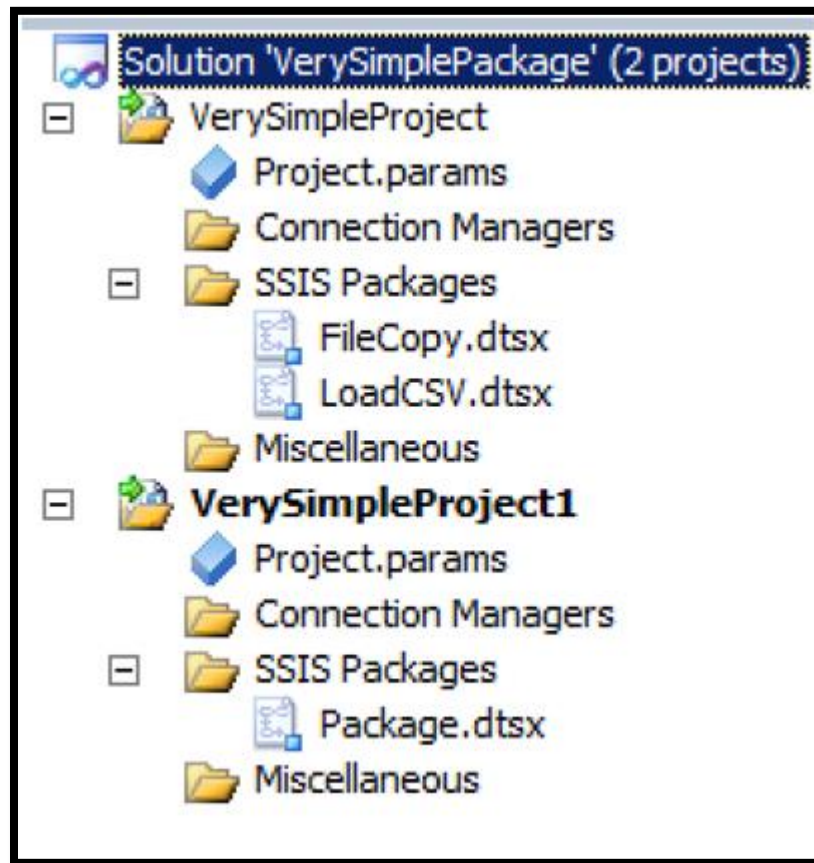


## SSIS-Deployment in SSIS.docx

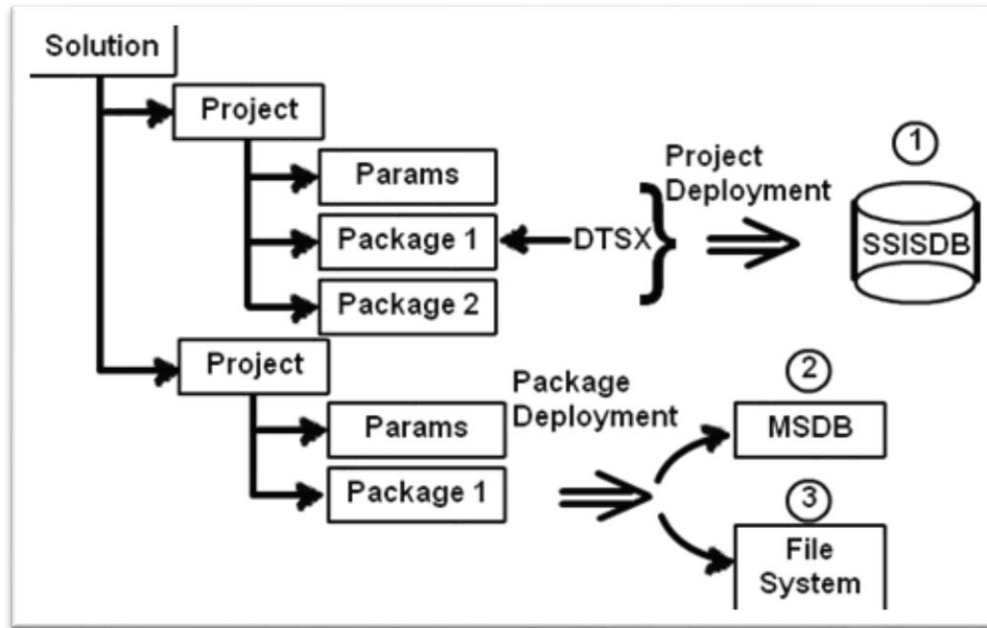
Before going in to deployment let's understand what elements gets deployed in a SSIS project. If you look at your SSIS project structure it looks something as shown below.



At the top we have the solution and inside the solution we have project files. Further every project has package file with extension DTSX and configuration for those packages are stored in project.params.

To understand SSIS deployment we need to think in terms of "What" and "Where". So what are the various ways of deploying and where can we deploy them.

## SSIS-Deployment in SSIS.docx



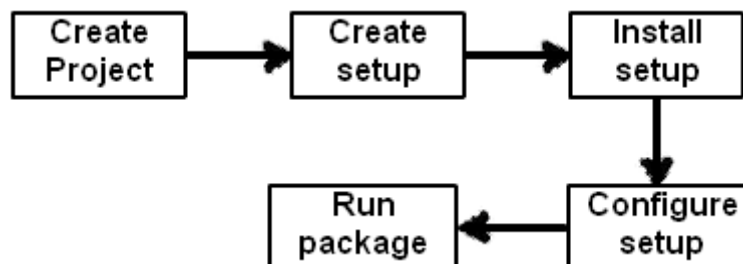
So let's answer first what? There are two ways of deploying SSIS projects, 1st Project level deployment and 2nd package level deployment.

When you do project level deployment you deploy all packages in one go. While in package level deployment, deployment happens at the DTSX file level. So you need to deploy individual DTSX files.

Project level deployment is the new way of doing deployment and was introduced in SQL 2012 while package level is the old way till SQL 2008. In this lab we will focus more on project level deployment rather package level.

The second thing we need to understand is "Where can we deploy these packages?" In other words what are the different ways of hosting these packages? So there are 3 primary source or hosting in which you can deploy:-

- In SQL Server service itself i.e. in SSISDB database.
- In SSIS service :-
  - In File system
  - MSDB



## SSIS-Deployment in SSIS.docx

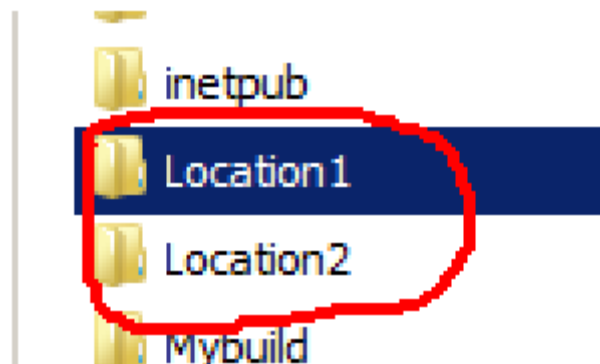
So in order to understand deployment step by step we will do this in 5 steps:-

- Create a simple file copy project which copies file from a source to a destination.
- Create setup of that project.
- Install the package setup.
- Configure the package before running.
- Finally running the package.

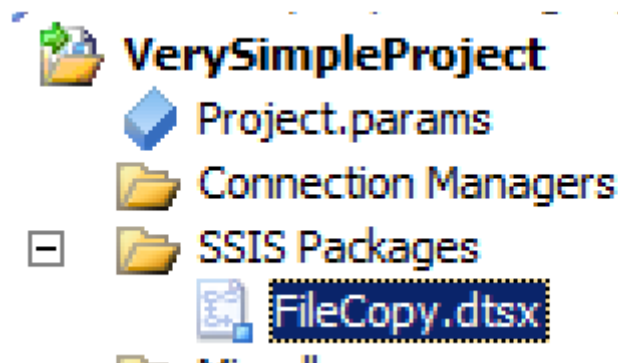
So let's start the journey.

### Step 1:- Create the file copy project

So let's create two folders "Location1" and "Location2" as shown in the below figure and let's create a simple text file "SimpleText" file in "Location1".

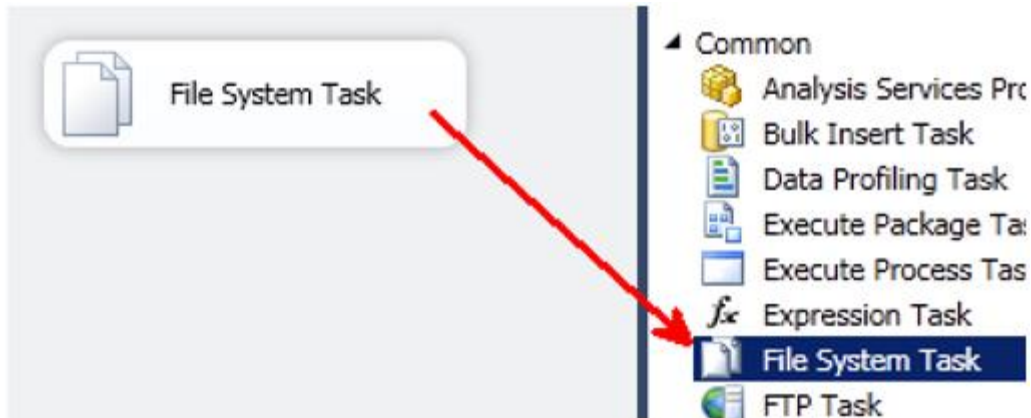


Now we would like to copy this "SimpleText" file from "Location1" folder to "Location2" folder. So in order to achieve the same we will be using the "File system task" control. So create a simple SSIS project and add a package "FileCopy.dtsx" file. Your project structure should look something as shown in the below figure.



On this package drag and drop the file system task from the control flow tool box and put it on the control flow designer pane as shown in the below figure.

## SSIS-Deployment in SSIS.docx



Also we need to provide source and destination files which needs to be copied. So right click on the "File system task" control and click edit and provide the "Destination" and "Source" connection. So in the source connection we will be pointing to location1 and in the destination we will be pointing to location2 folder.

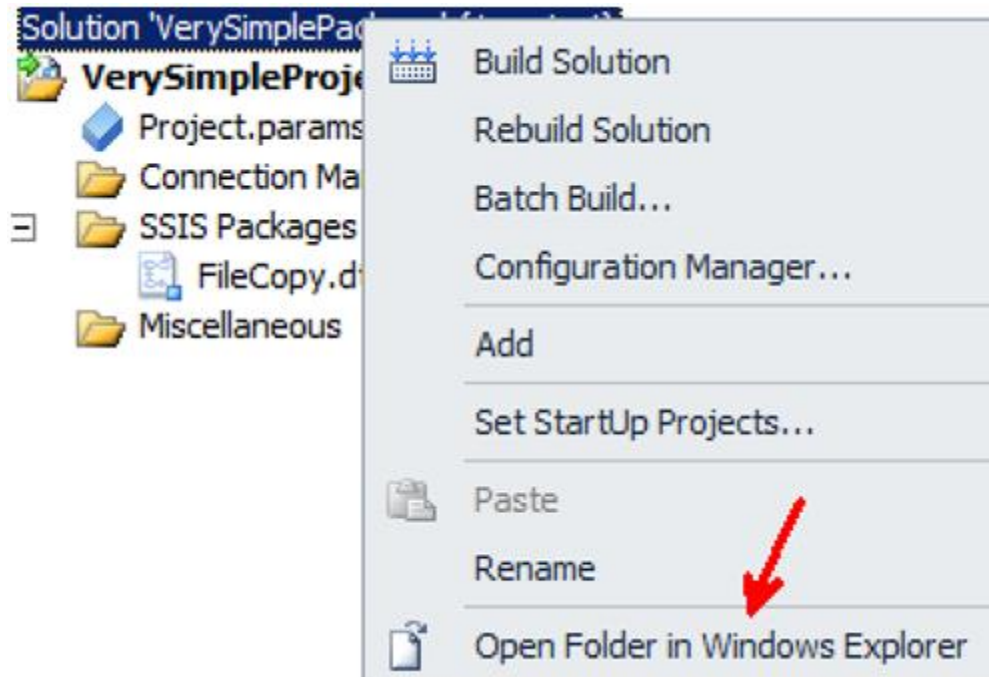
<b>Destination Connection</b>	
IsDestinationPathVariable	False
DestinationConnection	Dest
OverwriteDestination	<New connection...>
<b>General</b>	
Name	Sometext.txt
Description	File System Task
<b>Operation</b>	
Operation	Copy file
<b>Source Connection</b>	
IsSourcePathVariable	False
SourceConnection	Sometext.txt

Run this project and once test, if the file is getting copied from "Location1" folder to "Location2" folder.

We have purposely kept this project simple so that we can concentrate on the deployment more rather than the SSIS project.

### Step 2:- Create a setup

As said in the previous section in SQL Server 2012 the deployment model is now project deployment. So do a full build by clicking on build à rebuild all. Now go to the project folder by right clicking on the solution and click on "open folders in windows explorer".

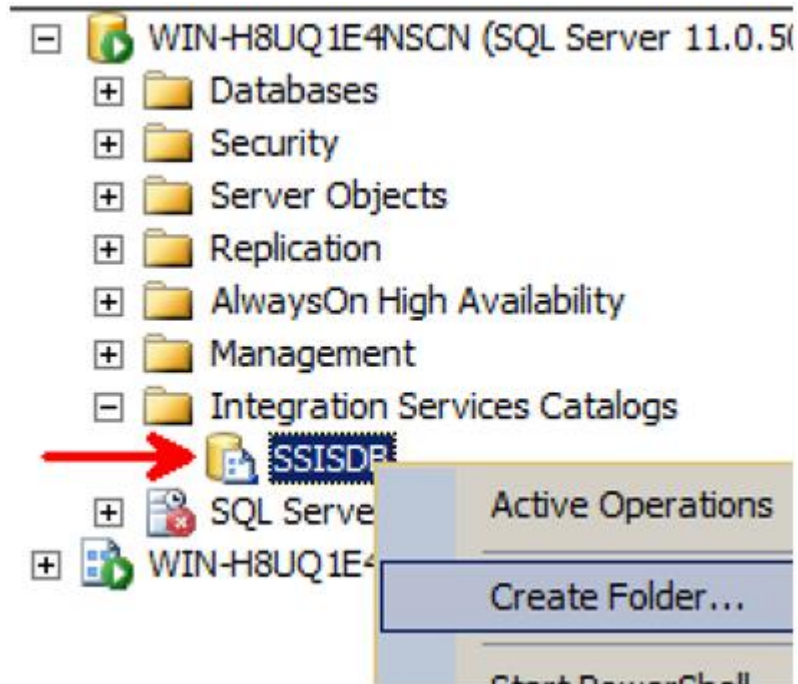


Go in to bin/Development folder you will find a full SSIS setup created with a file extension "ISPAC". Double click and run the same.

### Step 3:- Install the setup

For now we will be deploying in SQL Server SSISDB. So connect to SQL Server instance and browse to the "Integration Services catalogs" folder and right click on "SSISDB" and click on create folder. So we will create a folder "MyPackages" and we will deploy the packages in the same.

## SSIS-Deployment in SSIS.docx



Once you run the ISPAC file setup it will start a wizard. In the wizard there are two important things to be mentioned one is the source and the other is the destination. Source is from where the things will be loaded and i.e the ISPAC file and destination is where the deployment will happen.

We have already created a folder "Mypackages" so select the same and install in the same.

**Enter the destination server name and where the project will be located in the Integration Services catalog.**

Server name:

WIN-H8UQ1E4NSCN Browse...

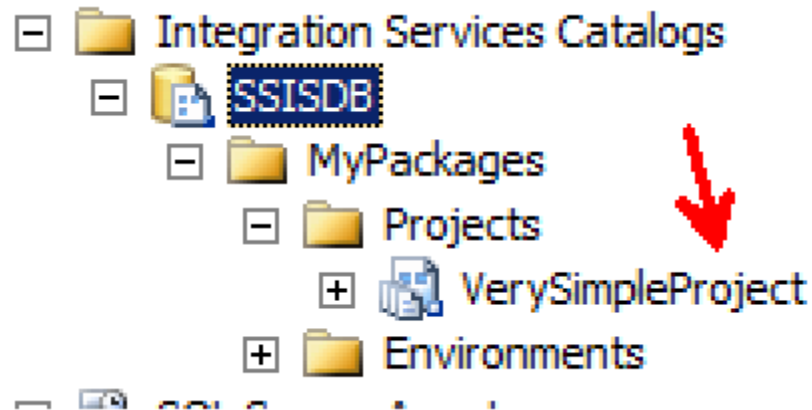
Path:

/SSISDB/MyPackages/VerySimpleProject Browse...

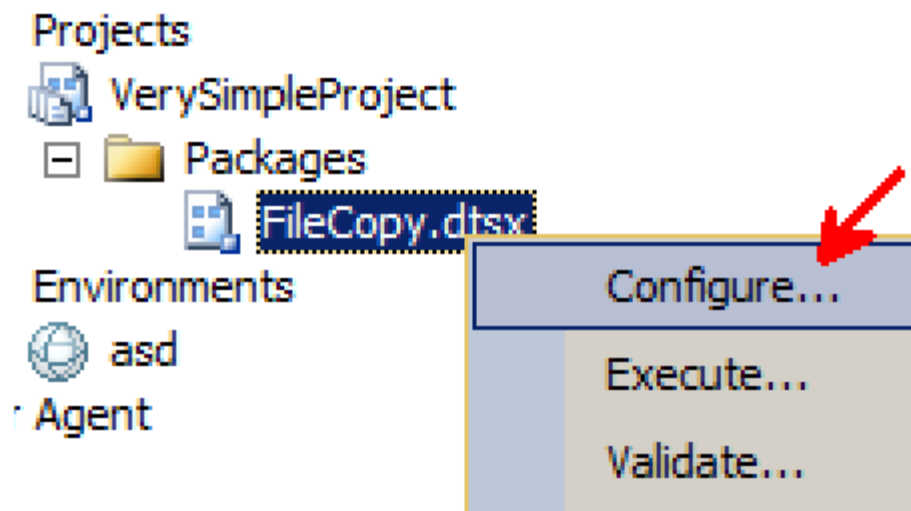
### Step 4:- Configure the package

Once the package is installed you should see your project inside "SSISDB\Mypackages" folder.

## SSIS-Deployment in SSIS.docx

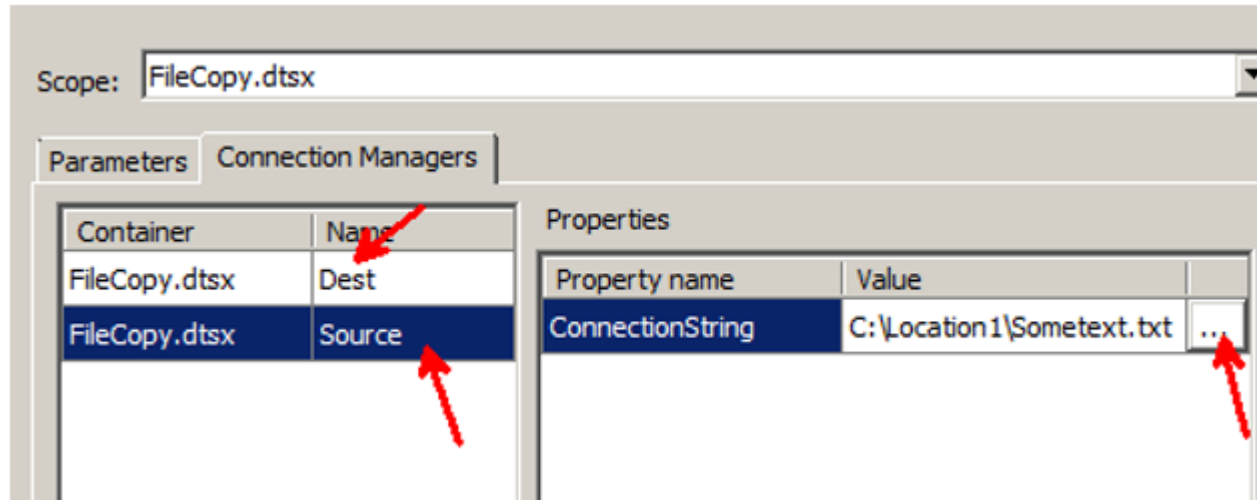


Expand the SSIS project folder and browse to the package and right click on the same. Now you can do three things first validate the package, configure and run it. So let's first configure and then run.



So once you click on configure the following screen comes up with two tabs. The first tab is for parameters we will talk about that later on. For now the second tab is important. The second tab has connection managers, remember we had two file connection one for source and other for destination.

## SSIS-Deployment in SSIS.docx



If you wish to configure the file paths you can click on the “...” and set a different value.

### Step 5:- Run the package

Once we have finished the configuration we can run the package again by right clicking and execute. After execution you can also see a report of your success and failure.

Operation ID	370034
Package	MyPackages\VerySimpleProject\FileCopy.dtsx
Environment	-
Status	Succeeded

#### Execution Overview

Filter: Result: All; (3 more)

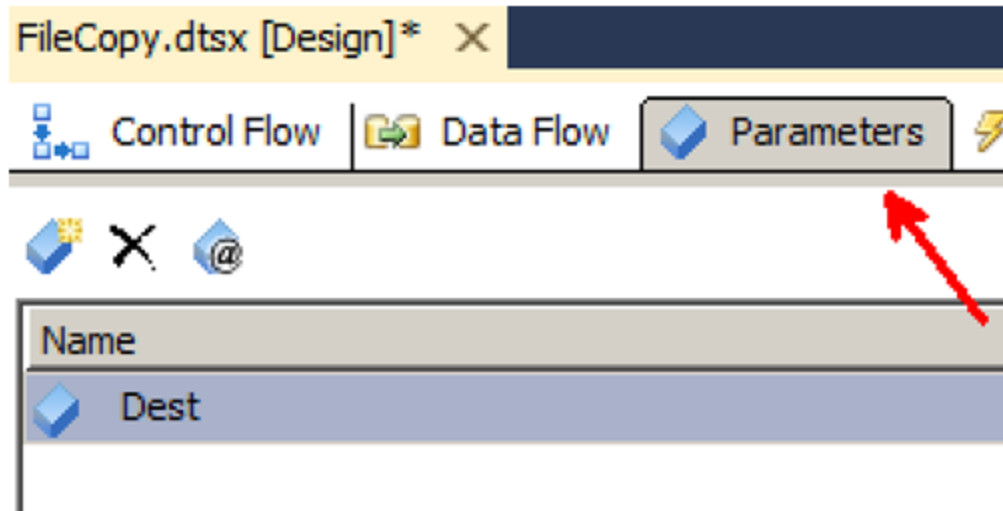
Result	Duration (sec)	Package Name	Task Name	Execution Path
Succeeded	0.624	FileCopy.dtsx	FileCopy	\FileCopy
Succeeded	0.25	FileCopy.dtsx	File System Task	\FileCopy\File System Task

Some points to remember before we conclude deployment:-

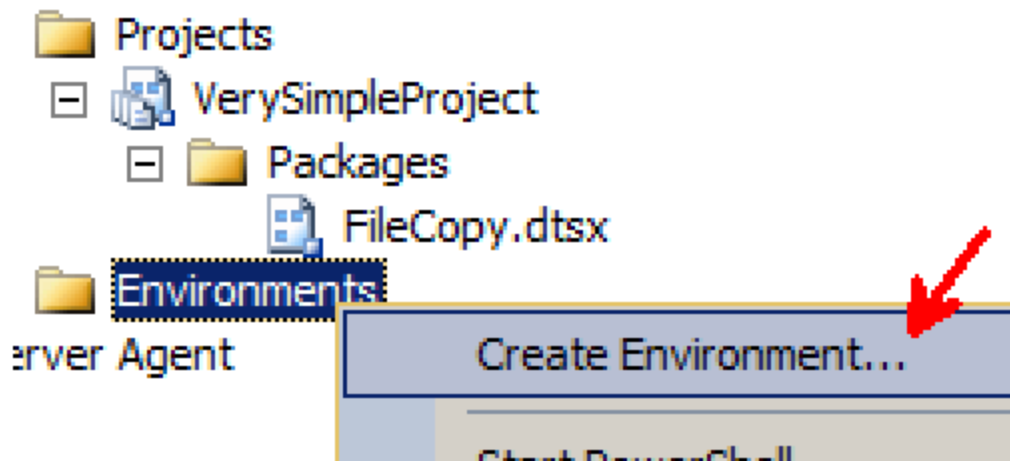
You can parameterize your package by creating parameter's using the parameter tab and these parameters can be connected with your connection manager properties of variables. These values can be supplied when you execute the package.



## SSIS-Deployment in SSIS.docx

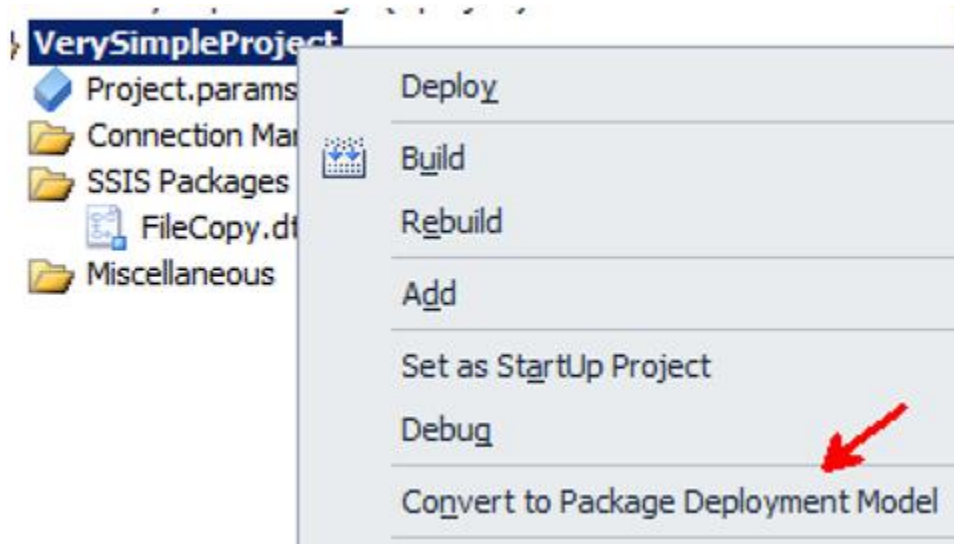


If you want to share configuration data at the project level. Like some values which are common for all packages like server name or some common folder etc you can create environment variables. You can then attach these environmental variables with package configuration or variables.



In this lab we have focused mainly on package deployment but in case you want to use the old way of deployment i.e. package you can always convert it as shown in the below figure.

## SSIS-Deployment in SSIS.docx



In case in project deployment you want to individually deploy packages you can do the same by right clicking on project folder and clicking import packages as shown below.

