**CALLING FROM STORE PROCEDURE**

USE master

GO

EXEC sp\_configure 'show advanced options', 1

GO

RECONFIGURE WITH OVERRIDE

GO

EXEC sp\_configure 'xp\_cmdshell', 1

GO

RECONFIGURE WITH OVERRIDE

GO

EXEC sp\_configure 'show advanced options', 0

GO

declare @ssisstr varchar(8000), @packagename varchar(200),@servername varchar(100), @params varchar(8000)

----my package name

set @packagename = 'ImportItemFile'

----my server name

set @servername = 'myserver\sql2k5'

set @params = '/set \package.variables[FileName].Value;"\"\\127.0.0.1\SSIS\NewItem.xls\""

/set \package.variables[ContractDbConnectionString].Value;"\"Data Source=myserver\SQL2K5;

User ID=sa;Password=sapass;Initial Catalog=Items;Provider=SQLNCLI.1;

Persist Security Info=True;Auto Translate=False;\""

/set \package.variables[SupplierID].Value;"\"22334\""'

----now making "dtexec" SQL from dynamic values

set @ssisstr = 'dtexec /sq ' + @packagename + ' /ser ' + @servername + ' ' + @params

-----print line for verification

--print @ssisstr

----now execute dynamic SQL by using EXEC.

DECLARE @returncode int

EXEC @returncode = xp\_cmdshell @ssisstr

select @returncode

**USING C#**

using Microsoft.SqlServer.Dts.Runtime;

private void Execute\_Package()

{

string pkgLocation = @"c:\test.dtsx";

Package pkg;

Application app;

DTSExecResult pkgResults;

Variables vars;

app = new Application();

pkg = app.LoadPackage(pkgLocation, null);

vars = pkg.Variables;

vars["A\_Variable"].Value = "Some value";

pkgResults = pkg.**Execute**(null, vars);

if (pkgResults == DTSExecResult.Success)

Console.WriteLine("Package ran successfully");

else

Console.WriteLine("Package failed");

}

// Load package from SQL Server

Package package2 = app.LoadFromSqlServer("ExamplePackage","server\_name", "sa", "your\_password", null);

package2.ImportConfigurationFile("c:\\ExamplePackage.dtsConfig");

Variables vars2 = package2.Variables;

vars2["MyVariable"].Value = "value from c# again";

DTSExecResult result2 = package2.**Execute**();

Console.WriteLine("Package Execution results: {0}", result2.ToString());

Place the SSIS Package in Server 1.

Create a SQL Job to run the Package.

Configure further steps.

Call the Job from Server 2 using DB Command ( SP will do ) like this,

EXEC msdb..sp\_start\_job @job\_name = 'JOB\_NAME'

Point to be noted is this process is asynchronous