



SQL Server Integration Services Using Visual Studio 2005: A Beginner's Guide

by Jayaram Krishnaswamy
Packt Publishing. (c) 2007. Copying Prohibited.

Reprinted for SATHYANARRAYANAN SHANMUGAM, Cognizant Technology Solutions

SathyaNarrayanan.Shanmugam@cognizant.com

Reprinted with permission as a subscription benefit of **Skillport**,
<http://skillport.books24x7.com/>

All rights reserved. Reproduction and/or distribution in whole or in part in electronic, paper or other forms without written permission is prohibited.



Chapter 9: Using an Aggregate Data Transformation

This chapter shows you how to create a package that can aggregate data for a given group of items in a Data Flow using the Aggregate Data Flow Transformation. You will also learn how to use the Percentage Sampling Data Flow Component. The data will be extracted from SQL Server 2005 and loaded to an in-memory Recordset Destination.

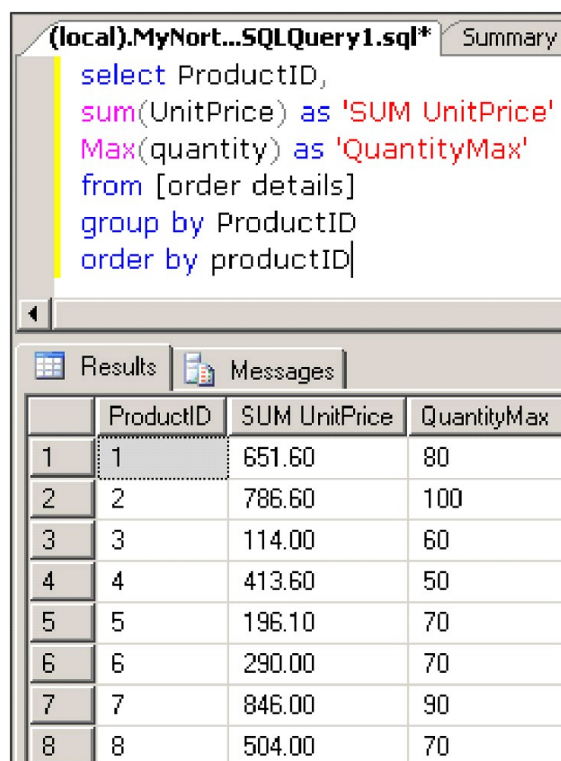
Structured Query Language's SELECT query is perhaps one of the most important statements that helps in identifying a single piece of information from a large database. In developing enterprise reports, there are often times you need to know consolidated specific pieces of information. For example, as a hiring manager you may be interested in knowing the average and minimum salaries you have to pay in a given geographical region for hiring. SQL provides the aggregate functions to address those kinds of questions to the database by working on multiple records of data. Using aggregate functions, you can find sum, maximum, minimum, average values of items in a table or a group.

The aggregate functions Count, Sum, Average, Min, and Max are all defined according ANSI SQL-92 standard. This standard also describes how nulls are handled while aggregating. The Aggregate Data Flow Transformation does the aggregation of data in the context of SSIS.

Hands-On Exercise: Using Aggregate Data Flow Transformation

In order to follow the steps as indicated, you will need a data flow task that connects to a data source and a Recordset Destination to which the data can flow. You will introduce an aggregate data flow component and a percentage sampling data flow item in the path of the data.

In this exercise, you will be aggregating data from the **OrderDetails** table discussed in Chapter 6. The next screenshot shows how an aggregate query is posed and the result of running this query in SQL Server 2005's Management Studio. The aggregate values (**SUM UnitPrice** and **QuantityMax**) are grouped by OrderID.



The screenshot shows a SQL query window titled '(local).MyNort...SQLQuery1.sql*' with a 'Summary' tab. The query is as follows:

```
select ProductID,
sum(UnitPrice) as 'SUM UnitPrice'
Max(quantity) as 'QuantityMax'
from [order details]
group by ProductID
order by productID
```

Below the query window, the 'Results' tab is active, displaying a table with 4 columns: ProductID, SUM UnitPrice, and QuantityMax. The table contains 8 rows of data.

	ProductID	SUM UnitPrice	QuantityMax
1	1	651.60	80
2	2	786.60	100
3	3	114.00	60
4	4	413.60	50
5	5	196.10	70
6	6	290.00	70
7	7	846.00	90
8	8	504.00	70

The following are the major steps in this exercise:

- Create a BI Project and add a Data Flow Task. Add and configure the DataReader Source to extract data from the Local SQL Server.
- Add an Aggregate Data Transformation

- Establish a path to connect DataReader Source with the Aggregate Data Transformation.
- Configure the Aggregate Data Flow Transformation.
- Add a Percentage Sampling Data Transformation.
- Establish a path from Aggregate Data Transformation to the Percentage Sampling Data Transformation.
- Configure the Percentage Sampling Data Flow Item.
- Add a Recordset Destination Data Flow component.
- Configure the Recordset Destination Data Flow Component.
- Build and execute the package, and review results.

Step 1: Create a BI Project and Add a Data Flow Task. Add and Configure the DataReader Source to Pull Data from the Local SQL Server

1. Create a BI project **Ch 9**, change the name of the default Package to `aggregate.dtsx`, add a DataReader Source and configure it to provide at its output, data selected from the database with the following SQL statement: **Select * from [Order Details] order by ProductID**

(The output of this query without the sort is shown in Chapter 6.)

Step 2: Add an Aggregate Data Transformation

The data output from the DataReader Source will be the input to the aggregate transformation. In the aggregate transformation's editor, the aggregation details will be set.

1. Drag and drop an **Aggregate Data Flow** item from the **Data Flow Transformations** group in the **Toolbox** to the **Data Flow** page of the canvas.

Step 3: Establish a Path to Connect DataReader Source with the Aggregate Data Transformation

1. Right-click the **Data Reader Source** and from the drop-down click on **Add Path**.

This displays the **Data Flow** window with the "From:" showing **Data Source Reader**. The process of establishing a path is same as in the previous chapters.

2. After choosing **Aggregate** in the "To:" drop-down, click on the **OK** button in the **Data Flow** window.

This opens the Input/Output Selection window where the **input, Aggregate Input1** is displayed.

3. Click on the drop-down arrow for choosing the "Output:". From the list, choose **DataReader Source**. Now, click on the **OK** button.

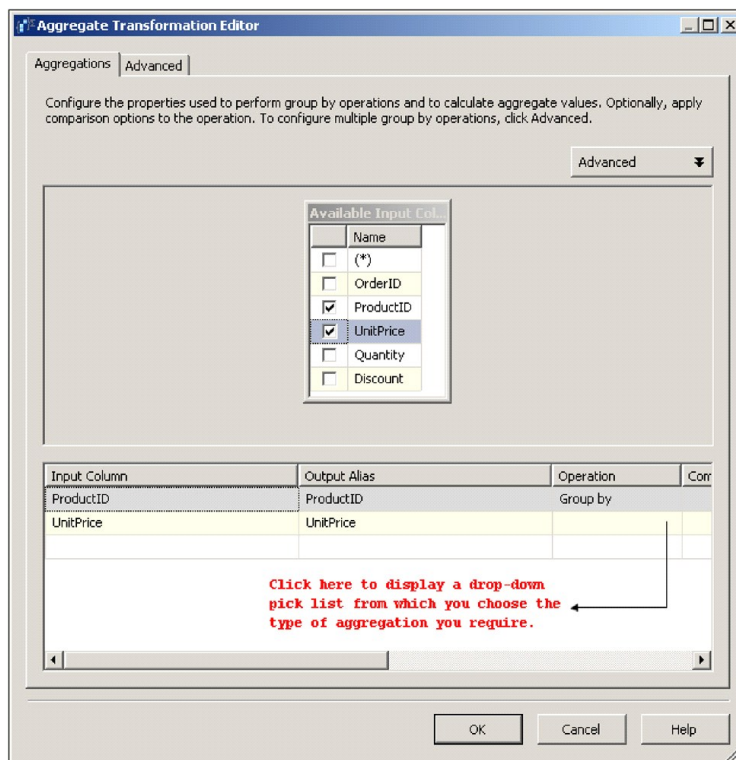
This establishes the path from the **DataReader Source** to the **Aggregate Data Flow** item. You should be able to see a thin green line from the **Data Reader Source** to the **Aggregate Data Flow Transformation** in the canvas. The input to the **Aggregate Data Flow** component is established.

Step 4: Configure the Aggregate Data Flow Transformation

1. Right-click the **Aggregate Data Flow** item and from the drop-down choose **Edit....**

This opens up the **Aggregate Transformation Editor** with two panes, a top pane showing all the columns from the **DataReader Source** output and a bottom pane with a list. In the top pane, all columns are unchecked. We are going to group the ProductID, and in that group we sum the UnitPrice. In order to do this:

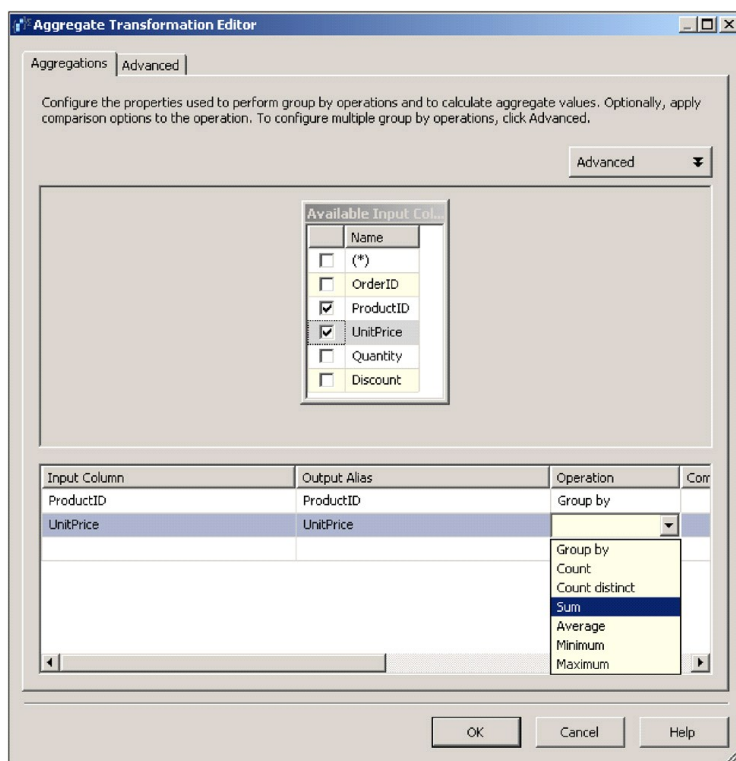
2. Place check marks on the **ProductID** and **UnitPrice** columns in the top pane of the **Aggregate Transformation Editor**, as shown in the next screenshot.



This adds the two line items to the list in the bottom pane, as shown in the previous screenshot. By default the Aggregate Transformation Editor already groups ProductID when the ProductID is checked.

3. Click in the indicated position under the column heading **Operation**.

This opens the pick list as shown. You may choose any of the items in the list. For this exercise, the aggregate function **SUM** will be used.



4. Pick **SUM** from the list and click the **OK** button on this editor.

The **Aggregate Data Flow** item output is now available. The **Aggregate Data Flow** item can be used to configure multiple aggregate functions using the **Advanced** tab of this editor. For this exercise only, the basic editor functionality has been used.

Step 5: Add a Percentage Sampling Data Transformation

The Percentage Data Flow component was described in **Chapter 1**. This component just randomly selects a preset percentage of rows of data to pass from its input to its output path. This component is useful in data mining and wherever you need a smaller representative set of data from a larger set, for example for testing.

1. Drag and drop a **Percentage Sampling Data Flow** item from the **Data Flow Transformations** group in the **Toolbox** to the **Data Flow** page of the canvas.

This adds the **Percentage Sampling Data Flow** item to the **Data Flow** page of the canvas.

Step 6: Establish a Path from Aggregate Data Transformation to the Percentage Sampling Data Transformation

1. Right-click the **Aggregate Data Flow** item and choose **Add Path**.

This opens the **Data Flow** window displaying the "From:" as an **Aggregate data flow** item.

2. Click on the drop-down arrow head in the "To:" window.

In the drop-down, you will see three options, **DataReader Source**, **Aggregate**, and **Percentage Sampling**.

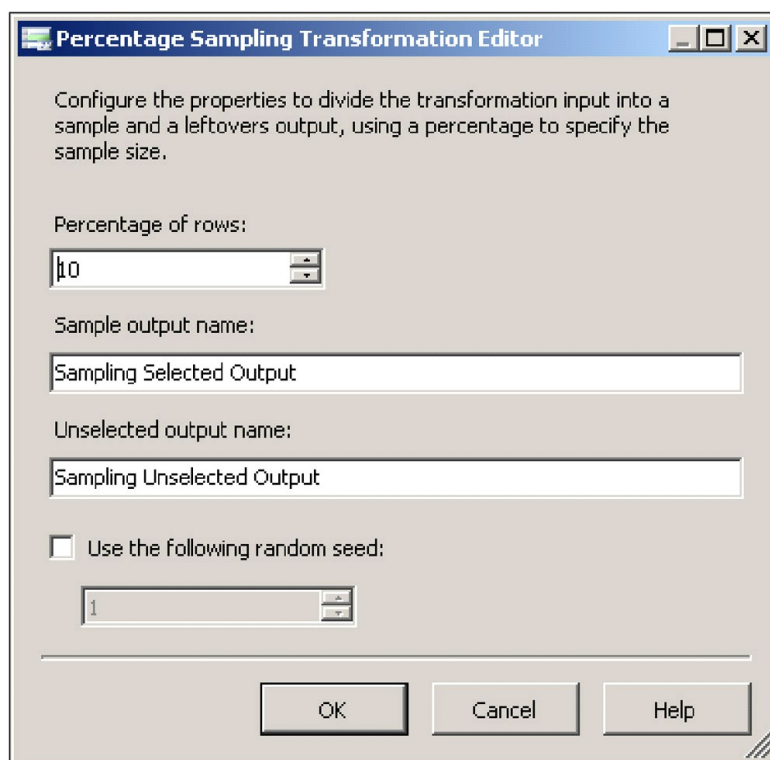
3. Choose **Percentage Sampling**. Click on the **OK** button in the **Data Flow** window.

This establishes the path from the **Aggregate Data Flow** Item to the **Percentage Sampling Data Flow** item, and you will be seeing a thin green line connecting the two.

Step 7: Configure the Percentage Sampling Data Flow Item

1. Right-click **Percentage Sampling Data Flow** item in the canvas and choose **Edit...**

This opens the **Percentage Sampling Transformation Editor** as shown in the next screenshot.



2. Read the instructions on this window. Change **Percentage of rows:** to 25 and use the default Random seed by

checking the **Use the following random seed**: checkbox in the above window. Click on the **OK** button.

When you place a check mark, a little message window opens below the random seed drop-down describing under what conditions it will be desirable to use a random seed. You will be using the default. This completes the editing of the **Percentage Sampling Data Flow** item.

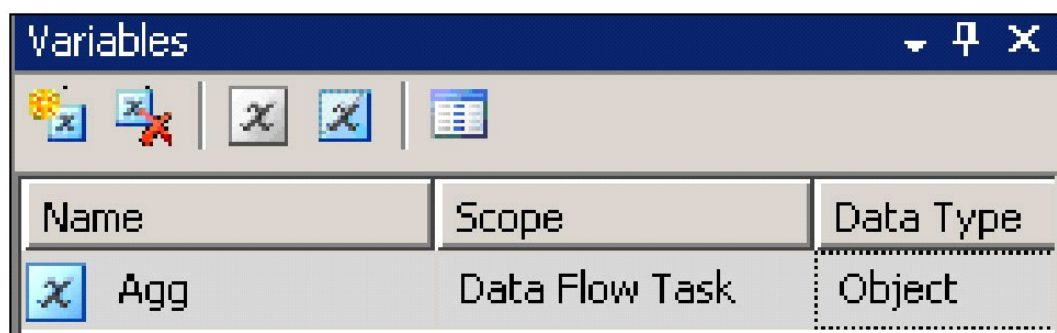
Step 8: Add a Recordset Destination Data Flow Component

We will add a **Recordset Destination Data Flow** component to the canvas and connect the output of the **Percentage Sampling Data Flow** item to the input of the **Recordset Destination**.

1. Drag and drop the **Recordset Destination** data flow component from the **Data Flow Destinations** group on to the canvas.
2. Right-click on the **Percentage Sampling Data Flow** item and from the drop-down choose **Add Path**.
3. Connect the Percentage Sampling Component's "Selected Sampling Output" to the **Recordset Destination**.

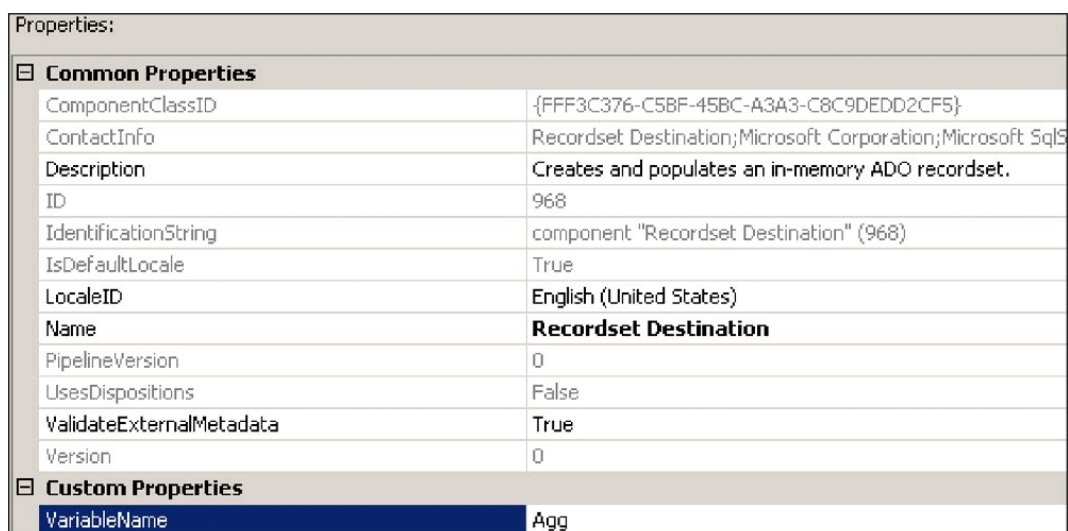
Step 9: Configure the Recordset Destination Data Flow Component

1. Right-click the **Recordset Destination Data Flow** component and from the drop-down list, click on variables to open the variables window.
2. In the variables window, click on **Add Variable**. Type in, or choose items as shown in the next screenshot (for details refer to the previous chapters).



3. Right-click the **Recordset Destination Data Flow** component and from the drop-down choose **Edit....**

This opens the **Advanced Editor for Recordset Destination**. In the **Component Properties** page, associate the above added variable **Agg** with the **VariableName** in the **Custom Properties** node, as shown in the next screenshot.



- Click on the **Input Columns** tabbed page and you should see both the ProductID and the UnitPrice selected in a two pane window.

If needed, you may change the Output aliases. If you do not, the default will be used. For this exercise, the Output alias for the **UnitPrice** column is chosen as, **Sum [Unit Price]** and **Product ID** for ProductID.

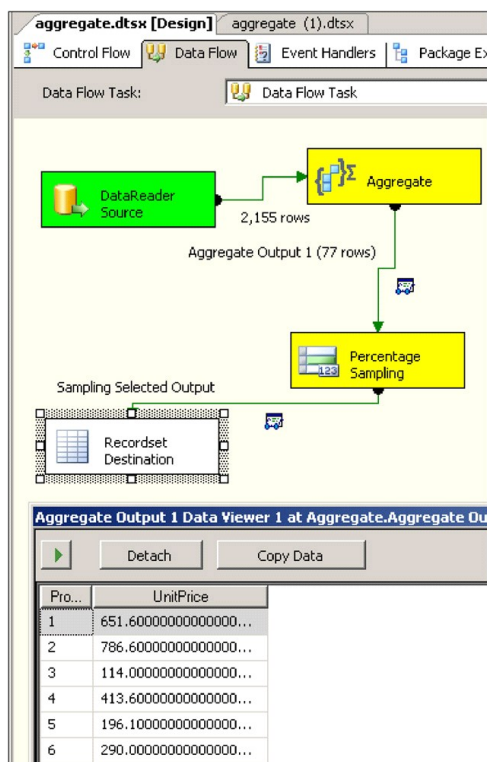
- Click on the **OK** button on the **Advanced Editor for Recordset Destination**.

This completes the Recordset Destination editor configuration.

- Right-click on the path from **Aggregate Data Flow** item to the **Percentage Sampling Data** item and add a **Grid type Data Viewer** as described in a Chapter 6.
- Similarly, add a **Grid type Data Viewer** to the path from **Percentage Sampling Data Flow** item to the **Recordset Destination**.

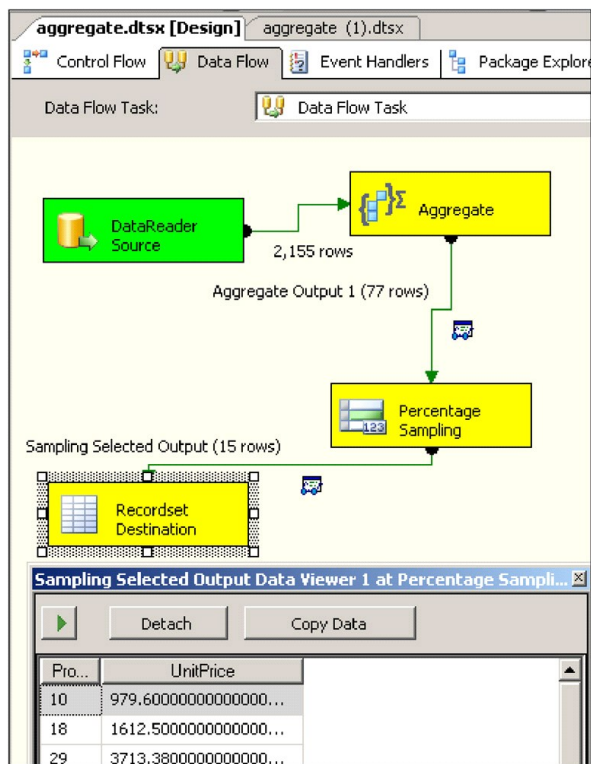
Step 10: Build and Execute the Package, and Review Results

Build the project and execute the package. The program runs and after a while you will see that the DataReader Source turns green while **Aggregate** and **Percentage Sampling** are yellow. 2155 rows transferred to the Aggregate which has produced 77 grouped outputs. The Aggregate output is displayed in the following **Data Viewer** shown next (only part of it is shown).



- Click on the green arrow head in the **Aggregate Output 1 Data Viewer1 at Aggregate.Aggregate Output 1** window.

This opens the path for the second data viewer as shown in the next screenshot, wherein 15 rows are passed from the **Percentage Sampling** component to the **Recordset Destination**. The number of rows that are picked by the **Percentage Sampling** depends on a random algorithm.



Summary

This chapter described details of a package with an **Aggregate Data Transformation** followed by a **Percentage Sampling** component. Recordset Destination, together with data viewers, provided visual confirmation of the success of this package. The Percentage Sampling (15 rows is not 25% of 77 in this present example) may not provide an exact percentage of the input as it uses special algorithms for random data selection. Interested readers may refer to the SQL Server Books online. The **Advanced Editor** for the **Aggregate Data Flow** item has more advanced features that were not explored in this exercise.