleColumnTableReference example
-->
    <Table Name="SampleChild" ConnectionName="TableConnection">
      <Columns>
        <TableReference Name="ForeignKeyColumn" TableName="SampleParent"
ForeignKeyNameOverride="FK_ForeignKey" />
      </Columns>
    </Table>
  </Tables>
</Biml>
```

Indexes

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Tables>
    <Table Name="SampleTable" ConnectionName="TableConnection">
      <Columns>
        <Column Name="Simple" DataType="Int32" />
      </Columns>
      <Indexes>
        <Index Name="IX_1" Clustered="false" Online="true" PadIndex="true" FillFactor="50"
SortInTempdb="true">
          <Columns>
            <Column ColumnName="Simple" />
          </Columns>
        </Index>
      </Indexes>
    </Table>
  </Tables>
</Biml>
```

Schema

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <!-- This example demonstrates referencing a database schema in a table. -->
  <Schemas>
    <Schema Name="mySchema" ConnectionName="TableConnection" />
  </Schemas>
  <Tables>
    <Table SchemaName="mySchema" Name="SampleTable" ConnectionName="TableConnection">
      <Columns>
        <Column Name="Simple" DataType="Int32" IsNullable="false"/>
      </Columns>
    </Table>
  </Tables>
</Biml>
```


Static Source

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Tables>
    <Dimension Name="DimStatus" ConnectionName="FederalReserveInstruments">
      <!-- This sample demonstrates a static source in Biml.
            Notice that there are the same number of ColumnValue elements in each ColumnValues list,
for each Row. -->
      <Sources>
        <StaticSource Name="StaticSource">
          <Rows>
            <Row>
              <ColumnValues>
                <ColumnValue ColumnName="StatusID" Value="-1" />
                <ColumnValue ColumnName="DisplayName" Value="N'Unknown'" />
              </ColumnValues>
            </Row>
            <Row>
              <ColumnValues>
                <ColumnValue ColumnName="StatusID" Value="1" />
                <ColumnValue ColumnName="DisplayName" Value="N'Active'" />
              </ColumnValues>
            </Row>
            <Row>
              <ColumnValues>
                <ColumnValue ColumnName="StatusID" Value="2" />
                <ColumnValue ColumnName="DisplayName" Value="N'Discontinued'" />
              </ColumnValues>
            </Row>
          </Rows>
        </StaticSource>
      </Sources>
      <Columns>
        <Column Name="StatusID" />
        <Column Name="DisplayName" DataType="String" Length="32" />
      </Columns>
    </Dimension>
  </Tables>
</Biml>
```

Custom Extension

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
  <Tables>
    <!-- Custom Extensions are custom SSIS tasks which can run after table creation -->
    <Table Name="SampleChild" ConnectionName="TableConnection">
      <Columns>
        <Column Name="Column1" DataType="Int32"/>
      </Columns>
      <CustomExtensions>
        <CustomExtension Name="CustomExtension" ConstraintMode="Linear">
          <Tasks>
            <ExecuteSQL Name="SimpleTask" ConnectionName="TableConnection">
              <DirectInput>
                ALTER TABLE SampleChild ADD Column2 BIGINT NULL
              </DirectInput>
            </ExecuteSQL>
          </Tasks>
        </CustomExtension>
      </CustomExtensions>
    </Table>
  </Tables>
</Biml>
```