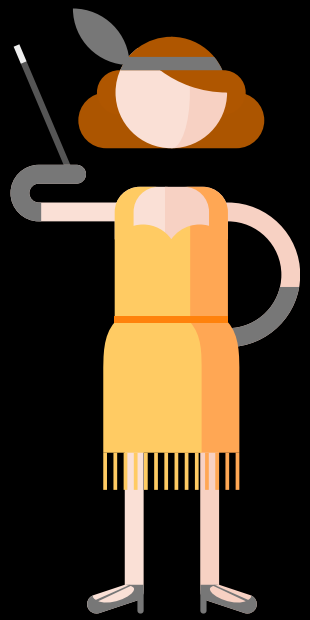
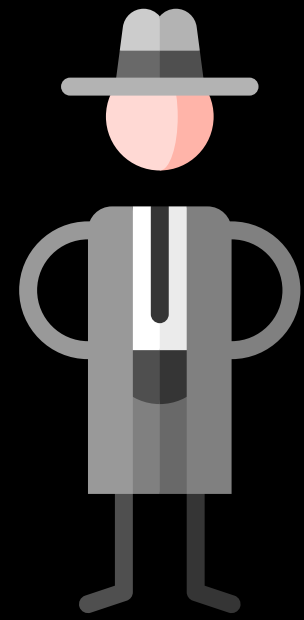
BIML TIPS AND TRICKS

NOT JUST FOR SSIS PACKAGES!

CATHRINE WILHELMSEN

SQLBITS - MARCH 2ND 2019





SESSION DESCRIPTION

"Wait, what? Biml is not just for generating SSIS packages?"

Absolutely not! Come and see how you can use Biml (Business Intelligence Markup Language) to save time and speed up other Data Warehouse development tasks. You can generate complex T-SQL statements with Biml instead of using dynamic SQL, create test data, and even populate static dimensions.

Don't Repeat Yourself, start automating those boring, manual tasks today!

cathrine WILHELMSEN

Microsoft® Most Valuable Professional

@cathrinew

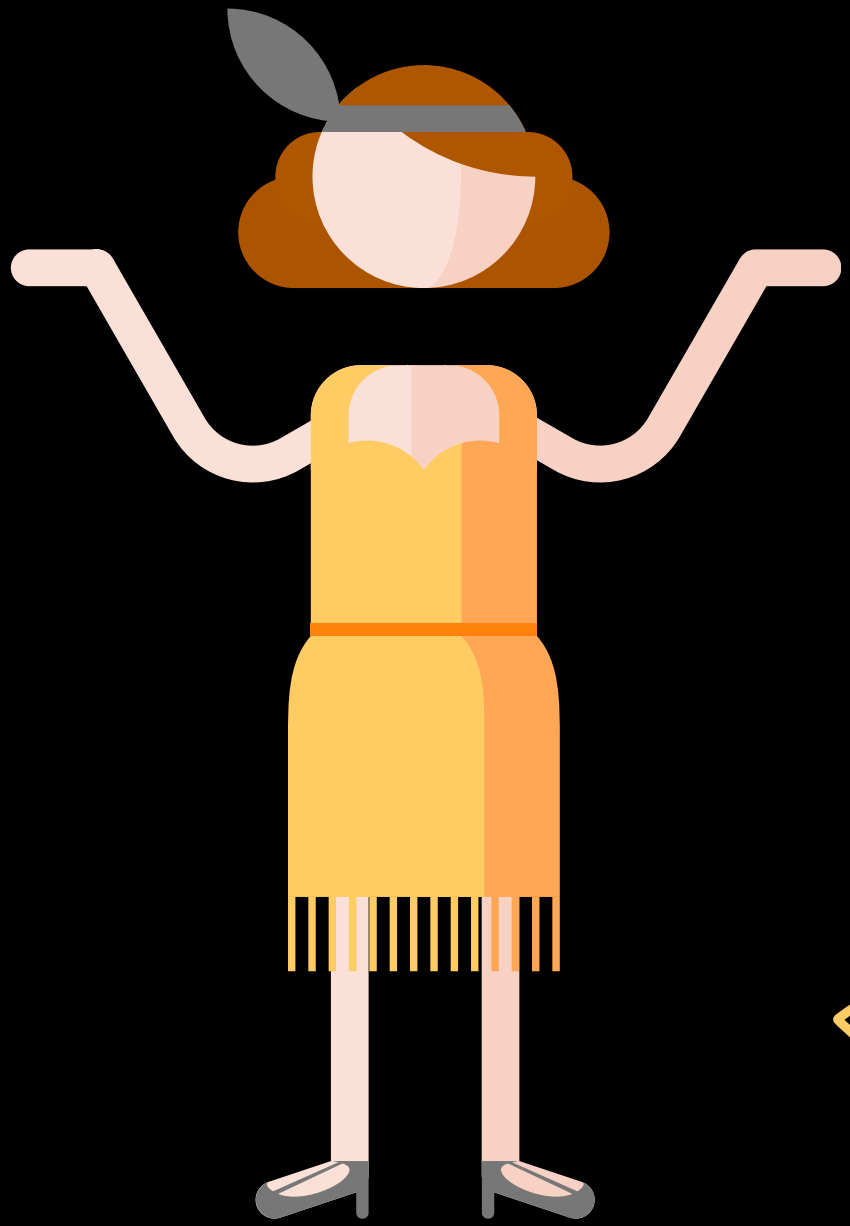
cathrinew.net

THE NEXT 50 MINUTES

T-SQL

Test Data

Dimensions

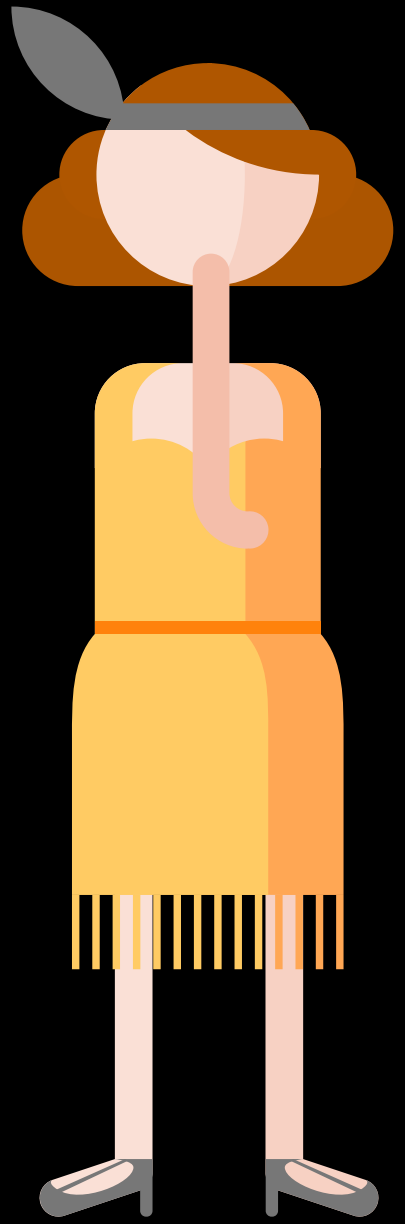


Wait...

Can't I use <something else>?

Of course!

But Biml works well with source metadata

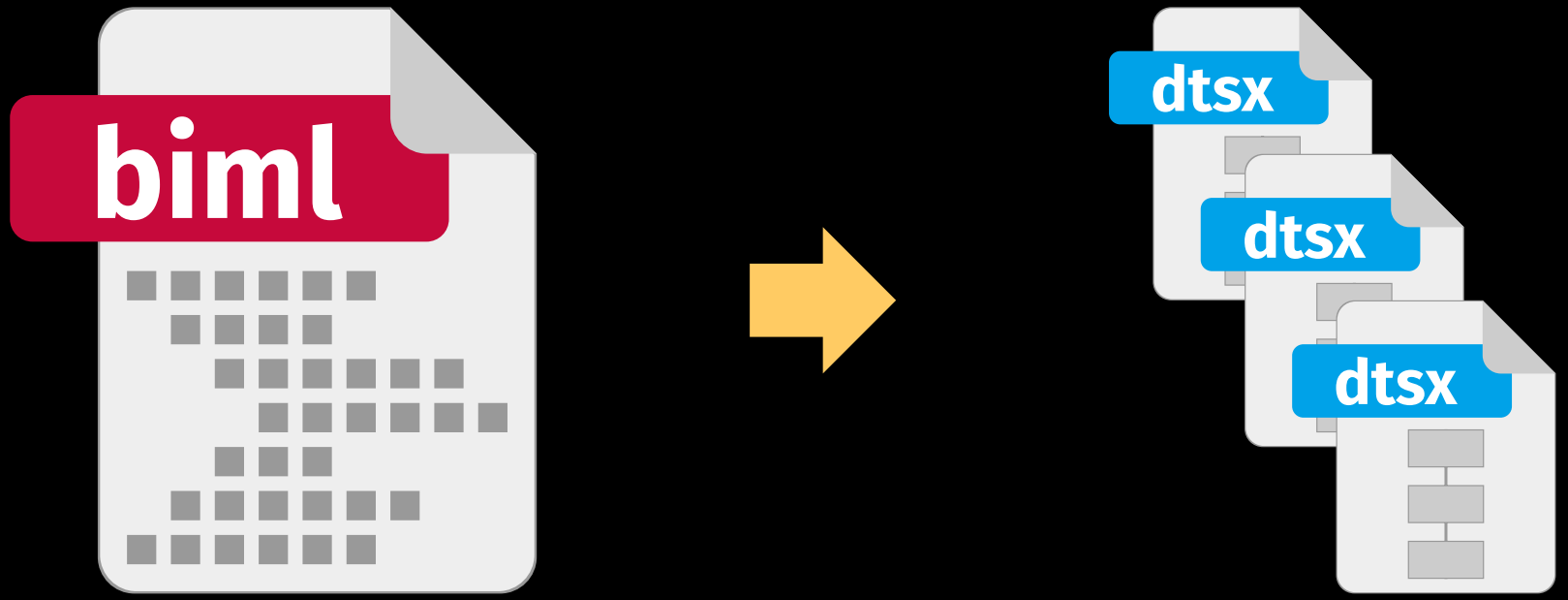


And...

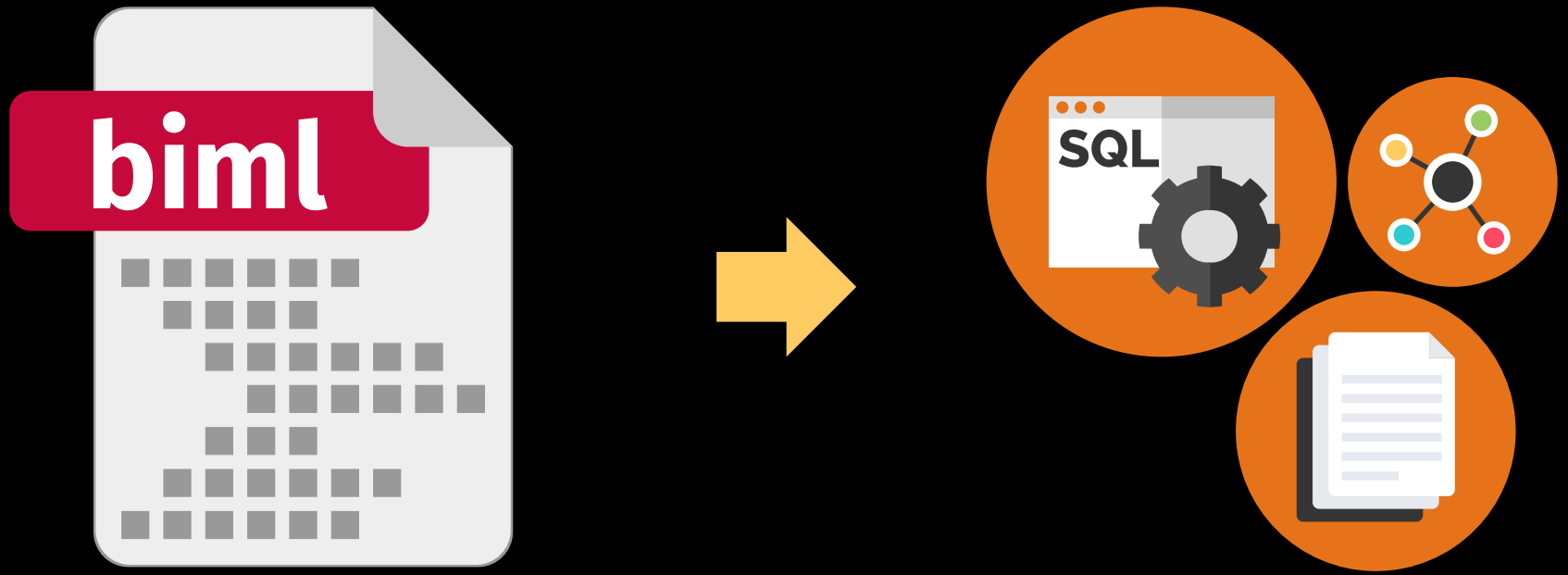
Biml is fun!

BUT... BIML? HOW?

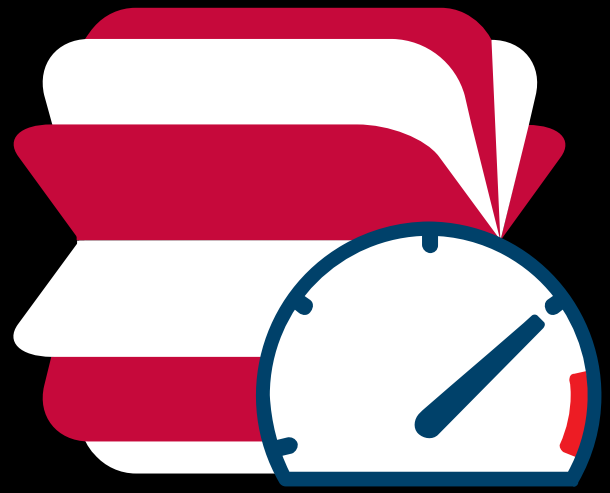
TRADITIONAL BIML



~~CRAZY~~ FUN BIML

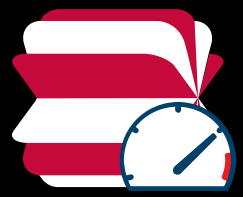


WHAT DO YOU NEED?



BimlExpress

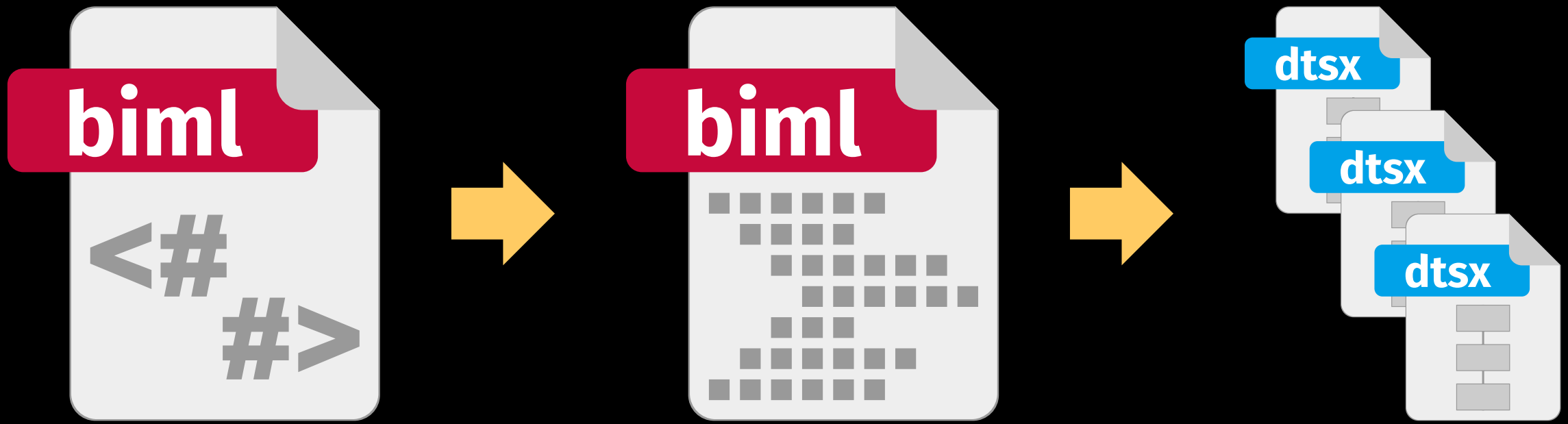
THE POWER IS IN THE...



BimlExpress

Preview Pane

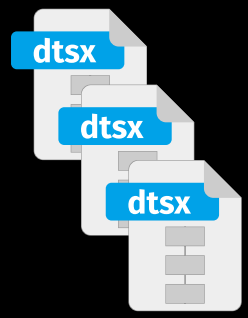
TRADITIONAL BIMLSCRIPT



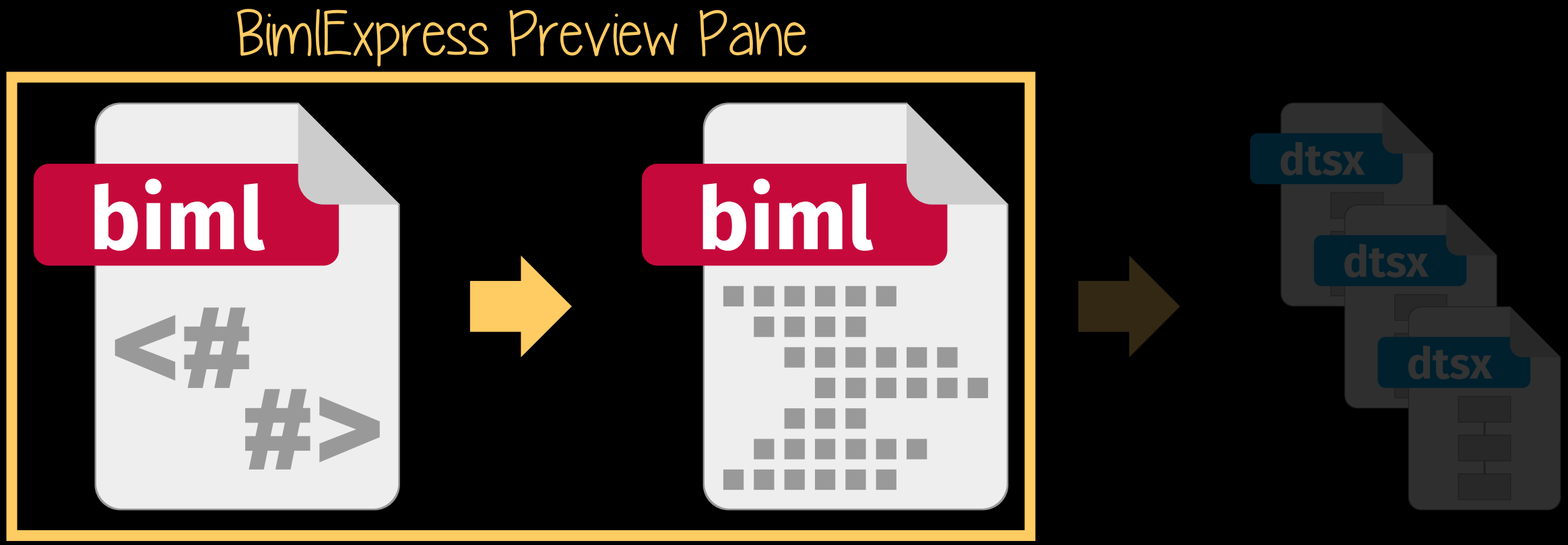
TRADITIONAL BIMLSCRIPT

```
<# foreach (var table in RootNode.Tables) { #>
  <Package Name="Load_<#=table.Name#>" />
<# } #>
```

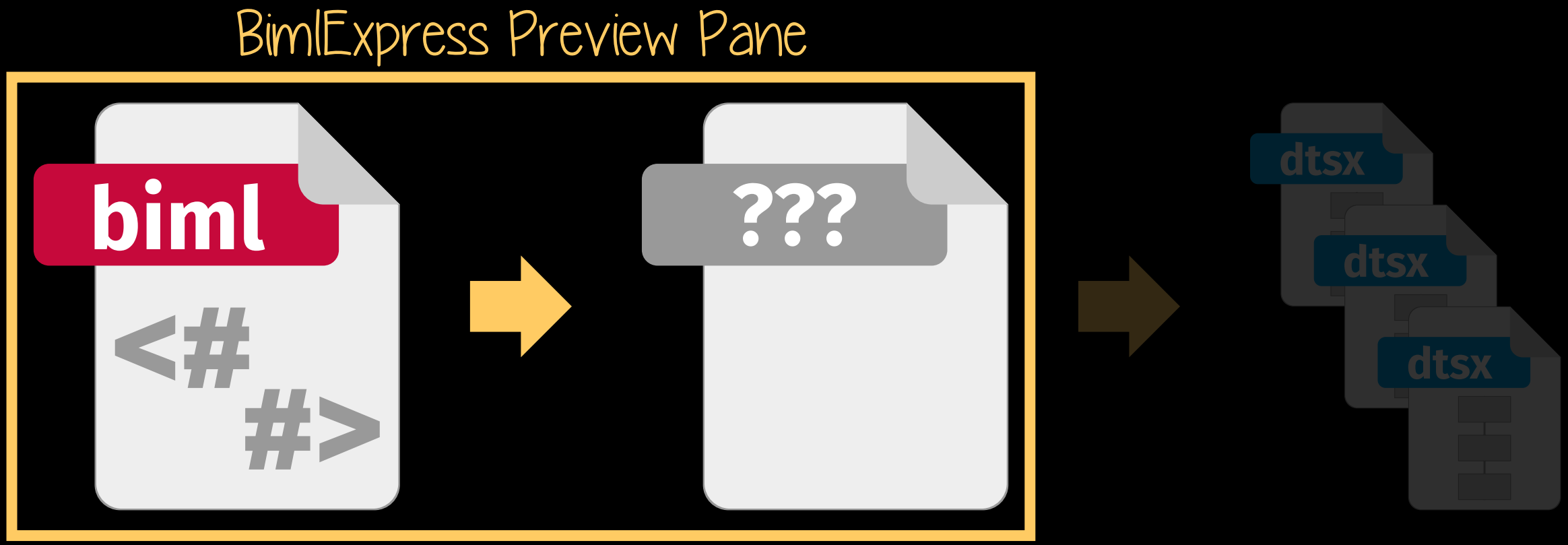
```
<Package Name="Load_Customer" />
<Package Name="Load_Product" />
<Package Name="Load_Sales" />
```



BIMLSCRIPT TO BIML



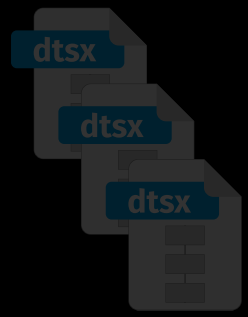
BIMLSCRIPT TO... ???



~~CRAZY~~ FUN BIMLSCRIPT

```
<# foreach (var table in RootNode.Tables) { #>
  Yay, a package! <#=table.Name#> :)
<# } #>
```

```
Yay, a package! Load_Customer :)
Yay, a package! Load_Product :)
Yay, a package! Load_Sales :)
```



T-SQL FROM BIML

GetColumnList()

GetColumnAssignmentList()

GetColumnComparisonList()

BIML COLUMN METHODS

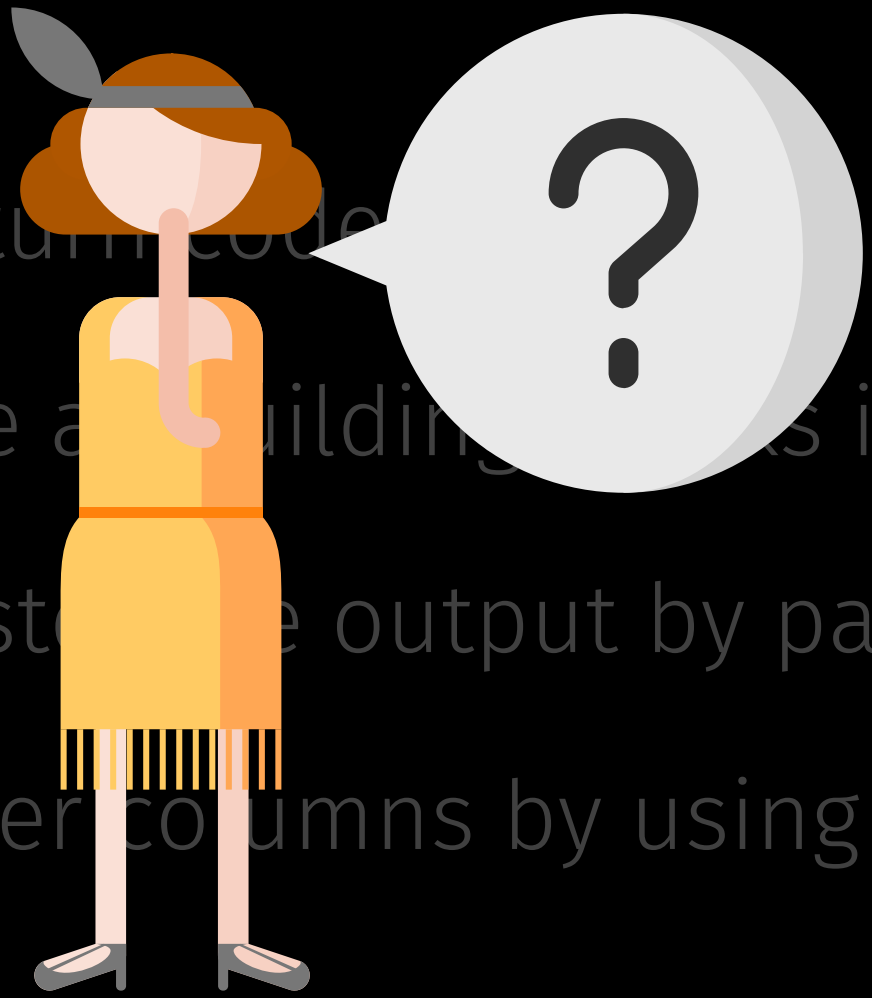
Return code fragments

Use as building blocks in custom T-SQL

Customize output by passing parameters

Filter columns by using lambda expressions

COLUMN METHODS



Return code

Use as building blocks in custom T-SQL

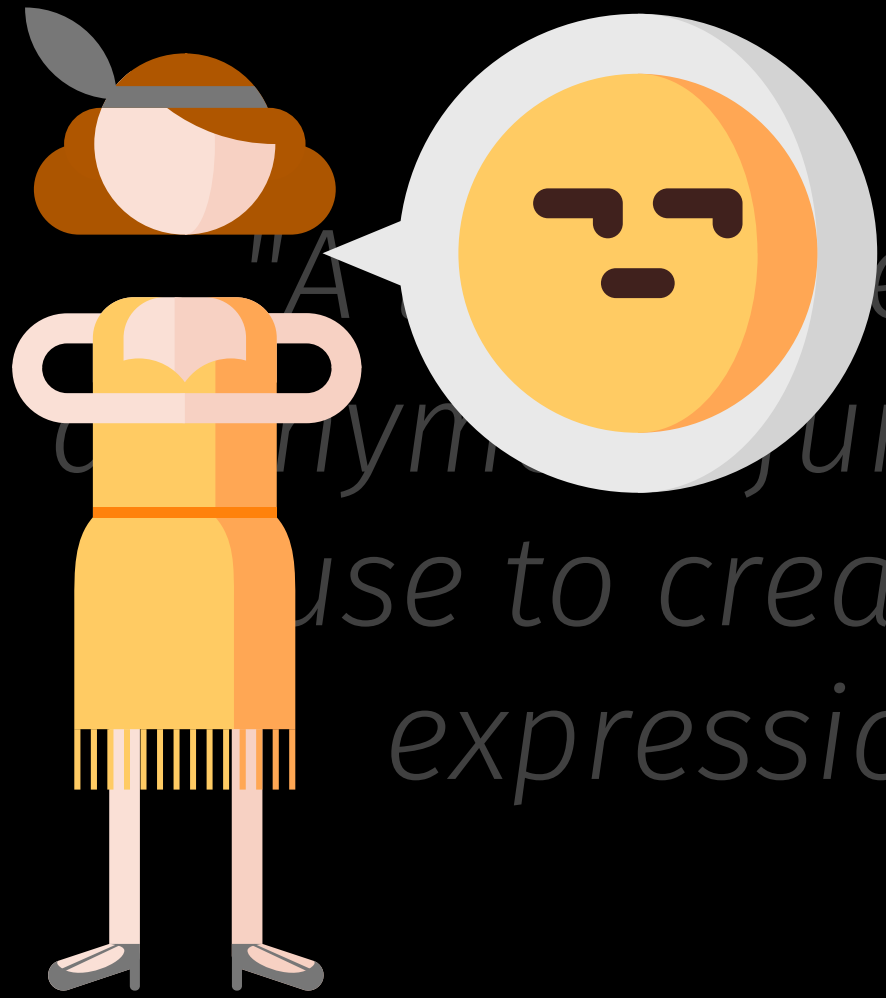
Customize output by passing parameters

Filter columns by using lambda expressions

LAMBDA EXPRESSIONS

"A lambda expression is an anonymous function that you can use to create delegates or expression tree types"

LAMBDA EXPRESSIONS



"A lambda expression is an anonymous function that you can use to create delegates or expression tree types"

LAMBDA EXPRESSIONS

```
column => column.Name == "ColumnID"
```

LAMBDA EXPRESSIONS

```
column => column.Name == "ColumnID"
```

The arrow is the lambda operator

LAMBDA EXPRESSIONS

```
column => column.Name == "ColumnID"
```

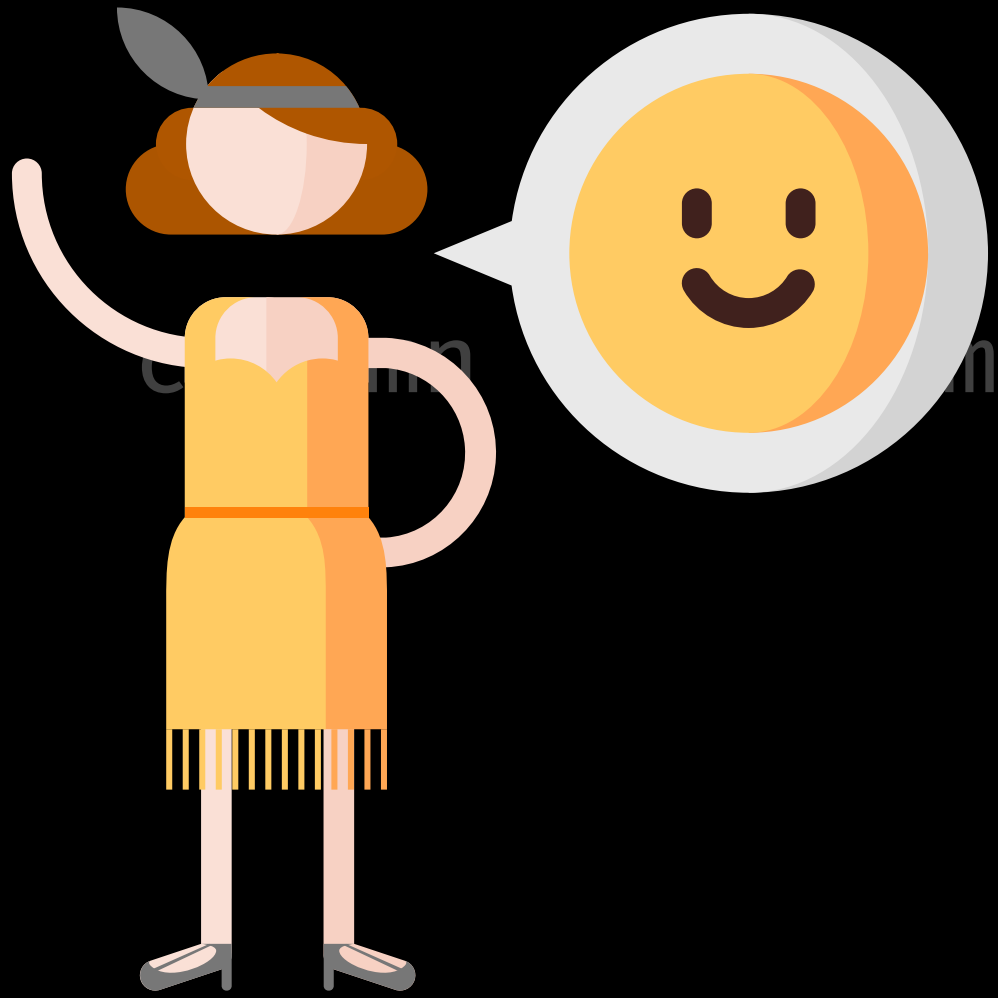
Input parameter is on the left side

LAMBDA EXPRESSIONS

```
column => column.Name == "ColumnID"
```

Expression is on the right side

LAMBDA EXPRESSIONS



```
c    mn    nn.Name == "ColumnID"
```

Get columns for **SELECT**:

```
GetColumnList()
```

```
GetColumnList()
```

```
[Col1], [Col2], [Col3]
```

```
GetColumnList("src")
```

```
[src].[Col1], [src].[Col2], [src].[Col3]
```

Get columns for **JOIN … ON**:

```
GetColumnComparisonList()
```

```
GetColumnComparisonList()
```

```
    [l].[Col1] = [r].[Col1]
AND [l].[Col2] = [r].[Col2]
AND [l].[Col3] = [r].[Col3]
```

```
GetColumnComparisonList("!=")
```

```
    [l].[Col1] != [r].[Col1]
AND [l].[Col2] != [r].[Col2]
AND [l].[Col3] != [r].[Col3]
```

```
GetColumnComparisonList("src", "dst")
```

```
    [src].[Col1] = [dst].[Col1]
AND [src].[Col2] = [dst].[Col2]
AND [src].[Col3] = [dst].[Col3]
```

```
GetColumnComparisonList
("!=", "src", "dst")
```

```
    [src].[Col1] != [dst].[Col1]
AND [src].[Col2] != [dst].[Col2]
AND [src].[Col3] != [dst].[Col3]
```

```
GetColumnComparisonList
("!=", "src", "dst", " OR " )
```

```
    [src].[Col1] != [dst].[Col1]
OR [src].[Col2] != [dst].[Col2]
OR [src].[Col3] != [dst].[Col3]
```

GETCOLUMNCOMPARISONLIST

```
GetColumnComparisonList()

GetColumnComparisonList("!=")

GetColumnComparisonList("dst", "src")

GetColumnComparisonList(c => c.IsUsedInPrimaryKey)

GetColumnComparisonList(c => c.IsUsedInPrimaryKey, "dst", "src")

GetColumnComparisonList(c => c.IsUsedInPrimaryKey, "!=", "dst", "src")
```

Get columns for **UPDATE … SET**:

```
GetColumnAssignmentList()
```

```
GetColumnAssignmentList()
```

```
 [l].[Col1] = [r].[Col1]
,[l].[Col2] = [r].[Col2]
,[l].[Col3] = [r].[Col3]
```

```
GetColumnAssignmentList("src", "dst")
```

```
 [src].[Col1] = [dst].[Col1]
,[src].[Col2] = [dst].[Col2]
,[src].[Col3] = [dst].[Col3]
```

DEMO

T-SQL FROM BIML

TEST DATA

BIML AND BOGUS

Create Random and Specific Test Data

BOGUS PROJECT

Simple and sane fake data generator for .NET

https://github.com/bchavez/Bogus

Advanced functionality for custom objects

...or simple functionality for Biml hacks :)

BOGUS API

Supports all data types plus built-in test data, including:

Address	Date	Name
Commerce	Finance	Person
Company	Internet	Phone
Database	Lorem	System

BOGUS METHODS

```
<#@ assembly name="Bogus.dll" #>
<#@ import namespace="Bogus" #>

<# var f = new Faker(); #>
<# Person p = new Person(); #>

INSERT INTO dbo.Employee(FirstName, LastName, Birthday) VALUES
(
  '<#=p.FirstName#>',
  '<#=p.LastName#>',
  '<#=p.DateOfBirth.ToString("yyyyMMdd")#>'
)
```

BOGUS INSTALLATION

Install via Nuget:

```
Install-Package Bogus
```

Or download latest release .zip:

https://github.com/bchavez/Bogus/releases

DEMO

TEST DATA WITH BOGUS

STATIC DIMENSIONS

STATIC DIMENSIONS

SCD Type 0: Not changing

- Unknown dimension members
- Date dimension
- Code tables

STATIC DIMENSION CREATION

1. Create table definition
2. Add static source rows
3. Generate INSERT statements

BIML STATIC SOURCES

Part of `<Table>` objects

Defines rows to be inserted when table is created

BIML STATIC SOURCE

```
<Table Name="Table" SchemaName="Database.Schema">
  <Columns>
    <Column Name="Column1" DataType="Int32" />
    <Column Name="Column2" DataType="String" Length="10" />
  </Columns>
  <Sources>
    <StaticSource Name="TableRows">
            ...
    </StaticSource>
  </Sources>
</Table>
```

BIML STATIC SOURCE

```
<StaticSource Name="TableRows">
  <Rows>
    <Row>
      <ColumnValues>
        <ColumnValue ColumnName="Col1" Value="-1" />
        <ColumnValue ColumnName="Col2" Value="'N/A'" />
      </ColumnValues>
    </Row>
  </Rows>
</StaticSource>
```

BIML STATIC SOURCE

```
<# if (table.Sources.Any()) { #>

  <#=TableToPackageLowerer.GetStaticSourceInsertStatements(
    table.SchemaQualifiedName,
    table.HasIdentity,
    (AstTableStaticSourceNode)table.Sources.FirstOrDefault()
  )#>

<# } #>
```

DEMO

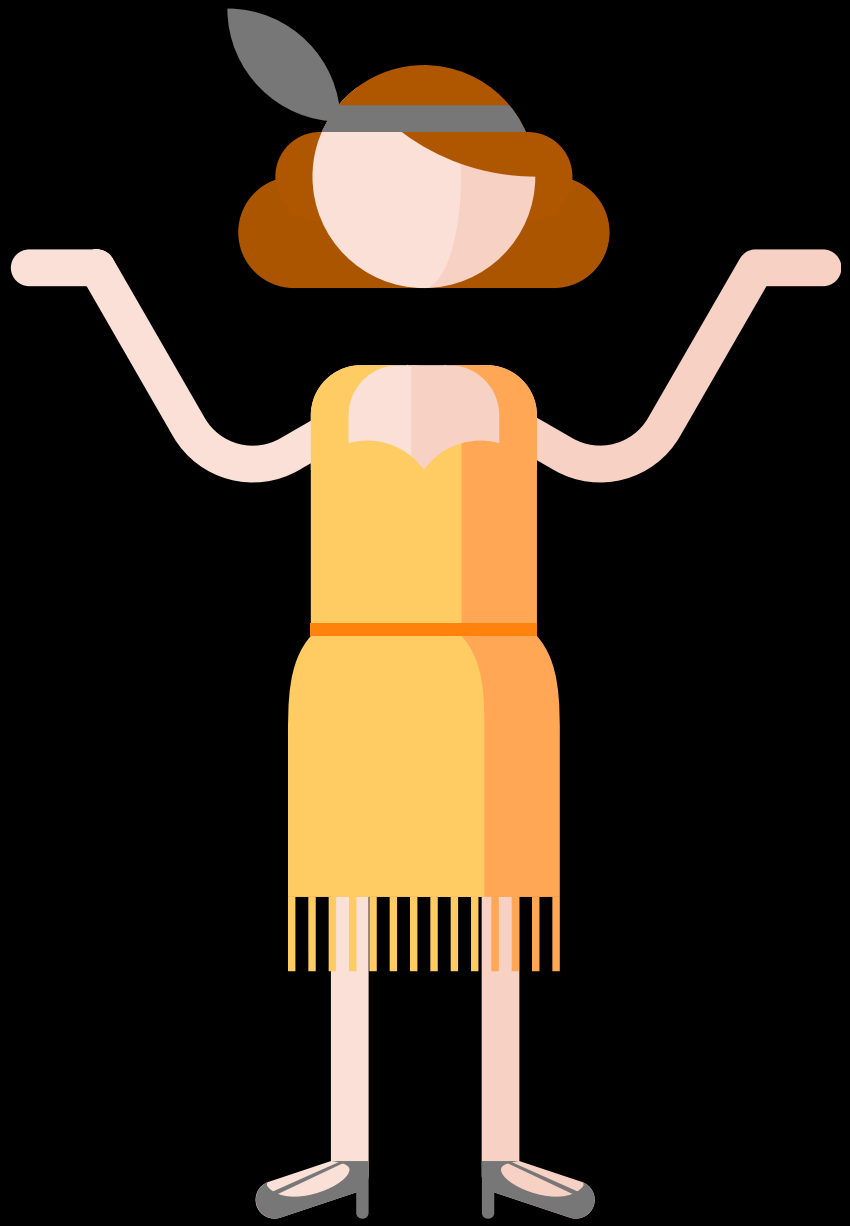
STATIC DIMENSIONS

THE PAST 50 MINUTES

T-SQL

Test Data

Dimensions



Is this useful?

What other ideas do you have?

Have fun!

Biml resources and demo files:

cathrinew.net/biml

hi@cathrinew.net

@cathrinew

CW cathrinew.net