**WHAT IS SSIS?**

SQL Server Integration Services (SSIS) is the anchor in a growing suite of products that make up the Microsoft SQL Server Business Intelligence (BI) platform. What makes SSIS so important is without the data movement and cleansing features that SSIS brings to the table, the other SQL Server BI products can’t operate. In its simplest form, SSIS is an enterprise-level, in-memory ETL tool. However, SSIS is not just a fancy wrapper around an import wizard. In a drag-and-drop development environment, ETL developers can snap together intricate workflows and out-of-the-box data-cleansing flows that rival custom coding and expensive million-dollar, third-party tools. The best thing about SSIS is that you have already paid for it when you license SQL Server.

**WHAT’S NEW IN SSIS**

The scope of the SQL Server 2014 release of SSIS resembles the scope of the SQL Server 2008 R2 release. With the last release of SQL Server 2008 R2, the Microsoft SSIS team did very incremental changes after a very large SQL Server 2008 release. In SQL Server 2012 release, Microsoft had focused on SSIS manageability, making it easier to deploy and execute. Also added in 2012 are robust new data cleansing components that help you standardize and detect data anomalies. Furthermore, improvements to the development tools will help make SSIS developers more productive and help new developers get up to speed more easily. The SQL Server 2014 release uses a newer version of Visual Studio but all in all, it will feel much like SQL Server 2012.

**The SQL Server Data Tools Experience**

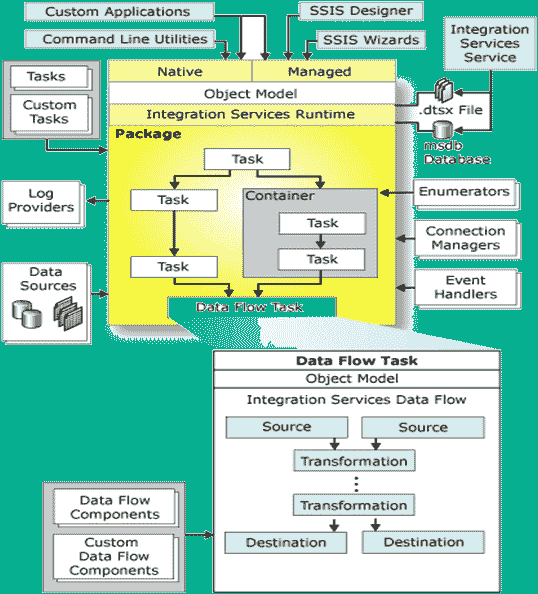
The SQL Server Data Tools (SSDT) was previously called Business Intelligence Development Studio (BIDS) in SQL Server 2008, and it is the central environment in which you’ll spend most of your time as an SSIS developer. SSDT is just a specialized use of the familiar Visual Studio development environment. In SQL Server 2014, SSDT no longer installs when you install SQL Server. Instead, you’ll have to download and install the SQL Server Data Tools (Business Intelligence for Visual Studio) from the Microsoft website. Visual Studio can host many different project types, from Console applications to Class Libraries and Windows applications. Although you may see many project types when you create a project, SSDT actually contains project templates for only Analysis Services, Integration Services, Report Server, and variants thereof. SSIS in particular uses a BI project type called an Integration Services project (see below Figure), which provides a development design surface with a completely ETL-based set of tools in the Toolbox window.

## Why we use SSIS?

Here, are key reasons for using SSIS tool:

* SSIS tool helps you to merge data from various data stores
* Automates Administrative Functions and Data Loading
* Populates Data Marts & Data Warehouses
* Helps you to clean and standardize data
* Building BI into a Data Transformation Process
* Automating Administrative Functions and Data Loading
* SIS contains a GUI that helps users to transform data easily rather than writing large programs
* It can load millions of rows from one data source to another in very few minutes
* Identifying, capturing, and processing data changes
* Coordinating data maintenance, processing, or analysis
* SSIS eliminates the need of hardcore programmers
* SSIS offers robust error and event handling

**SSIS Architecture**

SSIS Architecture

Following are components of SSIS architecture:

* Control Flow (Stores containers and Tasks)
* Data Flow (Source, Destination, Transformations)
* Event Handler (sending of messages, Emails)
* Package Explorer (Offers a single view for all in package)
* Parameters (User Interaction)

Let's understand each component in detail:

**Control Flow**

Control flow is a brain of SSIS package. It helps you to arranges the order of execution for all its components. The components contain containers and tasks which are managed by precedence constraints.

**Precedence Constraints**

Precedence constrain are package component which direct tasks to execute in a predefined order. It also defines the workflow of the entire SSIS package. It controls the execution of the two linked tasks by executing the destination tasks based on the result of the earlier task — business rules which are defined using special expressions.

**Task**

A 'Task' is an individual unit of work. It is the same as a method/function used in a programming language. However, in SSIS, you don't use coding methods. Instead, you will use drag & drop technique to design surface and to configure them.

**Containers**

The container is units for grouping tasks together into units of work. Apart from offering visual consistency, it also allows you to declare variables and event handlers which should be in the scope of that specific container.

Four types of containers in SSIS are:

* A Sequence Container
* A For Loop Container
* Foreach Loop Container

**Sequence Container**: allows you to organize subsidiary tasks by grouping them, and allows you to you apply transactions or assign logging to the container.

**For loop container:** Provides the same functionality as the sequence Container except that it also lets you run the tasks multiple times. However, it is based on an evaluation condition, like a looping from 1 to 100.

**For each Loop Container:** It also allows looping. But the difference that instead of using a condition expression, loop s done over a set of objects, likes files in a folder.

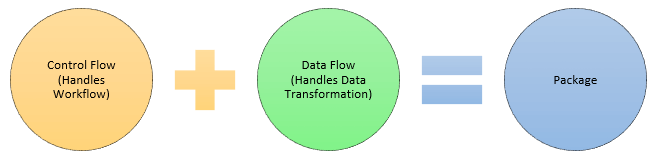
**Data Flow**

The main use of the SSIS tool is to extract data into the server's memory, transform it, and write it to another destination. If Control Flow is the brain, Data Flow is the heart of SSIS

**Packages**

Another core component of SSIS is the notion of a package. It is a collection of tasks which execute in an orderly fashion. Here, president constraints help manage the order in which the task will execute.

A package can help you to saves files onto a SQL Server, in the msdb or package catalog database. It can save as a .dtsx file, which is a structured file very similar to .rdl files are to Reporting Services.



**7.Parameters**

Parameters behave much like a variable but with a few main exceptions. It can be set outside the package easily. It can be designated as values that must be passed in for the package to start.

**SSIS Tasks Types**

In SSIS tool, you can add a task to control flow. There are different types of tasks which perform various kinds of works.

Some important SSIS tasks are mentioned below:

|  |  |
| --- | --- |
| **Task Name** | **Descriptions** |
| Execute SQL Task | As its name suggests, it will execute a SQL statement against a relational database. |
| Data Flow Task | This task can read data from one or more sources. Transform the data when it is in the memory and write it out against one or more destinations. |
| Analysis Services Processing Task | Use this task to process objects of a Tabular model or as an SSAS cube. |
| Execute Package Task | Use can use this SSIS task to execute other packages from within the same project. |
| Execute Process Task | With the help of this task, you can specify command line parameters. |
| File System Task | It performs manipulations in the file system. Like moving, renaming, deleting files, and creating directories. |
| FTP Tasks | It allows you to perform basic FTP functionalities. |
| Script Task | This is a blank task. You can write NET code which performs any task; you want to perform. |
| Send Mail Task | You can send an email to notifying users that your package has is finished, or some error occurs. |
| Bulk Insert Task | Use can loads data into a table by using the bulk insert command. |
| Script Task | Runs a set of VB.NET or C# coding inside a Visual Studio environment. |
| Web Service Task | It executes a method on a web service. |
| WMI Event Watcher Task | This task allows the SSIS package to wait for and respond to certain WMI events. |
| XML Task | This task helps you to merge, split, or reformat any XML file. |

**Lesson tasks**

This lesson contains the following tasks:

* Step 1: Create a new Integration Services project
* Step 2: Add and configure a Flat File connection manager
* Step 3: Add and configure an OLE DB connection manager
* Step 4: Add a Data Flow task to the package
* Step 5: Add and configure the flat file source
* Step 6: Add and configure the lookup transformations
* Step 7: Add and configure the OLE DB destination
* Step 8: Annotate and format the Lesson 1 package
* Step 9: Test the Lesson 1 package