PASS SUMMIT 2016

A Crash-Course in Biml

Tim Mitchell, Tyleris Data Solutions

Please silence cell phones

Explore Everything PASS Has to Offer

FREE ONLINE WEBINAR EVENTS

FREE 1-DAY LOCAL TRAINING EVENTS

VOLUNTEERING OPPORTUNITIES

LOCAL USER GROUPS AROUND THE WORLD

ONLINE SPECIAL INTEREST USER GROUPS

PASS COMMUNITY NEWSLETTER

PASS BLOG

WHITE PAPERS

SESSION RECORDINGS

FREE ONLINE RESOURCES

BUSINESS ANALYTICS TRAINING

BA INSIGHTS NEWSLETTER

Session Evaluations

3
ways to access

Your feedback is important and valuable.

Submit by 5pm Friday November 6th to **WIN** prizes

Go to **passSummit.com**

Download the GuideBook App and search: **PASS Summit 2016**

Follow the QR code link displayed on session signage throughout the conference venue and in the program guide

---

Tim Mitchell
Data Architect,
Tyleris Data Solutions

Microsoft Data Platform MVP
Coauthor, SSIS Design Patterns
Dallas, Texas area
Does not like piña coladas or getting caught in the rain

TimMitchell.net   /in/TimMitchell   @Tim_Mitchell

Agenda

- Overview of Biml
- BimlScript
- Practical uses
- Language Overview
- Demos

About the presenter

Tim Mitchell
- Business intelligence consultant
- Principal, Tyleris Data Solutions
- Microsoft Data Platform MVP
- TimMitchell.net
- @Tim_Mitchell

Logistics

- Schedule and break
- Slide deck and code

Overview of Biml

- Business Intelligence Markup Language
- User-friendly XML (*really, stop laughing*) to allow relatively easy design of BI assets in code

Overview of Biml

Patterns-based development
- Faster development
- More consistent development
- Reliable development

Overview of Biml

Patterns-based development
- Supports atomic package design

Overview of Biml

- Practical uses
  - Simple staging
  - Rapidly changing metadata
  - Large number of similar packages
  - Template engine

Overview of Biml

Patterns-based development
- Multiple unique patterns with low repetition = not usually a good fit

Overview of Biml
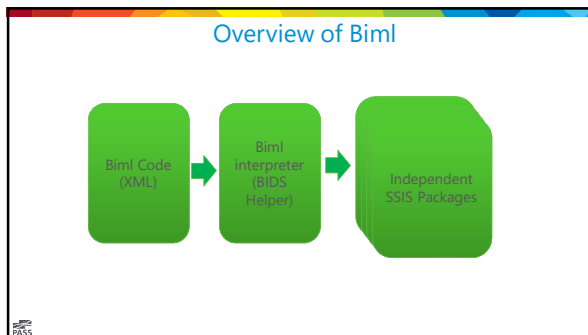
- Can be used against SSIS packages/projects
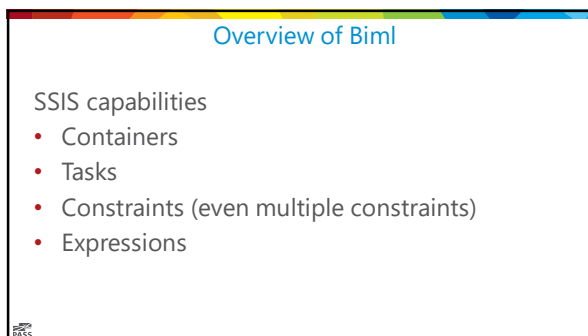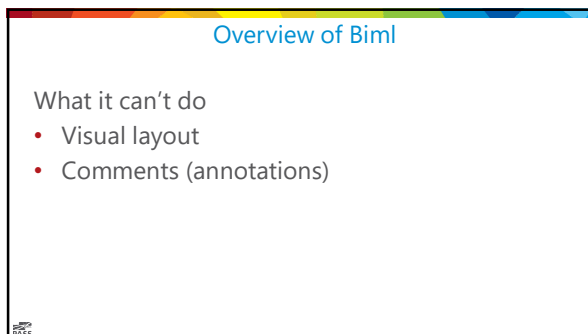- SSAS cubes supported (in commercial version)

Overview of Biml

- BIDS Helper
- Biml Express
- Biml Online
- Mist (commercial tool)

10/28/2016

Overview of Biml

Biml Code (XML) → Biml interpreter (BIDS Helper) → Independent SSIS Packages

---

Overview of Biml

SSIS capabilities
- Containers
- Tasks
- Constraints (even multiple constraints)
- Expressions

Overview of Biml

What it can't do
- Visual layout
- Comments (annotations)

Language structure

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
    <Connections>
        <OleDbConnection Name="AdventureWorks" ConnectionString="Provider=SQLNCLI10;Data Source=localhost..." />
    </Connections>
    <Packages>
        <Package Name="DemoPackage" ConstraintMode="Parallel">
            <Tasks>
                <ExecuteSQL Name="SQLT - Truncate Table" ConnectionName="AdventureWorks">
                    <DirectInput>TRUNCATE TABLE [Sales].[Store]</DirectInput>
                </ExecuteSQL>
            </Tasks>
        </Package>
    </Packages>
</Biml>
```

Language structure

Packages
- Collection of SSIS package objects
- Multiple packages per Biml file

Language structure

Loops and containers
- Sequence containers
- Loops

Language structure

Tasks
- Support for all valid SSIS tasks
- Precedence constraints

Language structure

Data Flow
- Every source and destination available in SSIS
- Transformations

Language structure

Data Flow
- Easy data lineage through automatic mapping
- Manual overrides

Language structure

Variables and Parameters
- Define and set variables
  - Expressions
- Create parameters and set defaults
  - SSIS 2012+ only

Language structure

BimlScript
- Biml XML mixed with .NET code

Language structure

```
<#@ import namespace="System.IO" #>
<#  string directory = "C:\\Where\\Packages\\Are\\Located\\"; #>
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
    <Packages>
        <Package Name="Execute All Packages" ConstraintMode="Parallel">
            <Tasks>
                <# foreach (string filePath in Directory.GetFiles(directory, "*.dtsx") { #>
                <ExecutePackage Name="Run Package - <#=filePath#>">
                    <ExternalFile ExternalFilePath="<#=filePath#>" />
                </ExecutePackage>
                <# } #>
            </Tasks>
        </Package>
    </Packages>
</Biml>
```

Language structure

<# Code Block #>
Any .NET Code (C# or Visual Basic)

<#= Inline Code Block #>
Calls the .NET ToString() method on the expression
Cannot be a statement

<#@ Directive #>
Set configuration options

<#+ Module Level Code Block #>
Functions that are callable from within code block

Language structure

Text Blocks

Text Blocks

XML

< ... >

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">
    <Packages>
        <Package Name="DemoPackage" ConstraintMode="Parallel" />
    </Packages>
</Biml>
```

Language structure

Text Blocks

Directives

XML

<#@ ... #>

```
<#@ template language="C#" tier="0" #>

<#@ include file="i-Parameters.biml" #>
```

Language structure

Standard Control Blocks

Control Blocks

Standard Control Blocks

C# or VB

<# ... #>

```
<# var sourceConnection = RootNode.DbConnections["Source"]; #>
<# var importResult = sourceConnection.ImportDB("", "", ImportOptions.ExcludeForeignKey ); #>
```

```
<#
    var sourceConnection = RootNode.DbConnections["Source"];
    var importResult = sourceConnection.ImportDB("", "", ImportOptions.ExcludeForeignKey );
#>
```

Language structure

Expression Control Blocks

Control Blocks

Expression Control Blocks

C# or VB

<#= ... #>

```
<# var sourceConnection = RootNode.DbConnections["Source"]; #>
<# var importResult = sourceConnection.ImportDB("", "", ImportOptions.ExcludeForeignKey ); #>

<Biml xmlns="http://schemas.varigence.com/biml.xsd">
    <Tables>
    <# foreach (var table in importResult.TableNodes) { #>
        <Table Name="Staging_<#= table.Schema.Name #>_<#= table.Name #>" SchemaName="AW.dbo">
            <Columns>
            <# foreach (var column in table.Columns) { #>
                <#= column.GetBiml() #>
            <# } #>
            </Columns>
        </Table>
    <# } #>
    </Tables>
</Biml>
```

Language structure

Class Feature Control Blocks

Control Blocks

Class Feature Control Blocks

C# or VB

<#+ ... #>

```
<Biml xmlns="http://schemas.varigence.com/biml.xsd">

    <Packages>
        <Package Name="<#= UpperCase("Test Package") #>" ConstraintMode="Linear" />
    </Packages>

</Biml>

<#+
    // Note: Class Feature Control Block must be at the end of the file
    private string UpperCase(string originalValue)
    {
        return originalValue.ToUpper();
    }
#>
```

Language structure

Source control
- Biml files or SSIS packages?
- Why not both?

Language structure

The good:
- Rapid development of lots of packages
- Reusable patterns
- Built into Visual Studio (with BIDS Helper)
- It's free!

Language structure

The challenges:
- It's XML with C# (or VB.net)
- Intellisense
- Debugging

Demos

Resources

- BimlOnline.com
- BIDS Helper: bidshelper.codeplex.com
- BimlScript library/forum: bimlscript.com

Session Evaluations

3 ways to access

Your feedback is important and valuable.

Submit by 5pm Friday November 6th to **WIN** prizes

Go to **passSummit.com**

Download the GuideBook App and search: **PASS Summit 2016**

Follow the QR code link displayed on session signage throughout the conference venue and in the program guide

Thank You

Learn more from
Tim Mitchell
tmitchell@Tyleris.com or follow @tim_mitchell