



# Table of Contents

[BimlExpress Documentation](#)

[BimlExpress Getting Started](#)

[Installing BimlExpress](#)

[Creating BimlExpress Account](#)

[BimlExpress Menus and Options](#)

[Adding and Editing Biml files](#)

[Preview Expanded Biml](#)

[Generating SSIS Packages](#)

[BimlExpress Comparisons](#)

[BimlExpress vs BimlStudio](#)

[BimlExpress vs BIDS Helper vs BI Developer Extensions](#)

[Release Notes](#)

[BimlExpress 2018](#)

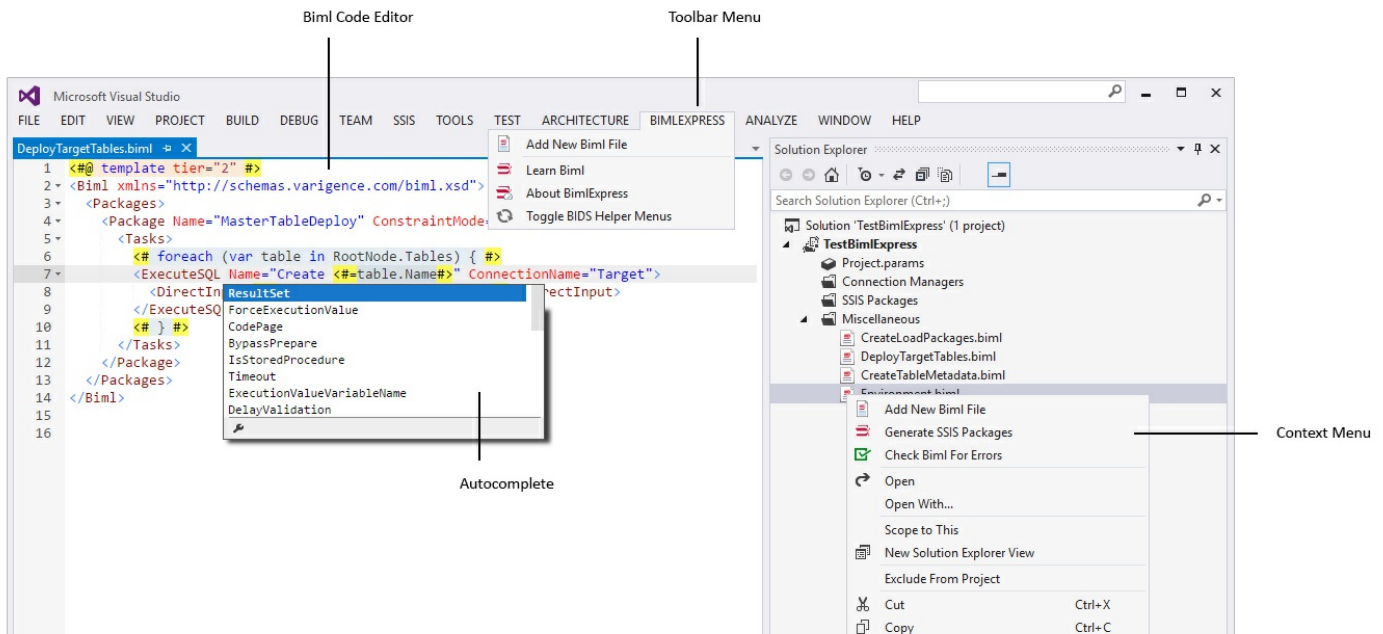
[What's New in BimlExpress 2017?](#)

[BimlExpress 2017](#)

# BimlExpress Documentation

## BimlExpress

BimlExpress is a Visual Studio extension that provides the ability to create Biml files, edit them in a custom code editor, and generate SSIS packages from Biml.



BimlExpress provides the ability to create packages from Biml, an XML-based language that allows you to describe your BI solution in a declarative fashion. This is similar to using HTML to describe how a web page should appear. In addition, you can embed BimlScript (C# or VB.NET code) into Biml, in the same way that ASP.NET works with HTML. This allows for large numbers of packages to be created with a minimal amount of code.

## Resources

Learn Biml on [BimlScript.com](http://BimlScript.com)

[Biml Language Reference](#)

[Biml API Reference](#)

## Help and Support

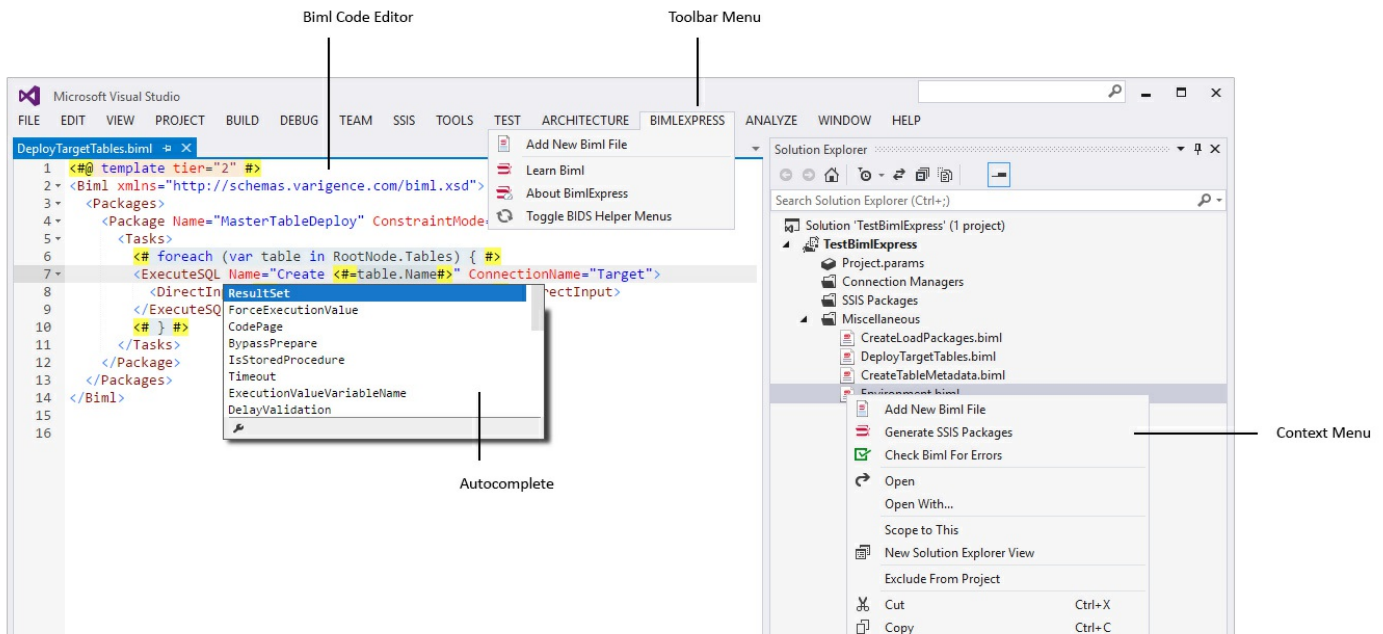
[BimlExpress Forums](#)

[Contact Varigence Support](#)

# BimlExpress Documentation

## BimlExpress

BimlExpress is a Visual Studio extension that provides the ability to create Biml files, edit them in a custom code editor, and generate SSIS packages from Biml.



BimlExpress provides the ability to create packages from Biml, an XML-based language that allows you to describe your BI solution in a declarative fashion. This is similar to using HTML to describe how a web page should appear. In addition, you can embed BimlScript (C# or VB.NET code) into Biml, in the same way that ASP.NET works with HTML. This allows for large numbers of packages to be created with a minimal amount of code.

## Resources

Learn Biml on [BimlScript.com](http://BimlScript.com)

[Biml Language Reference](#)

[Biml API Reference](#)

## Help and Support

[BimlExpress Forums](#)

[Contact Varigence Support](#)

# Installing BimlExpress

## Download BimlExpress

Download the latest `.vsix` file from [varigence.com/bimlexpress](http://varigence.com/bimlexpress).

## Installing

Select which version of Visual Studio you want to install and click `Install`:

VSIX Installer

**BimlExpress**

Select the product(s) you want to install the extension to:

☒ Microsoft Visual Studio 2015 Shell (Integrated)

This extension does not contain a digital signature.

Varigence, Inc.  
End User Software License Agreement  
NOTICE: THIS AGREEMENT CONTAINS IMPORTANT TERMS AND  
CONDITIONS THAT REQUIRE YOUR COMPLIANCE. CAREFULLY READ  
THE ENTIRE DOCUMENT.  
This Agreement describes the license terms between Varigence, Inc.

By clicking "Install", you agree with the  
above license terms (if any).

Install

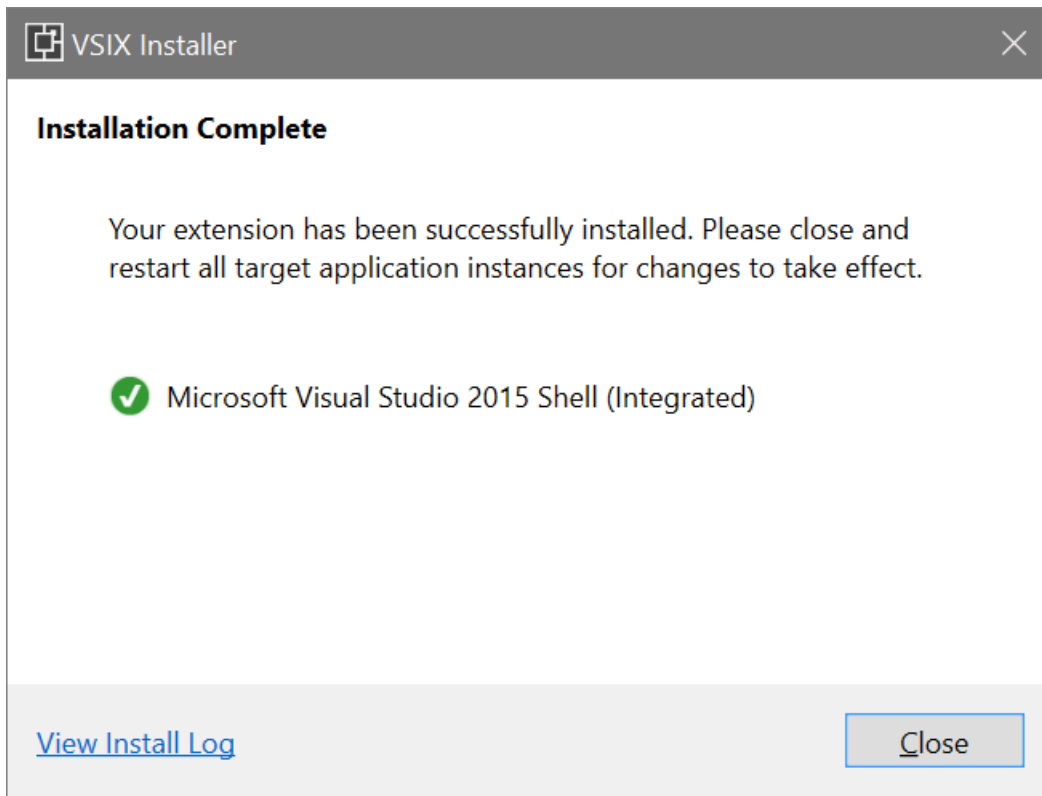
Cancel

VSIX Installer

**Installing...**

Status: Microsoft Visual Studio 2015 Shell (Integrated)

Cancel



## Product Key

Once you have installed BimlExpress, you need to enter your product key. Read more about why you need a product key on the [Creating BimlExpress Account](#) page.

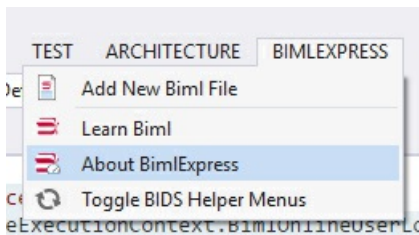
# Creating BimlExpress Account

## Why do I need an account for a free product?

Whenever we talk to Microsoft or other potential partners about doing things that would benefit Biml users, the very first question we get is, "How many Biml users are there?" We basically can't tell them. We know how many BimlStudio users we have. We know how many BIDSHelper downloads there are. Neither is a good measure of the Biml user base, though. Some BIDSHelper users don't use Biml. Some BIDSHelper downloads are put on a company share for hundreds of developers and consultants to use. Some users download BIDSHelper many times as they reinstall often. The list goes on and on. We don't want to be nosey, but we do think that there is a middle ground to get better user counts if we include user accounts.

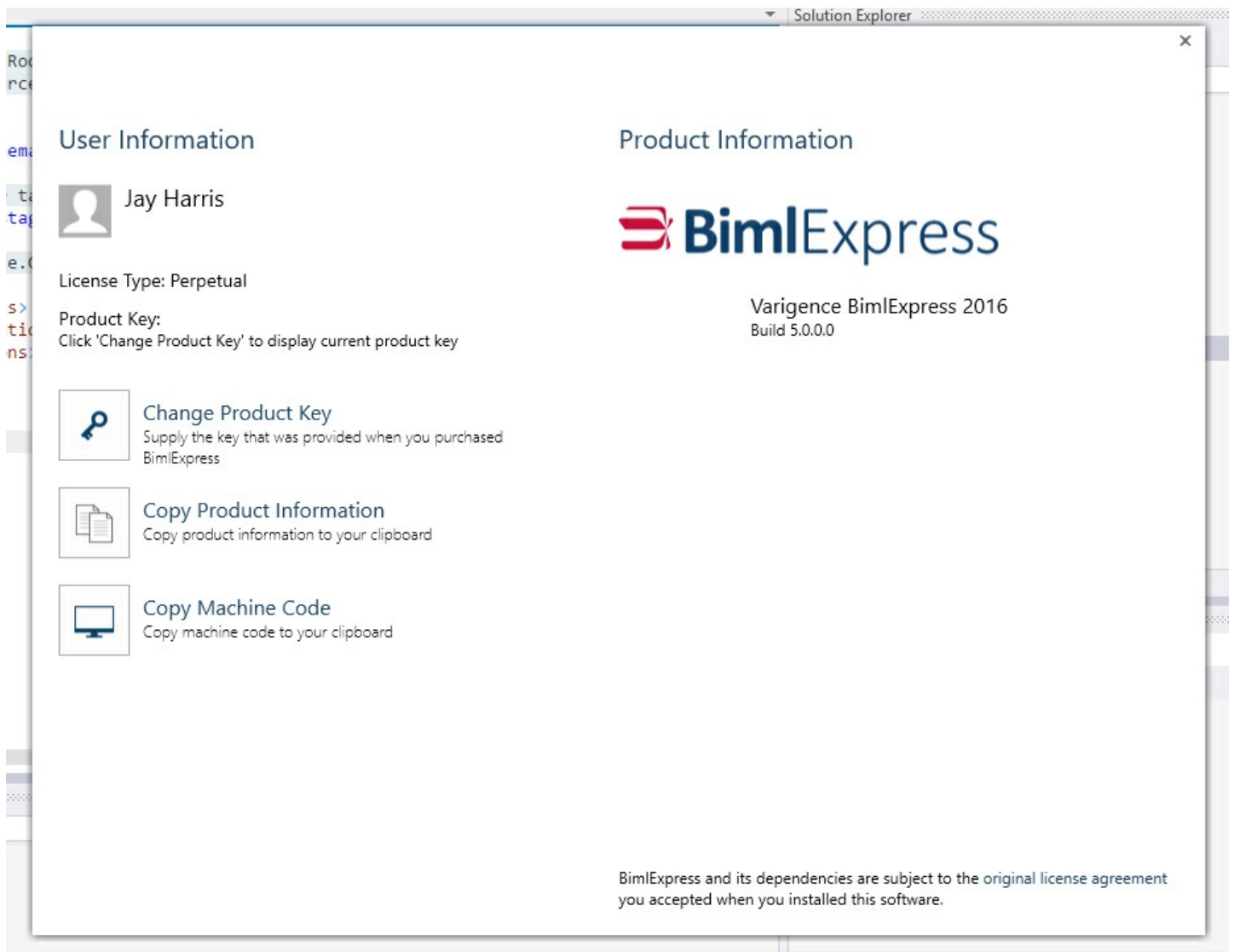
## Account Features

BimlExpress creates a toolbar menu in Visual Studio, which allows you to view your account settings by clicking **About BimlExpress**.

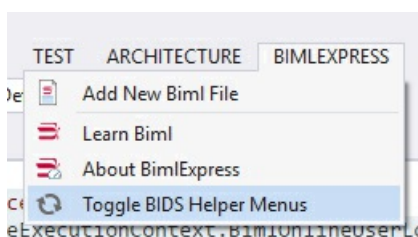


This opens a dialog that allows you to access the following features:

- **Change your product key:** If you receive a new product key, enter it here. You may need to restart Visual Studio for any additional features to apply.
- **Copy your product information:** This button copies information about the BimlExpress version and build to your clipboard. This information is often needed when answering support requests. To access this, paste it into an email or text file.
- **Copy your machine code:** This button copies your unique machine code to your clipboard. To access this information, paste it into an email or text file.



By default, if you already have BIDS Helper installed, BimlExpress removes the Biml-specific menu options from BIDS Helper and replaces them with BimlExpress functionality. To undo this, click **Toggle BIDS Helper Menus** in the BimlExpress toolbar and restart Visual Studio.







# BimlExpress Menus and Options


## BimlExpress Toolbar


From the BimlExpress Toolbar, you can add files to your project, learn more about Biml, find information about BimlExpress and your product key, toggle the BIDS Helper (*BI Developer Extensions*) menus if you have both add-ins installed, and customize the BimlExpress options:


BimlExpress


 Add New Biml File Ctrl+Shift+N, Ctrl+Shift+B


 Add New C# File Ctrl+Shift+N, Ctrl+Shift+C


 Add New VB File

 Learn Biml

 About BimlExpress

 Toggle BIDS Helper Menus







 Disable .NET Autocomplete

 Options

## BimlExpress Buttons


You can also add the BimlExpress buttons to your toolbar for easy access to generating SSIS packages, checking Biml for errors, and adding new files:

BimlExpress SQL Prompt Analyze Window Help

Analysis Services


Application Insights

 BimlExpress







Build

## BimlExpress Context Menu

If you right-click on the SSIS Project, you will be able to add new files:

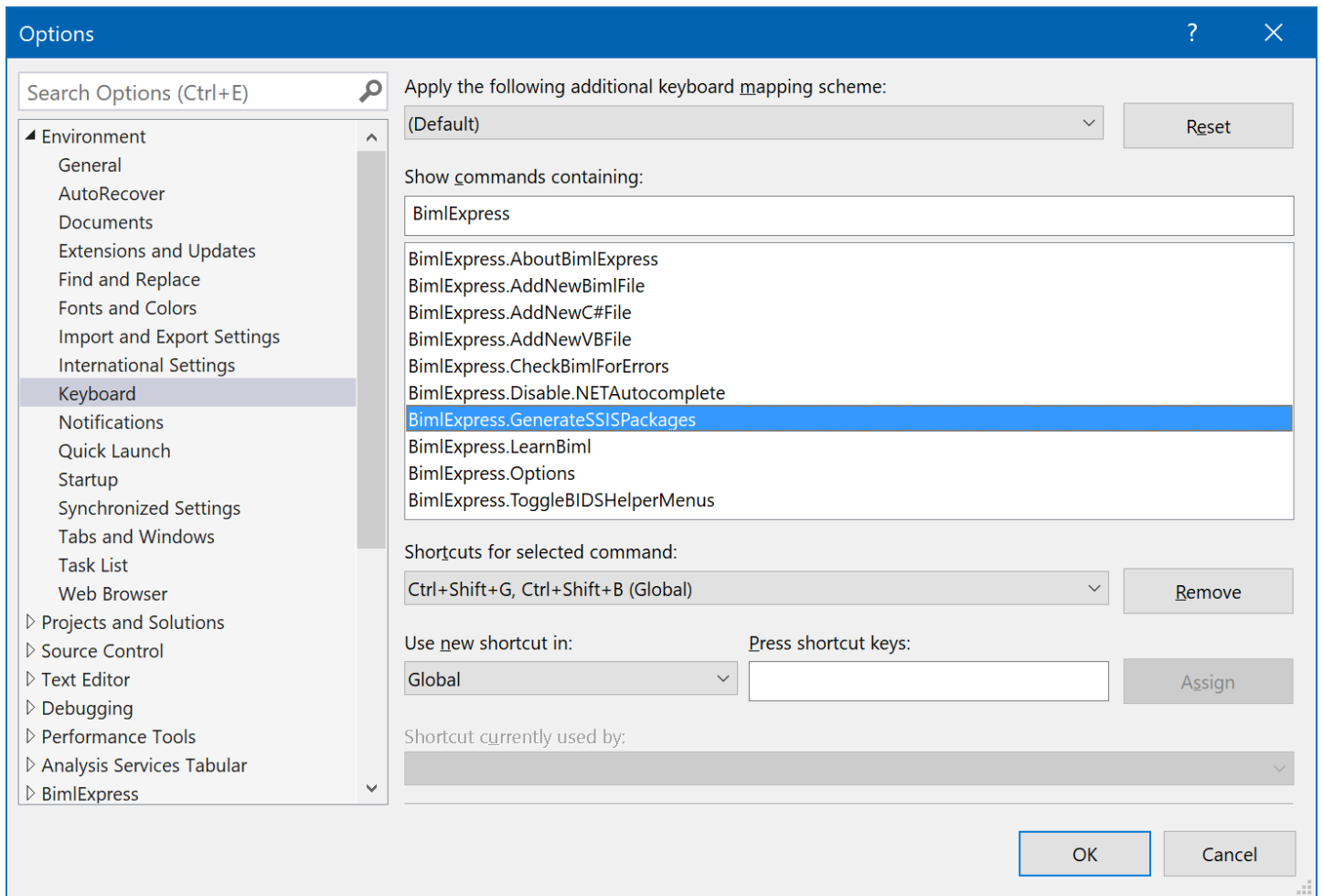
Deploy		
	Add New Biml File	Ctrl+Shift+N, Ctrl+Shift+B
	Add New C# File	Ctrl+Shift+N, Ctrl+Shift+C
	Add New VB File	
	Build	
	Rebuild	

If you right-click on a Biml file, you will be able to generate SSIS packages and check Biml for errors, in addition to adding new files:

	Generate SSIS Packages	Ctrl+Shift+G, Ctrl+Shift+B
	Check Biml For Errors	Ctrl+Shift+C, Ctrl+Shift+B
	Add New Biml File	Ctrl+Shift+N, Ctrl+Shift+B
	Add New C# File	Ctrl+Shift+N, Ctrl+Shift+C
	Add New VB File	
	Open	
	Open With...	

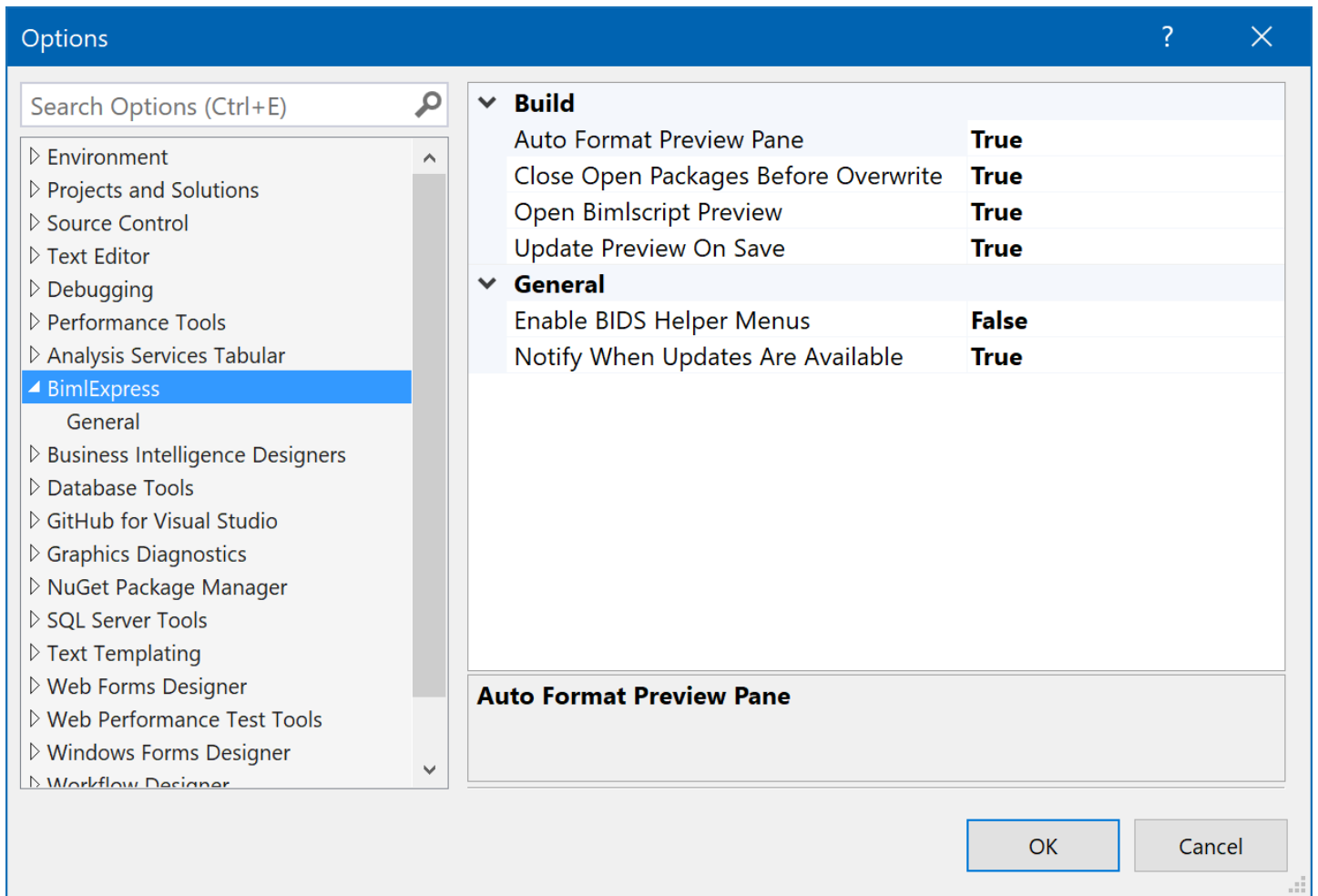
## Customizing Keyboard Shortcuts

If you prefer to use Keyboard Shortcuts instead of menus and buttons, you can assign your own shortcuts to BimlExpress:



## BimlExpress Options

BimlExpress has several options that you can customize:



**Auto Format Preview Pane:** This option enables or disables the output in the Preview Pane

**Close Open Packages Before Overwrite:** This option enables or disables closing open packages when you re-generate them

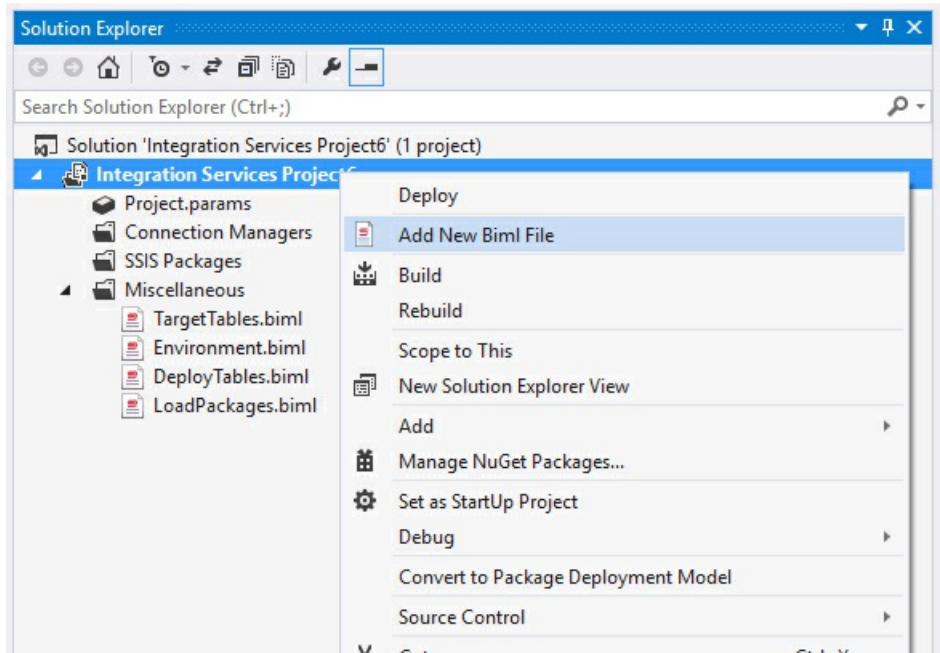
**Open BimlScript Preview:** This option enables or disables whether the Preview Pane opens by default when opening a Biml file

**Update Preview On Save:** This option enables or disables whether the Preview Pane is automatically updated when a Biml file is saved

# Adding and Editing Biml files

To add a Biml file to the project, right-click on the project, or on the Data Sources, Data Source Views, or SSIS Packages folders. The context menu will have an **Add New Biml File** menu option. This will add a Biml file to the Miscellaneous folder in the project.

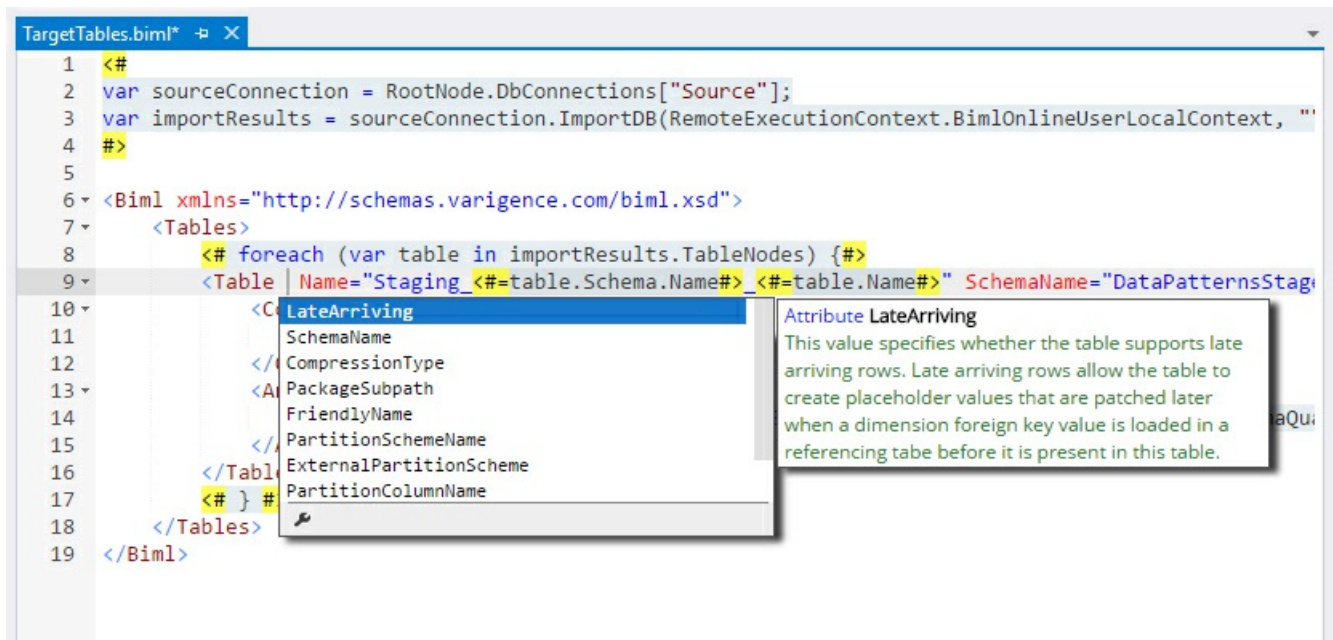
Note: Right-clicking on the Miscellaneous folder will not open a context menu.



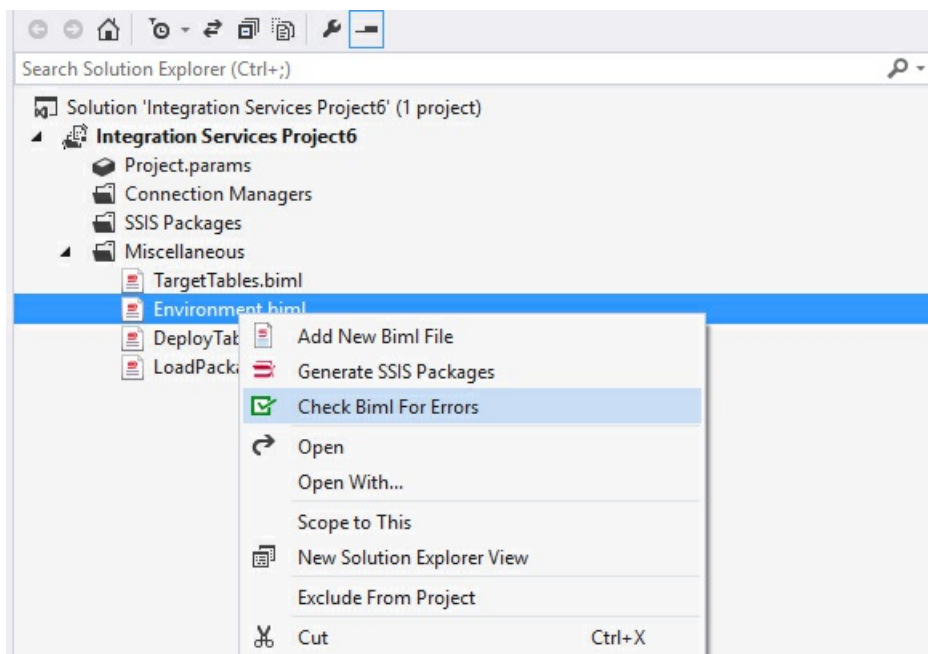
After adding a Biml file, double-click to open it in the BimlExpress code editor, which includes syntax highlighting for Biml and BimlScript code nuggets, autocomplete, and error highlighting.



Autocomplete can be opened by pressing **Ctrl+Space** while editing a Biml tag.

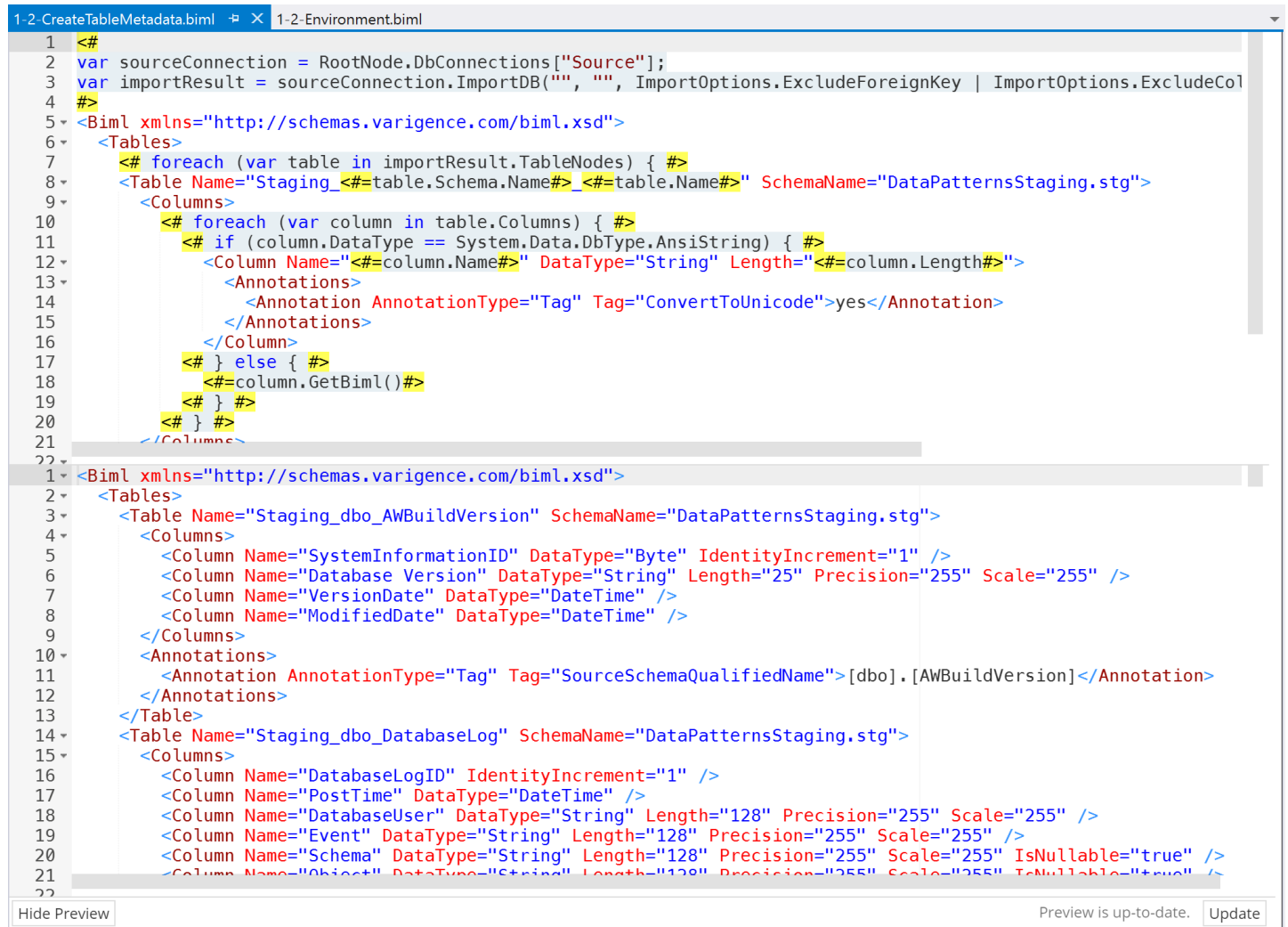


After adding Biml code to a file, it can be checked for errors by choosing the Check Biml for Errors option from the context menu.



# Preview Expanded Biml

The Preview Pane was introduced in BimlExpress 2017. It lets you preview the expanded BimlScript, making development and debugging much easier:



```
1-2- CreateTableMetadata.biml 1-2- Environment.biml
1 <#
2 var sourceConnection = RootNode.DbConnections["Source"];
3 var importResult = sourceConnection.ImportDB("", "", ImportOptions.ExcludeForeignKey | ImportOptions.ExcludeCol
4 #>
5 <Biml xmlns="http://schemas.varigence.com/biml.xsd">
6   <Tables>
7     <# foreach (var table in importResult.TableNodes) { #>
8       <Table Name="Staging_<#=table.Schema.Name#>_<#=table.Name#>" SchemaName="DataPatternsStaging.stg">
9         <Columns>
10          <# foreach (var column in table.Columns) { #>
11            <# if (column.DataType == System.Data.DbType.AnsiString) { #>
12              <Column Name="<#=column.Name#>" DataType="String" Length="<#=column.Length#>">
13                <Annotations>
14                  <Annotation AnnotationType="Tag" Tag="ConvertToUnicode">yes</Annotation>
15                </Annotations>
16              </Column>
17            <# } else { #>
18              <#=column.GetBiml()#>
19            <# } #>
20          <# } #>
21        </Columns>
22      </Table>
23    </Tables>
24  </Biml>
25</pre>

```
1 <Biml xmlns="http://schemas.varigence.com/biml.xsd">
2   <Tables>
3     <Table Name="Staging_dbo_AWBuildVersion" SchemaName="DataPatternsStaging.stg">
4       <Columns>
5         <Column Name="SystemInformationID" DataType="Byte" IdentityIncrement="1" />
6         <Column Name="Database Version" DataType="String" Length="25" Precision="255" Scale="255" />
7         <Column Name="VersionDate" DataType="DateTime" />
8         <Column Name="ModifiedDate" DataType="DateTime" />
9       </Columns>
10      <Annotations>
11        <Annotation AnnotationType="Tag" Tag="SourceSchemaQualifiedNames">[dbo].[AWBuildVersion]</Annotation>
12      </Annotations>
13    </Table>
14    <Table Name="Staging_dbo_DatabaseLog" SchemaName="DataPatternsStaging.stg">
15      <Columns>
16        <Column Name="DatabaseLogID" IdentityIncrement="1" />
17        <Column Name="PostTime" DataType="DateTime" />
18        <Column Name="DatabaseUser" DataType="String" Length="128" Precision="255" Scale="255" />
19        <Column Name="Event" DataType="String" Length="128" Precision="255" Scale="255" />
20        <Column Name="Schema" DataType="String" Length="128" Precision="255" Scale="255" IsNullable="true" />
21        <Column Name="Object" DataType="String" Length="128" Precision="255" Scale="255" IsNullable="true" />
22      </Columns>
23    </Table>
24  </Tables>
25</Biml>
```



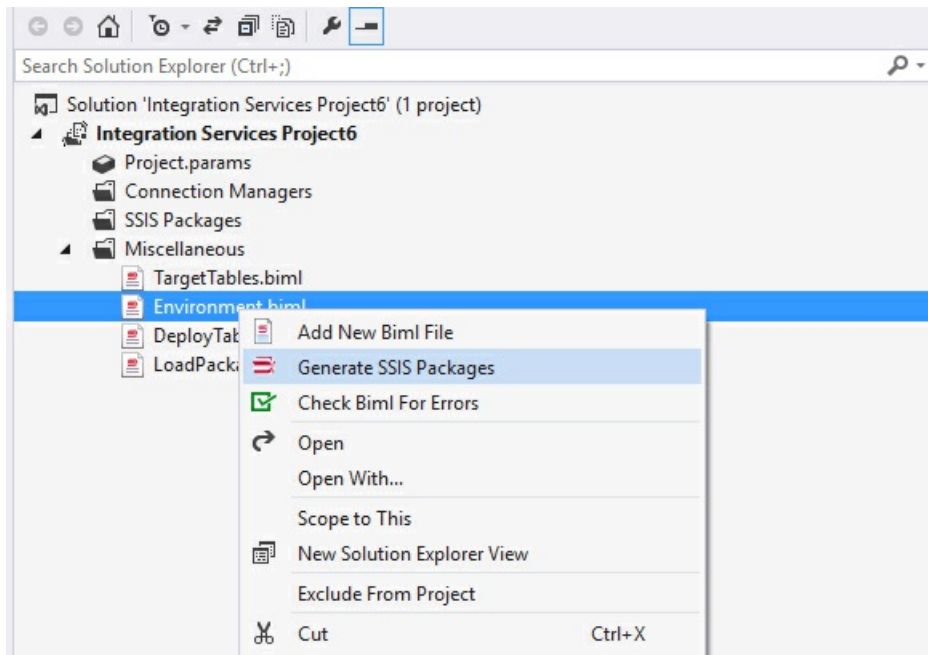
Hide Preview Preview is up-to-date. Update


```

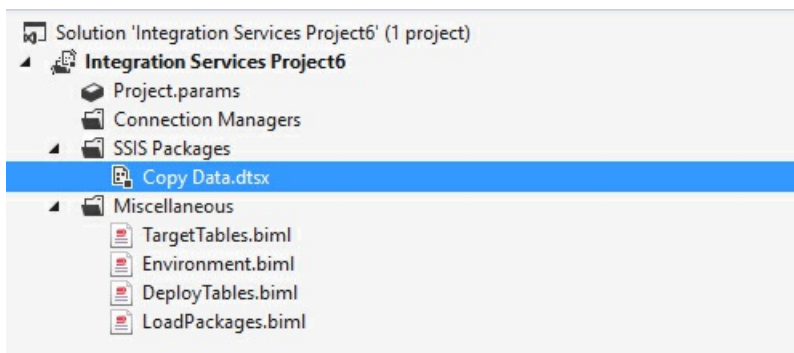
When working with tiered Biml files, make sure to open the files in the correct order to ensure that the preview pane shows all the code.

# Generating SSIS Packages

A Biml file can be expanded into one or more SSIS packages by choosing **Generate SSIS Packages** from the menu. Expanding a file will automatically run an error check.



If the Biml file was expanded successfully, it will be added to the SSIS Packages folder in the project.





# BimlExpress Comparison

## BimlExpress

BimlExpress is a Visual Studio extension that provides the ability to create Biml files, edit them in a custom code editor, and generate SSIS packages from Biml.

In this section the differences between BimlExpress, BimlStudio, BIDS Helper and BI Developer Extensions are detailed.

## Resources

Learn Biml on [BimlScript.com](http://BimlScript.com)

[Biml Language Reference](#)

[Biml API Reference](#)

## Help and Support

[BimlExpress Forums](#)

[Contact Varigence Support](#)

# Comparing BimlExpress and BimlStudio

BimlExpress is Varigence's free Visual Studio add-in, created specifically for working on Biml projects.

BimlStudio is Varigence's full-featured integrated development environment (IDE) for Biml, created to replace Visual Studio.

FEATURE	BIMLEXPRESS	BIMLSTUDIO
Relational Databases		
Model relational database objects	X	X
Produce DDL for any ANSI SQL compliant RDBMS	X	X
Import relational database objects from Bimlscript code	X	X
Import relational database objects using visual UI		X
Integration Services (SSIS)		
Model SSIS Packages	X	X
Generate packages (DTSX files)	X	X
Generate SSIS projects (DTProj, Params, ISPAC files)		X
Import SSIS packages and projects		X
Easier use of Feature Packs & 3rd party Task/Components		X
Analysis Services		
Model SSAS Cubes		X
Model SSAS Tabular and PowerPivot		X
Generate SSAS cubes, tabular models, and PowerPivot		X
Import cubes, tabular models, and PowerPivot		X
Scripting		
Automate Biml with BimlScript code nuggets	X	X
Customize validation with your own errors and warnings	X	X

FEATURE	BIMLEXPRESS	BIMLSTUDIO
Use Transformers to modify objects and inject patterns		X
Organize Transformers and Bimlscripts into reusable Frameworks		X
<b>User Interface</b>		
BimlScript code editor with advanced Intelliprompt	X	X
BimlScript code editor with C# & VB code completion		X
Visual Designers for all Biml objects		X
Live view of objects as you code		X
<b>Automation</b>		
Command line compiler		X
MSBuild support		X
API Access from external applications		X
<b>Extensibility</b>		
Extend the Biml Language		X
Extend Mist UI		X
Redistribute BimlEngine binaries (OEM)		X
Host Bimlengine binaries in a value-added service (SaaS)		X
<b>Metadata Management</b>		
Create reusable models for metadata storage		X
Autogeneration of metadata entry UI based on metadata models		X
Consistent and friendly API for accessing stored metadata		With separate agreement
Plugin model allows storage of metadata anywhere		With separate agreement

# Comparing BimlExpress and BIDS Helper / BI Developer Extensions

Note: In August 2017, BIDS Helper was renamed to [BI Developer Extensions](#). You may still see references to BIDS Helper in menus and documentation.

BI Developer Extensions is a free Visual Studio add-in that enhances development functionality in Business Intelligence Development Studio (BIDS) and SQL Server Data Tools (SSDT). It includes the Biml engine, but also includes many other features.

BimlExpress is Varigence's free Visual Studio add-in, created specifically for working on Biml projects. In addition to all the features found in BI Developer Extensions, BimlExpress also provides a full BimlScript code editor with advanced Intellisense, support for adding code files from menus, and additional features can be enabled per product key.

FEATURE	BI DEVELOPER EXTENSIONS	BIMLEXPRESS
<b>Relational Databases</b>		
Model relational database objects	X	X
Produce DDL for any ANSI SQL compliant RDBMS	X	X
Import relational database objects from Bimlscript code	X	X
Import relational database objects using visual UI		
<b>Integration Services (SSIS)</b>		
Model SSIS Packages	X	X
Generate packages (DTSX files)	X	X
Generate SSIS projects (DTProj, Params, ISPAC files)		
Import SSIS packages and projects		
Easier use of Feature Packs & 3rd party Task/Components		
<b>Analysis Services</b>		
Model SSAS Cubes		
Model SSAS Tabular and PowerPivot		
Generate SSAS cubes, tabular models, and PowerPivot		
Import cubes, tabular models, and PowerPivot		

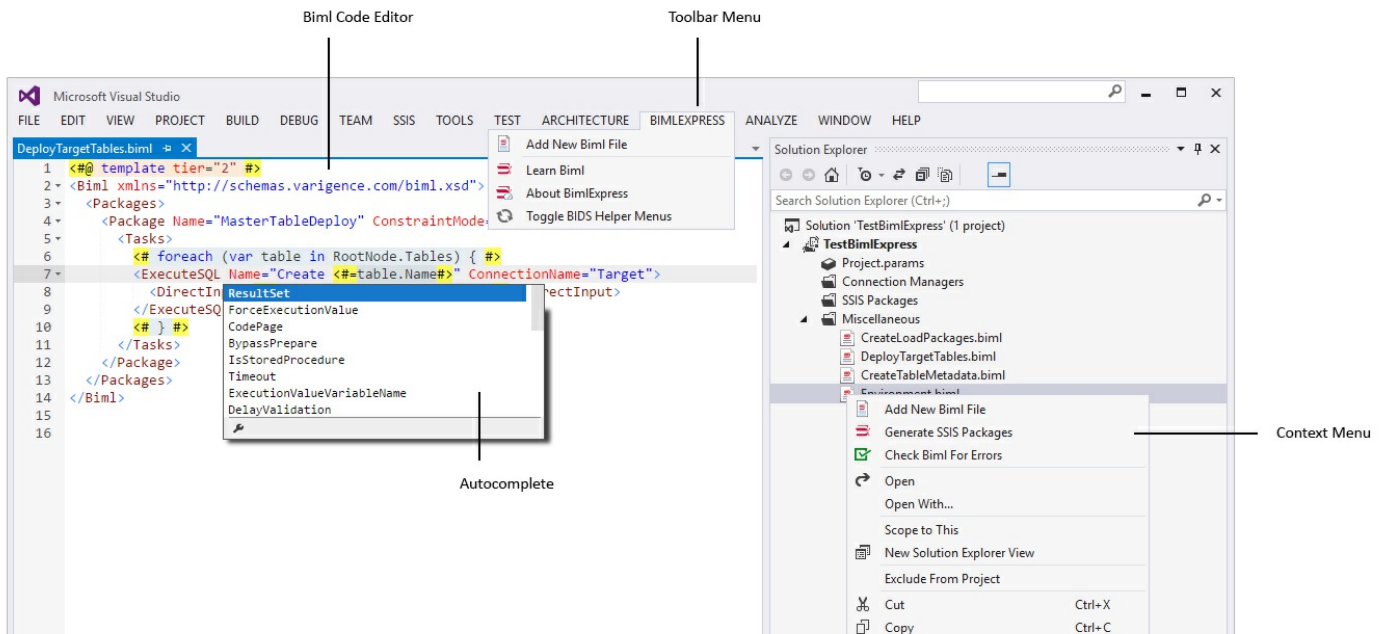
FEATURE	BI DEVELOPER EXTENSIONS	BIMLEXPRESS
<b>Scripting</b>		
Automate Biml with BimlScript code nuggets	X	X
Customize validation with your own errors and warnings	X	X
Use Transformers to modify objects and inject patterns		
Organize Transformers and Bimlscripts into reusable Frameworks		
<b>User Interface</b>		
BimlScript code editor with advanced Intelliprompt		X
BimlScript code editor with C# & VB code completion		
Visual Designers for all Biml objects		
Live view of objects as you code		
<b>Automation</b>		
Command line compiler		
MSBuild support		
API Access from external applications		
<b>Extensibility</b>		
Extend the Biml Language		
Extend Mist UI		
Redistribute BimlEngine binaries (OEM)		
Host Bimlengine binaries in a value-added service (SaaS)		
<b>Metadata Management</b>		
Create reusable models for metadata storage		

FEATURE	BI DEVELOPER EXTENSIONS	BIMLEXPRESS
Autogeneration of metadata entry UI based on metadata models		
Consistent and friendly API for accessing stored metadata		
Plugin model allows storage of metadata anywhere		

# BimlExpress Release Notes

## BimlExpress

BimlExpress is a Visual Studio extension that provides the ability to create Biml files, edit them in a custom code editor, and generate SSIS packages from Biml.



BimlExpress provides the ability to create packages from Biml, an XML-based language that allows you to describe your BI solution in a declarative fashion. This is similar to using HTML to describe how a web page should appear. In addition, you can embed BimlScript (C# or VB.NET code) into Biml, in the same way that ASP.NET works with HTML. This allows for large numbers of packages to be created with a minimal amount of code.

## Resources

Learn Biml on [BimlScript.com](http://BimlScript.com)

[Biml Language Reference](#)

[Biml API Reference](#)

## Help and Support

[BimlExpress Forums](#)

[Contact Varigence Support](#)

# BimlExpress 2018 Release Notes

Significant changes between BimlExpress 2017 and BimlExpress 2018

## New Features

- Added the Import Packages window to enable users to import existing SSIS packages and projects and convert them to Biml. The window is prepopulated with settings that are highly configurable to produce the desired results.
- Updates to BimlExpress project and manifest for compatibility
- Updated jquery in BimlExpress

## Bug Fixes

- Fixed an issue where exceptions thrown by user BimlScripts might not show up correctly in the BE preview pane
- Fixed issue where Ctrl+S would not save Biml files
- Fixed `No path to BIDS devenv for Ssis2016 was found.` when trying build and open with 2016 target
- Fixed a potential race condition on file saving in BimlExpress
- Fixed an assembly resolution issue that was breaking script projects in BimlExpress OneDesigner scenarios
- Added the `ICollection<T>` interface to `TierFilteredVulcanCollection`. This results in a larger number of extension methods working as expected in Live preview mode in BimlExpress.



# What's New

## What's New in BimlExpress 2018

You can also read the [full BimlExpress 2018 release notes](#)

### Convert SSIS Packages to Biml

You can now reverse-engineer and convert your SSIS packages to Biml directly in BimlExpress!

## What's New in BimlExpress 2017

You can also read the [full BimlExpress 2017 release notes](#)

### Preview Pane

BimlExpress 2017 introduced the Preview Pane, allowing you to preview expanded Biml for easier development and debugging.

### Global Include Files

With the new Global Include Files, you can now add things like global variables or code files to your entire project without having to use the include directive in each file.

### SSIS Annotations

BimlExpress 2017 introduced a feature to add SSIS annotations to your packages from Biml.

# BimlExpress 2017 Release Notes

## BimlExpress 2018

Release Date: ???

### Breaking Changes from Previous Release

???

### Major Features

- SSIS Package Importer

## BimlExpress 2017

Release Date: July 15th, 2017

### Breaking Changes from Previous Release

- Moved PathAnnotation from PrecedenceConstraints to TaskflowInputPath.
- Fixed issue in CallBimlScript property directives where all properties were required and marking them as required would create an error. To fix this, we apply all of the required properties first and then the optional properties in order of specification. Since required properties basically didn't work before, this shouldn't affect any actual code, but it should be called out as a potentially significant behavioral change.

### Major Features

- Expanded Biml Preview pane built into editor
- Biml global include files
- Improved Biml code editing experience
- New VSIX installer
- Improved VB Support
- Significantly improved runtime performance and reduced memory usage
- Multi-product support in product keys

### Platform

- Moved all DLLs to use .NET 4.6 rather than .NET 3.5
- All assemblies are now strong-named
- Updated EULA to cover frameworks including BimlFlex

### BimlExpress UI

- Improved Find/Replace within document
- Added option to disable autoformat in preview pane
- Significantly improved load time of existing Biml projects by eliminating upfront Biml file parsing.
- Fixed an issue where bulk opening multiple files and then closing them could cause a VS crash or an error when opening additional files
- Biml file icon is now registered by the add-in if it isn't already registered

### Biml Code Editor

- Fixed a large number of bugs
- Switched to better fonts for the completion list and quick info popup windows

- Improved quick info and code completion performance
- Improved automatic code insertions to enable a smoother coding workflow
- Fixed quick info and completion lists in include scenarios.
- Fixed completion lists in multi-line text nuggets.
- Made some additional properties Advanced to prevent them from showing up in intellisense completion lists
- Added preview mode of .NET intellisense for early adopters
- Fixed interaction between code editor and compare tool in VS

## Biml Language

- See [Biml Release Notes](#)

## Biml API

- See [Biml Release Notes](#)

## Database Providers and Type Mappings

- Added formal support for the DevArt MySql provider.
- Added formal support for Ingres ODBC and ADO.NET.
- Added formal support for SAS local data provider (e.g. sas7bdat files).
- Fixed issue where custom types from SQL Server were not being imported correctly.
- Added Db2 OLEDB provider.
- Added OpenEdge Odbc provider.
- Adding basic Excel ACE OLEDB support to GetDatabaseSchema.

## SSIS Components

- Added Biml elements for Azure Feature Pack, including:
  - Task details controls and images.
  - Models for connections and tasks.
  - SSIS emitters and importers.

## Visual Basic

- Updated BimlEngine so that dynamic objects will work with both VB and C#.
- Added VB code for metadata model dynamic objects.
- Removed default timeout arguments for all of the external data access APIs and replaced them with explicit method overloads. This is primarily to better support VB, which does not handle overloads with default arguments as elegantly as C#.

## Table Import

- Fixed issue where identity information was not being imported properly from SQL Server tables.
- Added support for more custom types in SQL Server table imports.
- Cleaned up precision and scale settings on table import for types that don't support them.
- Added option to insert raw provider annotations into table import results.
- Fixed bug where importing multiple tables with the same name in different schemas wouldn't work in GetDatabaseSchema.

## Javascript Emission

- Added GetJson method with several variants to every Biml object and collection.

## Biml Validator

- Fixed issue where validator thread would be interrupted if a table had multiple keys with the same name but no assigned

schema.

- Added a validator for AnalysisServicesProcessing task to ensure that the correct mix of properties is specified for each type of processing job.
- Updated SsisEmitter dataflow type mismatch error to actually list the mismatched types.
- Fixed an issue where user generated errors/warnings/notifications would not show up on initial project load in BimlStudio.

## GetBiml Updates

- Complete rewrite of GetBiml functionality that will have better performance and be more reliable.
- Fixed an issue with the emission of least qualified names for Biml object references.

## Package Updates

- Fixed static properties for several package tasks to match the latest options and data types.
- Fixing data types for some task nodes.
- Biml can be used to generate package parts.
- Package parts can now be referenced from Biml control flow.

## SSIS Emission

- Added support for SSIS annotations to packages, containers, event handlers, and data flows for SSIS 2012 and higher.
- Added support for dataflow and controlflow path annotations for SSIS 2012 and higher.
- Added error for SSIS 2016 and higher to not use the new virtual column names for errors: 'ErrorCode - Description', and 'ErrorColumn - Description'.
- Fixed an issue where BalancedDataDistributor output paths were not being placed into an exclusion group.
- Fixed issue where \_\_\$seqval column was being added incorrectly for CDC component outputs with CDC processing mode of Net.
- Fixed an issue where the Aggregate component would use the wrong data type when averaging certain column types.
- Fixed issue where package parameters with datetime type were not being emitted with the correct format into the project deployment manifest.
- Fixed issue where binary parameters were not being properly mapped in SQL queries (e.g. ExecuteSQL with parameters).
- Added support for the LEFT string operator to SSIS expression parsing.
- Added support for package configurations even when using project deployment model. This is not a recommended solution, but it is supported.
- Fixed an issue with permissions violations when deleting temp directories used in script project compilation in some corner cases.
- Added CachedComparisonFlags to input columns in SSIS emitter language definition.
- Fixed sorting of DTSX XML elements for 2012+.
- Fixed date mapping in ADO.NET Source.
- Fixed date and string mappings in OLEDB Source.
- Fixed date mapping in ADO.NET Destination.
- Fixed date and string mappings in OLEDB Destination.
- Fixed DontSaveSensitive support in OLEDB connection.
- Stopped emitting \_\_\$seqval as an external column in CDC Source when using an incompatible CDC Processing Mode.
- Fixed issue where format file references were being improperly emitted for BulkInsert tasks.
- Fixed issues with XmlSource where outputs were not being constructed correctly for unbounded elements.
- Fixed issue with CDC Control Task where emission would fail if StateConnection or StateName were not supplied.
- Fixes for SSIS Lookup type mappings for date and text.
- Fixed an issue where errors would be produced if you used a parameterized query in an SSIS component where the variable used to set the parameter contained an expression.
  - Fixed and NRE when parameters do not exist in an OLEDB source or lookup build.
- Updated emission properties to correctly handle SSIS 2008/2005 packages.
- Added XML escaping to SSIS emitter to fix 2012+ emission errors.

- Updates to SSISEmitter for 2016 support.

## Bug Fixes

- Fix for case insensitivity in VB within metadata dynamic object templates.
- Fixed issue where code directives inside includes were not being added to the parent file.
- Fixed issue where language selection in global includes would not apply to dynamic object code files.
- Fixed issue where any non-Biml files in the logical view would be removed by not deleted from the harddrive.
- Fixed issue where Bundle settings editor did not have a scrollbar.
- Fixed issue where failed builds would be reported as successful when running in unicode support mode.
- Fixed issue where dataflow and control flow edges would not render on machines with some locale settings.
- Fixed partition source selector in partition editor.
- Fixed issue where items duplicated from a BimlScript object were not fully populated and could not be modified.
- Fixed issue where custom dataflow properties for custom components could have invalid types - particularly for System.Null.
- Fixed issue where error messages for code files would not show the correct filename and line number. Performance
- Improved performance of CallBimlScript by around 30X by adding caching when possible.
- Added caching of internal build information for ScriptTask Projects and Script Component Projects. This will massively increase build performance where the same script task or component is reused in many locations in a project (provided that the ProjectCoreName is also reused).
- Improved performance issue in the editor when there were large blocks of plain text.