

## Business Intelligence Markup Language (Biml)

Biml was developed by Varigence and launched in 2008 to simplify the development and maintenance of Business Intelligence (BI) and Data Warehouse (DWH) solutions. The underlying idea was that why delivering these systems, that at first glance seem fairly straightforward, takes an incredible amount of time to develop and maintain. Surely there are ways to make this process easier?

Biml, which stands for Business Intelligence Markup Language, was defined as a solution to counter this problem. Biml is an XML-like declarative language, which describes the metadata and is used to generate BI and DWH components such as SSIS packages and SSAS cubes. The most prominent use-case is the generation of Extract, Transform and Load (ETL) processes based on metadata as this easily takes up 80% of the effort in typical traditionally developed DWH solutions. Unlike competing code-generation software for ETL generation purposes, Biml is essentially an open scripting environment where every property of the BI or DWH output can be configured. This means there is no 'shoehorning' into specific patterns: you can define your own approaches quickly and efficiently.

One of the key features of Biml is that the XML tags that describe objects or properties can be wrapped in C# or Visual Basic code to create dynamic output. When compiling the Biml script, the output can be directed using arrays or database connections which allow you to work against a metadata repository. You can either reuse an existing repository found on the internet or part of available software, or design your own. Combining coding and Biml also opens up options to include external libraries within the .Net Framework, which means approaches can be highly customised or, conversely, can be used to compile recurring logic into a Dynamic Link Library (DLL).

After the initial launch, Biml (the Biml compiler) was made available as part of a free / open-source version BidsHelper and the MIST development environment. BidsHelper contains many other functions that arguably were missing from the Visual Studio environment to develop SSIS with (BI Development Studio, or BIDS), such as ordering, differentials and reinitialisation of GUIDs. And, of course, it also includes the Biml compiler. To date, BidsHelper is a very popular plug-in for the Visual Studio BIDS environment, these days referred to as Visual Studio Data Tools (for BI).

Meeting the market need of not only creating a flexible development environment (Biml) but also providing guidance on implementation and design patterns, BimlFlex and BimlOnline were launched at a later stage. These solutions provide a starting point with pre-defined and configurable ETL generation patterns based on Biml. If you are looking for a head-start with predefined templates this could be an option for you. As a final note, there is a now a separate plug-in for Visual Studio Data Tools available: BimlExpress. This is now the de-facto standard plug-in used for Biml development (if you are not using BimlStudio, BimlFlex or BimlOnline) and essentially supersedes the Biml compiler provided in BidsHelper.

## How to get started with Biml?

To get working with Biml you need an installed instance of SQL Server (2008-2016) including SQL Server Integration Services and the SQL Server Management Studio. Also, Visual Studio Shell or Visual Studio with the Data Tools (SSDT – BI) is required. The shell comes installed with SSIS.

Once the environment is available, you can install either BimlExpress or BidsHelper although BimlExpress is recommended.

- BimlExpress can be downloaded from <https://varigence.com/BimlExpress>
- The BidsHelper plug-in can be downloaded at <https://bideveloperextensions.github.io/>

There is a varied palette of Biml scripts and explanation found on the internet, including many get-started guides and support forums. Get started on your first Biml today!