Sometimes we need to import thousands of images to SQL Server. This example shows how to import a list of images to SQL Server using SQL Server Integration Services.

**Solution**

**Requirements**

* SQL Server Enterprise or Standard (in this case I am using SQL Server 2008 R2, but it can work with SQL Server 2005 as well).
* SSIS installed (it is included in the SQL Server installer).

**Demonstration**

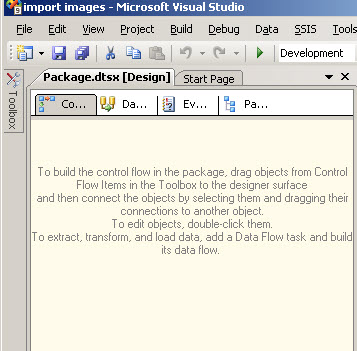
1. First of all, it is necessary to create a flat file named **listImages.txt** (or the name of your preference) with the paths of the images that you want to import to SQL Server such as the following:
2. C:\images\pic1.jpg
3. C:\images\pic2.jpg

C:\images\pic3.jpg

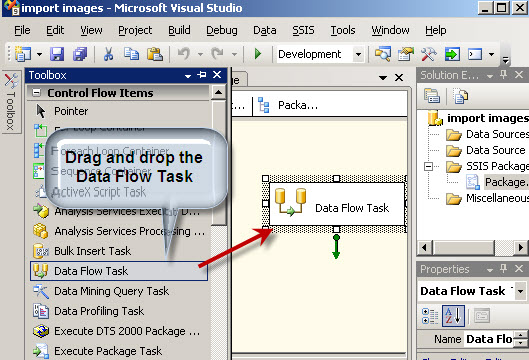
1. Create a table **myImages** with 3 columns: The ID is the primary key, image will store the picture and the path will store the image path:
2. CREATE TABLE [dbo].[myImages](
3. [id] [smallint] IDENTITY(1,1) NOT NULL,
4. [path] [varchar](200) NULL,
5. [image] [image] NULL,
6. CONSTRAINT [PK\_myImages] PRIMARY KEY CLUSTERED
7. (
8. [id] ASC
9. )WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF,
10. IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON)
11. ON [PRIMARY]
12. ) ON [PRIMARY] TEXTIMAGE\_ON [PRIMARY]

GO

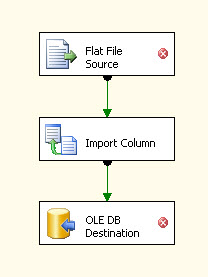
1. Start the SQL Server Business Intelligence Studio and create a New SQL Server Integration Project.



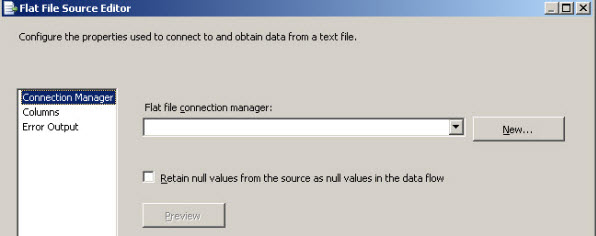
1. Drag and drop the **Data Flow Task** to the design pane.



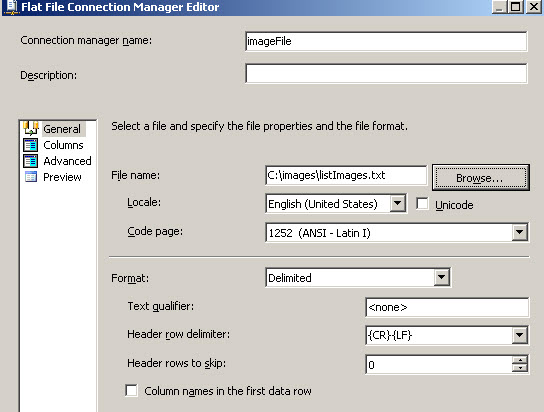
1. In the design pane double click the **Data Flow Task**
2. In the **Data Flow** tab drag and drop a **Flat File Source**, an I**mport Column** and an **OLE DB Destinatio**n. Join the tasks with the green arrows as shown below.



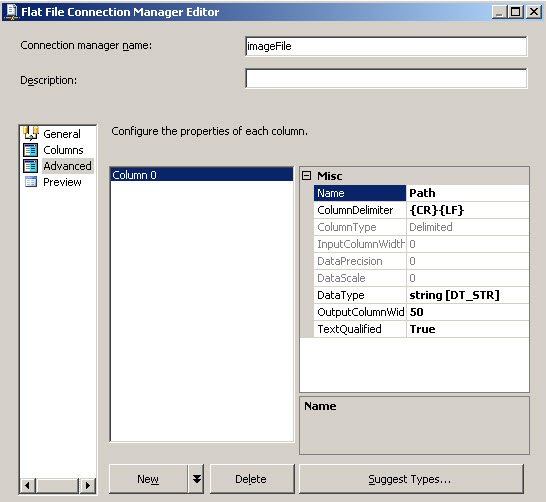
1. The Flat File Source will connect to the **listImages.txt** created in step 1. Double click on the Flat File Source to edit the settings.
2. In the Flat file connection manager, press **New...**



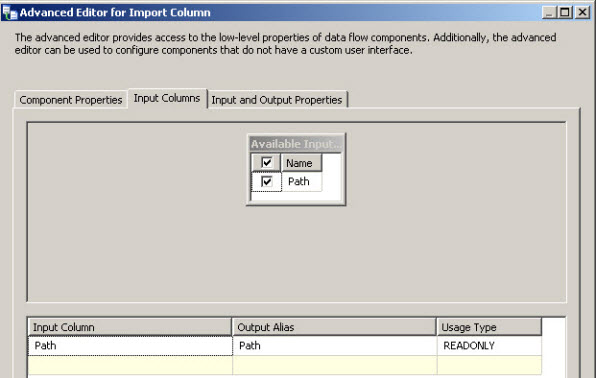
1. In the Flat File Connection Manager Editor write a Connection manager name (any name can be used. In this example we will use **imagefile**).
2. In the File name press **Browse...** and select the **listImages.tx**t file created in step 1 (it can be stored anywhere. In this example it is in **c:\images\**)



1. Select the **Advanced** options and in the **Name**, type **Path** to change the column name as shown below.

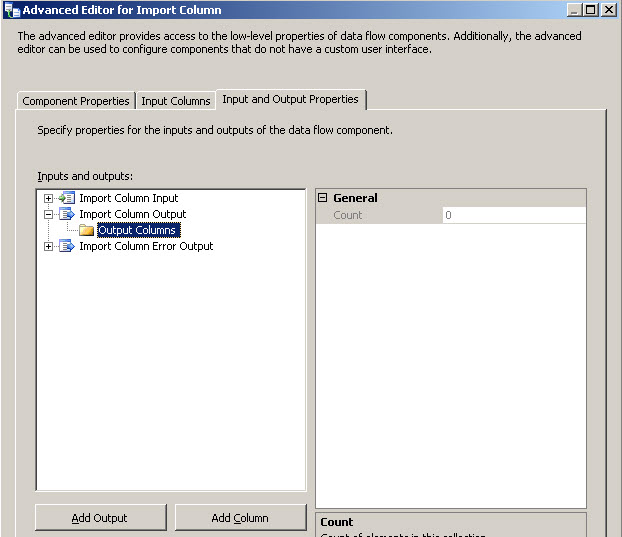


1. Then click **OK** on each window to save these settings.
2. Next double click the **Import Column** transform and click the **Input Columns** tab as shown below.
3. In the **Input Columns** tab check **Name**

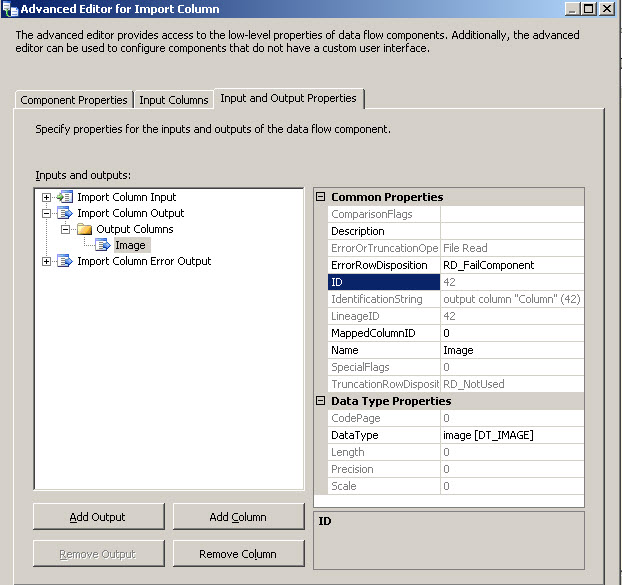


1. Click the **Input and Output Properties**
2. Open the **Import Column Output** tree and select the **Output Columns** as shown below.
3. Click the **Add Column...**

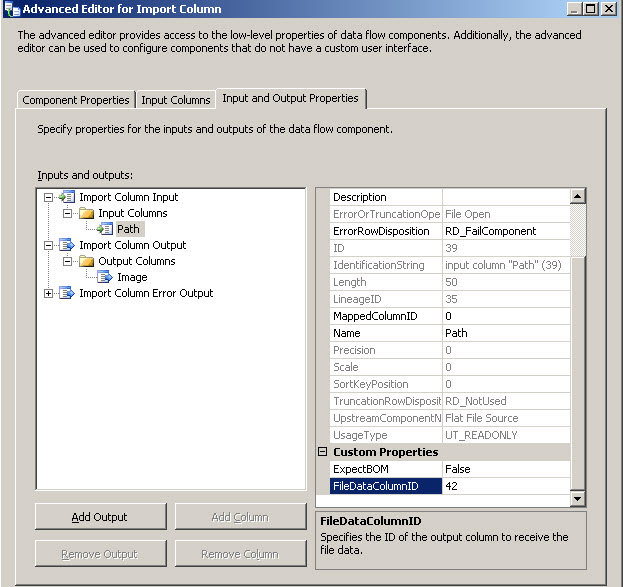
button and name the new column **Image**



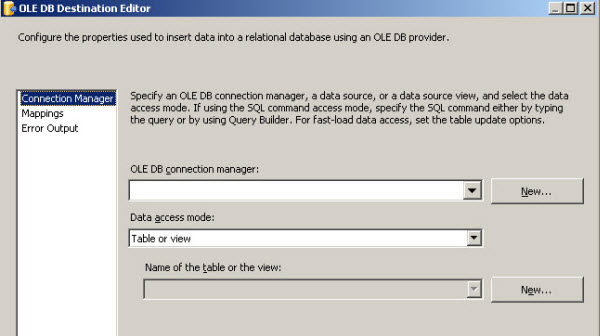
1. Get the ID property value of the column created (in this example the ID is 42).



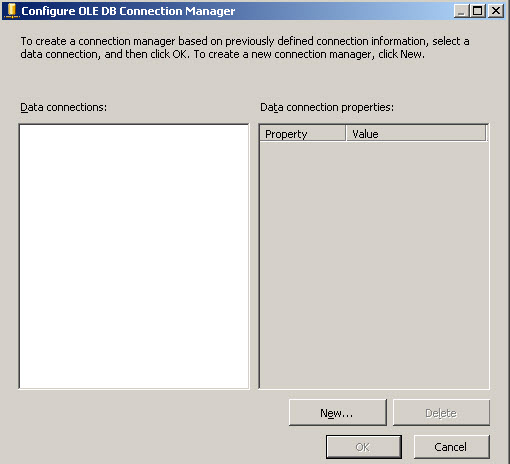
1. In the I**nput and Output Properties** tab, open the **Import Column Input** > **Input Columns** and select **Path** as shown below.
2. In the **FileDataColumnID** property write the ID from step 18 (in this example 42) and press OK to save these settings.



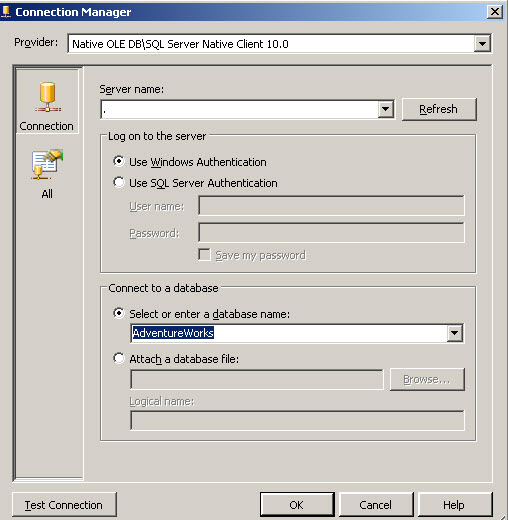
1. Double click the **OLE DB Destination.**
2. In the OLE DB Destination Editor Window, press **New...** for the OLE DB connection manager.



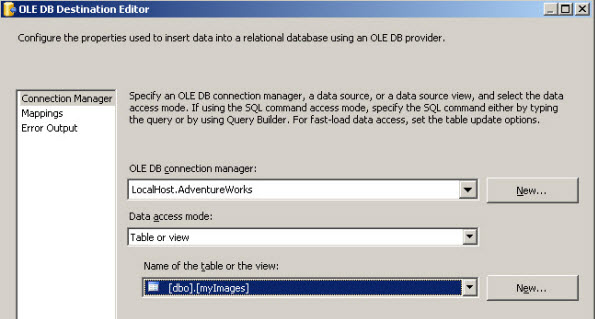
1. In the Configure OLE DB Connection Manager, press the **New...** button.



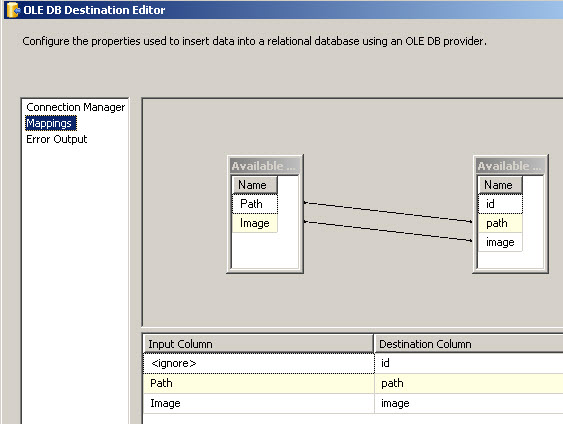
1. In the Connection Manager, in the **Provider** combo box, select Native OLE DB\SQL Server Native Client.
2. In the Server name write the SQL Server Name (in this example, the local server name is used which can be specified with a period).
3. Select the **Log on to the server** information (in this example Windows Authentication is used).
4. In the select or enter the database name, select the database used to create the table in step 2 (in this example, the Adventureworks database is used) and press OK.



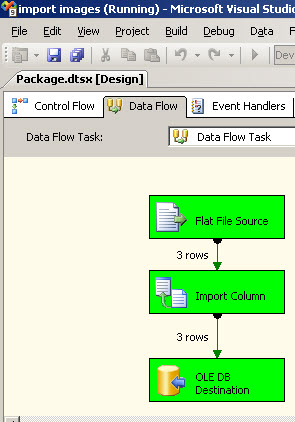
1. In the OLE DB Destination Editor **name of the table or the view**, select the table created in step 2 (in this example the table name is myImages).



1. In the OLE DB Destination Editor, press the **Mappings** page and then press OK.



1. Now we are ready to start the project. Press the start debugging icon (green arrow). If everything is OK, the tasks should be colored green and the number of rows imported should be displayed.



1. To verify that the data was imported successfully, open SQL Server Management Studio.
2. Go to the database (in this example, Adventureworks) and open the myImages table.

