

# **SQL Server Integration Services (SSIS) Training Kit(Part 4)**

Lesson 24: OLEDB Command  
Transformation

## OLEDB Command Transformation

"The OLE DB command transformation is a SSIS data flow item. It executes an SQL command for each row in a dataset"

### EXAMPLE

- You can run an SQL statement that inserts, updates, or deletes rows in a database table.
- The OLE DB command transformation is typically used for running parameterized queries.



## OLEDB Command Transformation

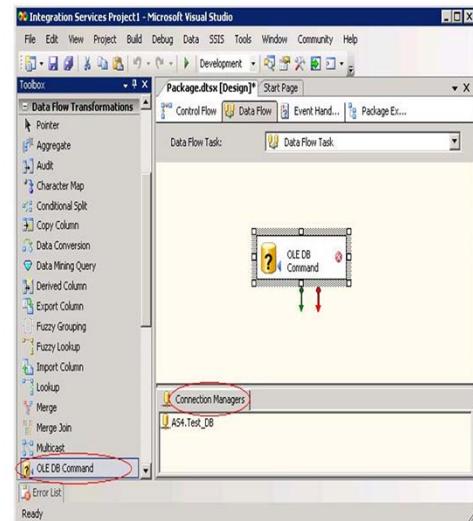
- You can configure the OLE DB Command Transformation in the following ways:
  - Provide the SQL statement that the transformation runs for each row.
  - Specify the number of seconds before the SQL statement times out.
  - Specify the default code page.
- Typically, the SQL statement includes parameters. The parameter values are stored in external columns in the transformation input, and mapping an input column to an external column maps an input column to a parameter.



Copyright © Capgemini 2015. All Rights Reserved 3

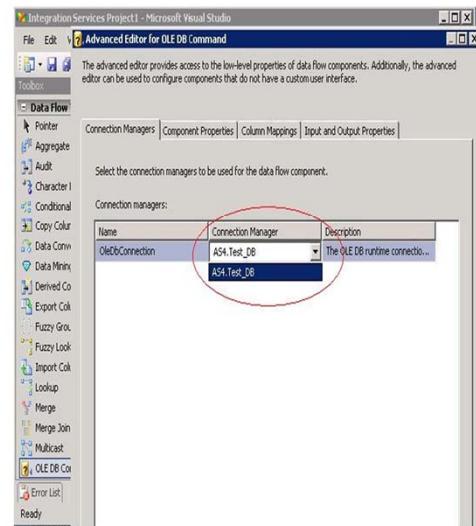
## OLEDB Command Transformation

- Drag the OLE DB Command transformation to the design surface.
- Connect the OLE DB Command transformation to the data flow by dragging a connector-the green or red arrow-from a data source or a previous transformation to the OLE DB Command transformation.



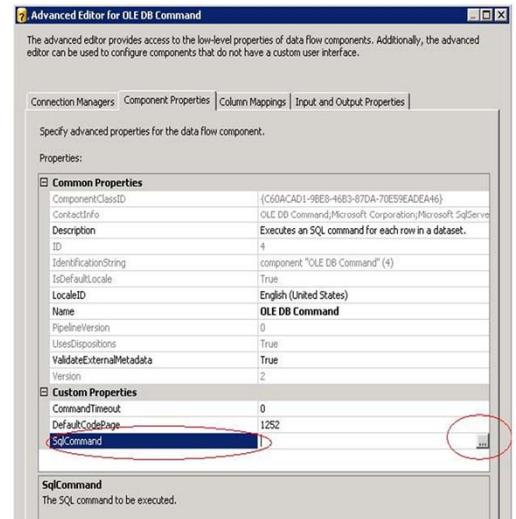
## OLEDB Command Transformation

- Right-click the component and select Edit or Show Advanced Editor.
- On the Connection Managers tab, select an OLE DB connection manager in the Connection Manager list.



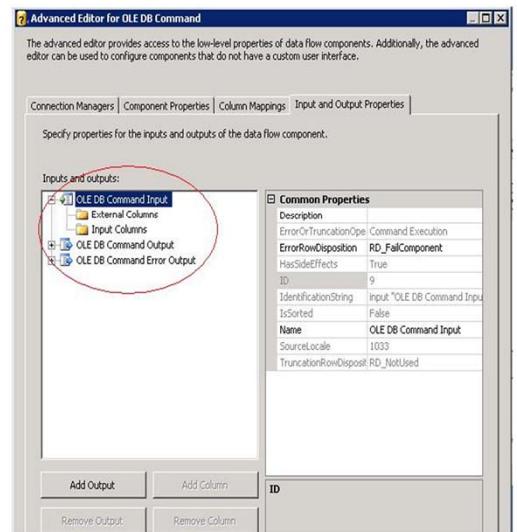
## OLEDB Command Transformation

- Click the Component Properties tab and click the ellipsis button (...) in the Sql Command box.
- In the String Value Editor, type the parameterized SQL statement using a question mark (?) as the parameter marker for each parameter.
- Click Refresh. When you click Refresh, the transformation creates a column for each parameter in the External Columns collection and sets the DBP Aram Info Flags property.



## OLEDB Command Transformation

- Click the Input and Output properties tab.
- Expand OLE DB Command Input, and then expand External Columns.
- Verify that External Columns lists a column for each parameter in the SQL statement. The column names are Param\_0, Param\_1 and so on. You should not change the names. The DataType property of each column is set to the correct data type. You should not change the data type.



## Percentage Sampling Transformation



Copyright © Capgemini 2015. All Rights Reserved 8

## Percentage Sampling Transformation

- The Percentage Sampling transformation creates a sample data set by selecting a percentage of the transformation input rows. The sample data set is a random selection of rows from the transformation input, to make the resultant sample representative of the input.
- The Percentage Sampling transformation is also useful for creating sample data sets for package development. By applying the Percentage Sampling transformation to a data flow, you can uniformly reduce the size of the data set while preserving its data characteristics. The test package can then run more quickly because it uses a small, but representative, data set.



Copyright © Capgemini 2015. All Rights Reserved 9

## Percentage Sampling Transformation

Example::

- The following example is showing a data set comprising of 10 rows. Load only randomly selected 20% of the rows in the O/P Data set.

"Account No"	"First Name"	"Last Name"	"Company"	"Address"	"City"
"10019"	"Bobbi"	"Arndt"	"Market Place"	"1000 S Nicolet..."	"Sametown"
"10023"	"Bruce"	"Beecher"	"Madson & Hut..."	"1037 W Wisco..."	"Smithville"
"10024"	"Bruce"	"Beyer"	"La Salle Clinic"	"108 E Wiscon..."	"Jonestown"
"10025"	"Butch"	"Bobbi"	"Town & Count..."	"108 Hillock Ct"	"Smithville"
"10026"	"Calla"	"Boshers"	"Saturn of App..."	"110 Fox River..."	"Smithville"
"10027"	"Carol"	"Brauer"	"Bemiss Corp."	"110 W North ..."	"Jonestown"
"10028"	"Carol"	"Braun"	"AAL Member ..."	"1115 E Glend..."	"Smithville"
"10029"	"Cheri"	"Buksyk"	"Office Support"	"1122 Milwauk..."	"Jonestown"
"10030"	"Chuck"	"Buss"	"EAA"	"1134 S Franklin..."	"Overton"
"10031"	"Chuck"	"Carpenter"	"Kurz Electric"	"115 S Drew St"	"Smithville"

## Percentage Sampling Transformation

### Output::

- The following is the output data set comprising of 3 rows after Percentage Sampling Transformation is done.

Account No	First Name	Last Name	Company	Address	City
"10019"	"Bobbi"	"Arndt"	"Market Place"	"1000 S Nicolet..."	"Sametown"
"10025"	"Butch"	"Bobbi"	"Town & Count..."	"108 Hillock Ct"	"Smithville"
"10029"	"Cheri"	"Buksyk"	"Office Support"	"1122 Milwauk..."	"Jonestown"

- In addition to the specified percentage, the Percentage Sampling transformation uses an algorithm to determine whether a row should be included in the sample output. This means that the number of rows in the sample output may not exactly reflect the specified percentage. For example specifying 10 percent for an input data set that has 25,000 rows may not generate a sample with 2,500 rows; the sample may have a few more or a few less rows.



Copyright © Capgemini 2015. All Rights Reserved 11

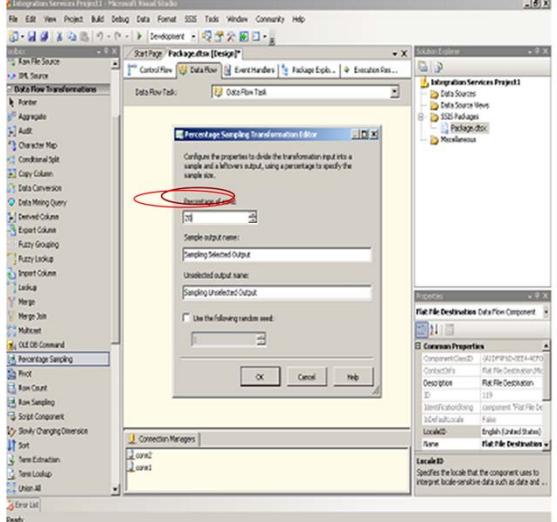
## Percentage Sampling Transformation

- Drag Percentage Sampling Transformation from Data Flow Transformations
- Link it with source.
- Double click on Percentage Sampling Transformation to open Percentage Sampling Transformation Editor.

The screenshot shows the Microsoft Visual Studio interface for an Integration Services Project. The main area is the Data Flow Task designer, where a data flow is being constructed. The flow starts with a 'Flat File Source' (represented by a document icon), followed by a 'Percentage Sampling' transformation (represented by a green diamond icon). An arrow points from the source to the transformation. From the transformation, an arrow points down to a 'Flat File Destination' (represented by a document icon). A specific output from the transformation, labeled 'Sampling Selected Output', is highlighted with a red oval. To the right of the designer is the 'Properties' window, which displays the properties for the current selected object, which is the Data Flow Task. The 'Data Flow Task' tab is selected in the Properties window. Other tabs like 'General', 'OLE DB Command', 'Script', and 'Advanced' are also visible. The 'Name' property is currently selected, showing the value 'Data Flow'. The Solution Explorer on the left shows the project structure, including packages and connection managers. The Connection Managers section lists 'src1' and 'dest1'.

## Percentage Sampling Transformation

➤ Percentage of rows:-  
Specify the percentage of rows in the input to use as a sample. The value of this property can be specified by using a property expression. Percentage of rows is set to 20

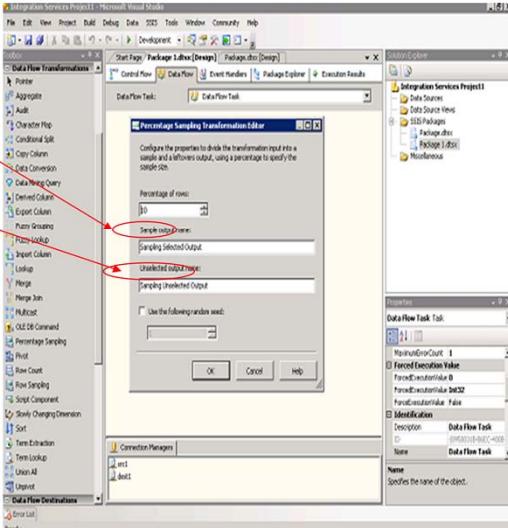


Copyright © Capgemini 2015. All Rights Reserved 13

## Percentage Sampling Transformation

➤ **Sample output name:-**  
Provide a unique name for the output that will include the sampled rows.

➤ **Unselected output name:-**  
Provide a unique name for the output that will contain the rows excluded from the sampling.

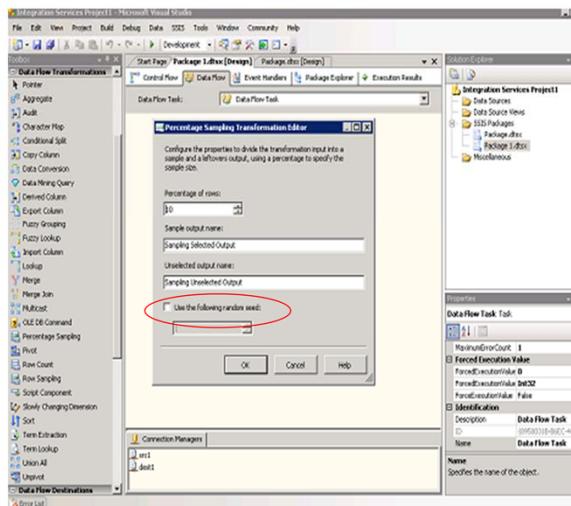


Copyright © Capgemini 2015. All Rights Reserved 14

## Percentage Sampling Transformation

➤ Use the following random seed

Specify the sampling seed for the random number generator that the transformation uses to create a sample. This is only recommended for development and testing.



Capgemini  
CONSULTING TECHNOLOGY OUTSOURCING

Copyright © Capgemini 2015. All Rights Reserved 15

## Pivot Transformation

Lesson 24:

## Pivot Transformation

- The Pivot transformation makes a normalized data set into a less normalized but more compact version by pivoting the input data on a column value.
- For example, a normalized Orders data set that lists customer name, product, and quantity purchased typically has multiple rows for any customer who purchased multiple products, with each row for that customer showing order details for a different product.
- By pivoting the data set on the product column, the Pivot transformation can output a data set with a single row per customer. That single row lists all the purchases by the customer, with the product names shown as column names, and the quantity shown as a value in the product column. Because not every customer purchases every product, many columns may contain null values



Copyright © Capgemini 2015. All Rights Reserved 17

## Pivot Transformation

- When a dataset is pivoted, input columns perform different roles in the pivoting process. A column can participate in the following ways:
  - The column is passed through unchanged to the output. Because many input rows can result only in one output row, the transformation copies only the first input value for the column.
  - The column acts as the key or part of the key that identifies a set of records.
  - The column defines the pivot. The values in this column are associated with columns in the pivoted dataset.
  - The column contains values that are placed in the columns that the pivot creates.



Copyright © Capgemini 2015. All Rights Reserved 18

## Pivot Transformation

- The following diagram shows a data set before the data is pivoted on the Product column.

<b>Usage Col Name</b>	<b>SetKey</b>	<b>PivotKey</b>	<b>PivotedVal</b>
<b>Data Records</b>	<b>Cust</b>	<b>Product</b>	<b>Qty</b>
	Kate	Ham	2
	Kate	Soda	6
	Kate	Milk	1
	Kate	Beer	12
	Fred	Milk	3
	Fred	Beer	24
	Fred	Chips	2



## Pivot Transformation

- The following diagram shows a data set after the data has been pivoted on the Product column.

Source Column	Cust	Qty	Qty	Qty	Qty	Qty
Pivot Key Value		Ham	Soda	Milk	Beer	Chips
Column Name	Cust	HAM	Soda	Milk	Beer	Chips
Data Records	Kate	2	6	1	12	
	Fred			3	24	2



## Pivot Transformation

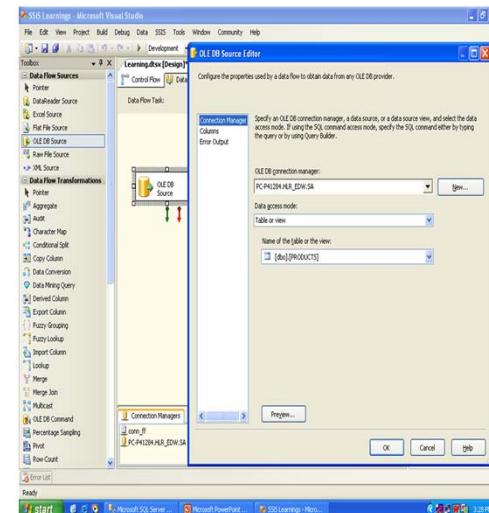
- To pivot data efficiently, which means creating as few records in the output dataset as possible, the input data must be sorted on the pivot column. If the data is not sorted, the Pivot transformation might generate multiple records for each value in the set key, which is the column that defines set membership. For example, if the dataset is pivoted on a Name column but the names are not sorted, the output dataset could have more than one row for each customer, because a pivot occurs every time that the value in Name changes.
- The Pivot transformation uses the properties on its input and output columns to define the pivot operation.
- The Pivot transformation has one input, one regular output, and one error output.



Copyright © Capgemini 2015. All Rights Reserved 21

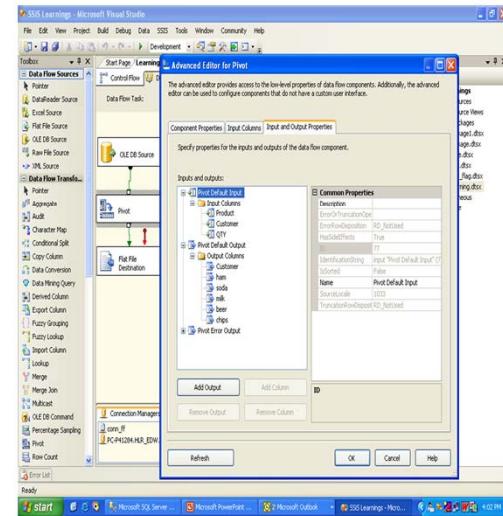
## Pivot Transformation

- Drag a ‘OLE DB Source’ from Data Flow Task Navigator to Data Flow Task Window.
- Double click or Right click on ‘OLE DB Source’ to open ‘OLE DB Source Editor’.
- Set the ‘Connection Manager’ by giving appropriate information, e.g. Connection Manager Name, Table or Query, Table Name.



## Pivot Transformation

- Drag ‘Pivot’ transformation from Data Flow Transformation Navigation Toolbox to the data flow window.
- Link ‘Pivot’ Transformation with ‘OLE DB Source’. Set the properties as mentioned in the next slide.
- Drag ‘Flat file Destination’ and select the proper flat file connection where the pivoted data will be written.



## Pivot Transformation

- The Pivot Usage property will be set as follows:

Customer column was set to 1, to indicate that it is a set key column.

Product input column was set to 2, to indicate that a column must be created for each product.

Qty input column was set to 3, to indicate that quantity values are placed into the pivot column.

- The transformation output was configured to include six columns.

The columns, which can be added by using the Advanced Editor dialog box, were named Cust, Ham, Soda, Milk, Beer, and Chips.

- The Pivot Key Value property of the Ham column was set to Ham, to indicate that the transformation should look for that value in the input column. Similarly, the Pivot Key Value property of the Soda column was set to Soda, and so on.



Copyright © Capgemini 2015. All Rights Reserved 24

## Pivot Transformation

- Columns in the transformation input were then mapped to columns in the output.

The Source Column property of the Cust column was configured to use the lineage identifier of the Cust input column. The Source Column properties of the Ham, Soda, Milk, Beer, and Chips columns were configured to use the lineage identifier of the Product input column. Another way to configure this would be to set the Source Column property of the Ham, Soda, Milk, Beer, and Chips columns to -1, which would insert the value True instead of the data value.



Copyright © Capgemini 2015. All Rights Reserved 25

## **Row Count Transformation**

Lesson 25:

## Row Count Transformation

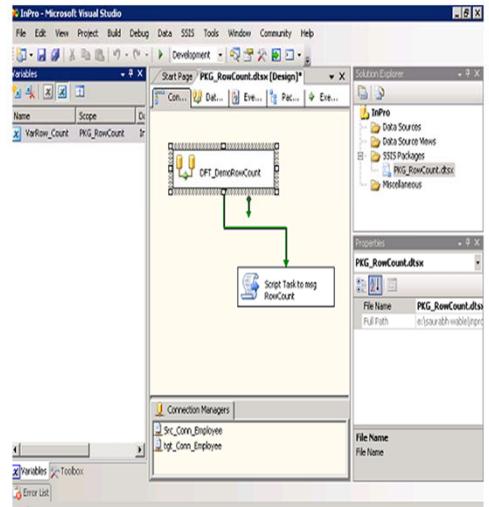
- “ The Row Count transformation counts rows as they pass through a data flow and stores the final count in a variable.”
- A SQL Server 2008 Integration Services (SSIS) package can use row counts to update the variables used in scripts, expressions, and property expressions. For example, the variable that stores the row count can update the message text in an e-mail message to include the number of rows.
- The variable that the Row Count transformation uses must already exist, and it must be in the scope of the Data Flow task to which the data flow with the Row Count transformation belongs.
- This transformation has one input and one output. It does not support an error output.



Copyright © Capgemini 2015. All Rights Reserved 27

## Row Count Transformation

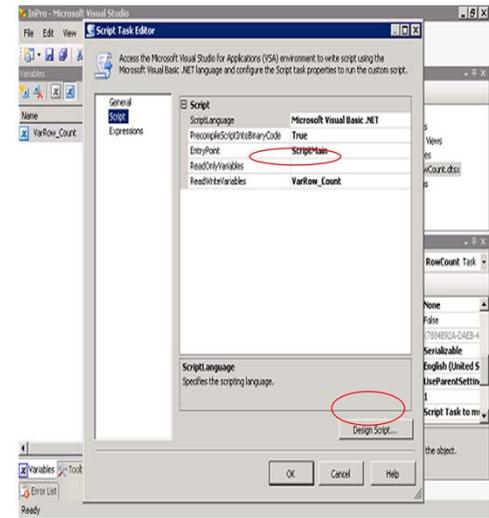
- Drag the Data flow task and script task from toolbox, link them as shown.
- Create a user variable VarRow\_Count in variable window.



Copyright © Capgemini 2015. All Rights Reserved 28

## Row Count Transformation

- Double click to open Script Task Editor.
- Describe the variable name in script tab, as shown in figure.
- Click on Design Script button.



Capgemini  
CONSULTING TECHNOLOGY OUTSOURCING

Copyright © Capgemini 2015. All Rights Reserved 29

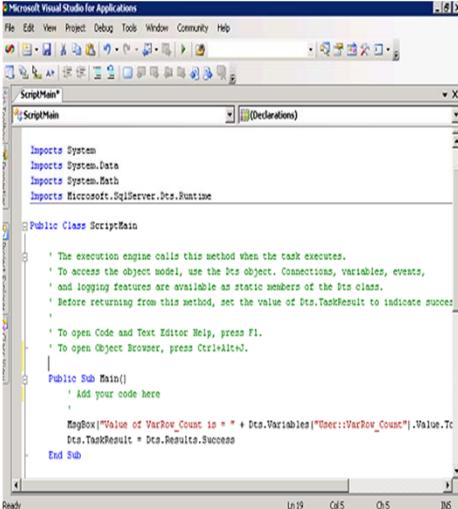
## Row Count Transformation

- Add the following code to popup a message as shown in fig. By the arrow. When data flow is completed.

```
MsgBox("Value of VarRow_Count is = " +  
Dts.Variables ("User::  
VarRow_Count").Value.ToString())
```

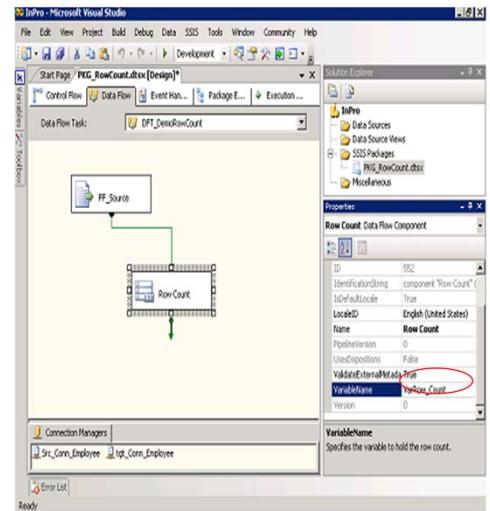
- Save the script. Close the editor and click ok on script task editor.

- Double click on Data flow task.→



## Row Count Transformation

- Configure Flat file source, Drag Row Count Transformation and link it with source.
- Select Row count Transformation to change the property Variable Name, specify the variable Var\_Row\_Count.
- See the description of property.



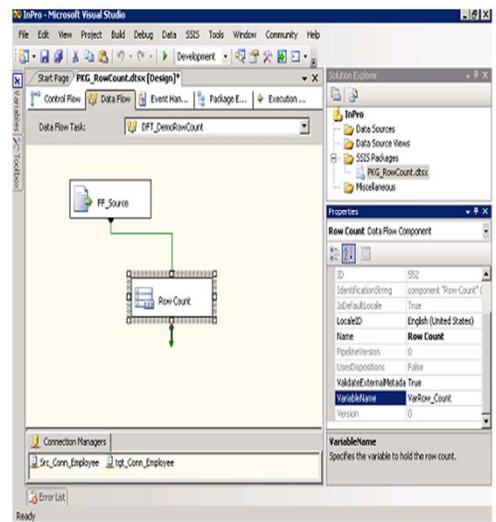
## Row Count Transformation

- Run the Package.

As FF\_source Reads 119 rows and pass it to row count transformation.

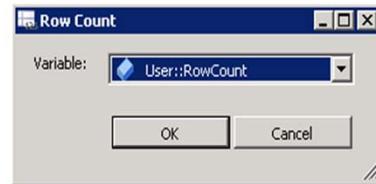
The row count transformation count the rows and assign the value to variable VarRow\_Count.

After Data flow task complete execution, the script task get activated and show the value assigned to VarRow\_Count.



## Row Count Transformation-2012 Improvements

- New User interface is introduced to assign variable to Row Count transformation.



## **Row Sampling Transformation**

Lesson 26:

## Row Sampling Transformation

- “The Row Sampling transformation is used to obtain a randomly selected subset of an input dataset. You can specify the exact size of the output sample, and specify a seed for the random number generator”.
- “There are many applications for random sampling. For example, a company that wanted to randomly select 50 employees to receive prizes in a lottery could use the Row Sampling transformation on the employee database to generate the exact number of winners”.
- “The Row Sampling transformation is also useful during package development for creating a small but representative dataset. You can test package execution and data transformation with richly representative data, but more quickly because a random sample is used instead of the full dataset. Because the sample dataset used by the test package is always the same size, using the sample subset also makes it easier to identify performance problems in the package”.



Copyright © Capgemini 2015. All Rights Reserved 35

Traditional tabulated presentation v/s graphical representation.  
Tables don't determine trends quickly as line charts do. Other

## Row Sampling Transformation

- “The Row Sampling transformation creates a sample dataset by selecting a specified number of the transformation input rows. Because the selection of rows from the transformation input is random, the resultant sample is representative of the input. You can also specify the seed that is used by the random number generator, to affect how the transformation selects rows”.
- “Using the same random seed on the same transformation input always creates the same sample output. If no seed is specified, the transformation uses the tick count of the operating system to create the random number. Therefore, you could use the same seed during testing, to verify the transformation results during the development and testing of the package, and then change to a random seed when the package is moved into production”.



Copyright © Capgemini 2015. All Rights Reserved 36

## Row Sampling Transformation

Example::

- The following example is showing a data set comprising of 10 rows. Load only randomly selected 5 rows in the O/P Data set.

	POLICYNUMBER	CARRIERCODE	PARTYID	OCCUPATION	ANNUALEARNINGS	ICD9	PHRASEID	MONTHSEMPLOYED	YEARSEMPLOYED	MODIFIEDDATE
1	POL1	HANN	1	Cricketer	500000	1	1	24	2	2007-12-01 00:00:00.000
2	POL2	HANN	2	Cricketer	200000	1	1	10	1	2007-12-02 00:00:00.000
3	POL3	HANN	3	Businessmen	1000000	1	1	60	5	2007-02-24 00:00:00.000
4	POL4	HANN	4	Businessmen	100000	1	11	5	1	2007-02-24 00:00:00.000
5	POL5	HANN	5	Businessmen	500000	1	1	20	2	2007-12-02 00:00:00.000
6	POL6	HANN	6	Service	200000	1	1	12	1	2006-12-06 00:00:00.000
7	POL7	HANN	7	Service	100000	1	1	24	2	2007-10-25 00:00:00.000
8	POL8	HANN	8	CA	600000	1	1	12	1	2007-06-24 00:00:00.000
9	POL9	HANN	9	Builder	1E+07	1	1	48	4	2006-07-06 00:00:00.000
10	POL10	HANN	10	Pilot	6000000	11	1	24	2	2006-10-10 00:00:00.000



## Row Sampling Transformation

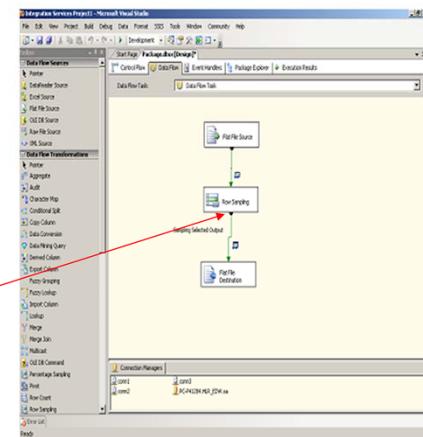
### Output:

- The following is the output data set comprising of 5 rows after Row Sampling Transformation is done. This is the output in case of Sampling Selected Output. The records not displayed here are part of Sampling Unselected Output.

	POLICYNUMBER	CARRIERCODE	PARTYID	OCCUPATION	ANNUALEARNINGS	ICD9	PHRASEID	MONTHSEMLOYED	YEARSEMLOYED	MODIFIEDDATE
1	POL1	HANN	1	Cricketer	500000	1	1	24	2	2007-12-01 00:00:00.000
2	POL7	HANN	7	Service	100000	1	1	24	2	2007-10-25 00:00:00.000
3	POL10	HANN	10	Pilot	6000000	11	1	24	2	2006-10-10 00:00:00.000
4	POL6	HANN	6	Service	200000	1	1	12	1	2006-12-06 00:00:00.000
5	POL5	HANN	5	Businessmen	500000	1	1	20	2	2007-12-02 00:00:00.000

## Row Sampling Transformation

- Drag Row Sampling Transformation from Data Flow Transformations.
- Link it with source.
- Double click on Row Sampling Transformation to open Percentage Sampling Transformation Editor.
- Optionally Data viewer can be set to view the I/P & O/P Data.



Copyright © Capgemini 2015. All Rights Reserved 39

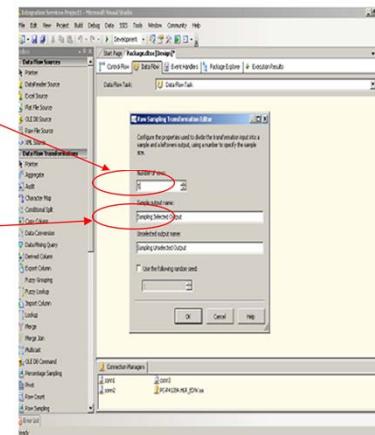
## Row Sampling Transformation

- Number of rows:-

Specify the Number of rows in the input to use as a sample. The value of this property can be specified by using a property expression. Here number Number of rows is set to 5.

- Sample output name

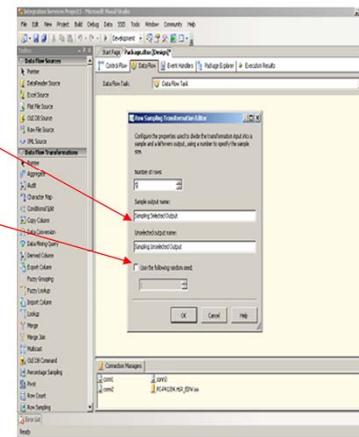
Provide a unique name for the output that will include the sampled rows. The name provided will be displayed within the SSIS Designer



## Row Sampling Transformation

- Unselected output name

- Provide a unique name for the output that will contain the rows excluded from the sampling. The name provided will be displayed within the SSIS Designer.



- Use the following random seed

- Specify the sampling seed for the random number generator that the transformation uses to create a sample. This is only recommended for development and testing. The transformation uses the Microsoft Windows tick count if a random seed is not specified.

## **Script Component Transformation**

Lesson 27:

## Script Component Transformation

- “The Script component extends the data flow capabilities of Microsoft SQL Server 2008 Integration Services (SSIS) packages with custom code written in Microsoft Visual Basic .NET that is compiled and executed at package run time”.
- “The Script component simplifies the development of a custom data flow source, transformation, or destination when the sources, transformations, and destinations included with SQL Server 2008 Integration Services do not fully satisfy your requirements”.
- “After you configure the component with the expected inputs and outputs, it writes all the required infrastructure code for you, letting you focus exclusively on the code that is required for your custom processing”.



Copyright © Capgemini 2015. All Rights Reserved 43

## Comparison between Script Component and Script Task

- Similarities:

Feature	Description
Two design-time modes by specifying	In both the component and the task, you begin properties in the editor, and then switch to the development environment to write code.
Visual Studio for Applications (VSA) code development environment	Both the component and the task use the same VSA IDE and the Visual Basic .NET programming language.
Scripts can be precompiled	Both the component and the task have a Boolean property, <b>True</b> by default, that lets you specify that the script should be precompiled into binary code, permitting faster execution, but at the cost of increased package size.



Copyright © Capgemini 2015. All Rights Reserved 44

## Differences:

Feature	Script Component	Script Task
Control flow / Data flow	The Script component is configured on the Data Flow tab of the designer and represents a source, transformation, or destination within the Data Flow task.	The Script task is configured on the Control Flow tab of the designer and runs outside the data flow of the package.
Purpose	You must specify whether you want to create a source, transformation, or destination with the Script component.	A Script task can accomplish almost any general-purpose task.
Execution	A Script component runs once, but normally it runs its main processing routine once for each row of data in the data flow.	A Script task runs custom code at some point in the package workflow. Unless you place it in a loop container or an event handler, it only runs once.



Copyright © Capgemini 2015. All Rights Reserved 45

## Script Component Transformation

The Input and Outputs page of the Script Transformation Editor is displayed for sources, transformations, and destinations. On this page, you add, remove, and configure inputs, outputs, and output columns that you want to use in your custom script, within the following limitations:

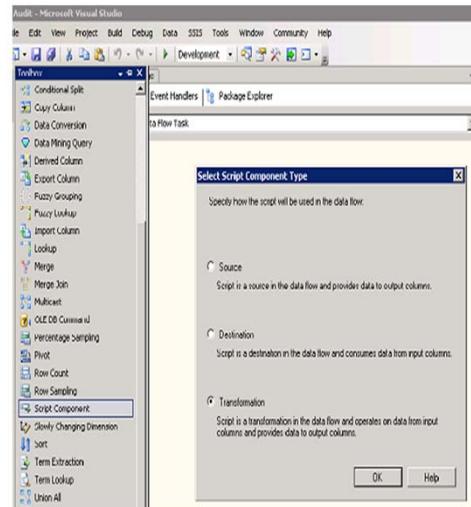
- When used as a source, the Script component has no input and supports multiple outputs.
- When used as a transformation, the Script component supports one input and multiple outputs.
- When used as a destination, the Script component supports one input and has no outputs.



Copyright © Capgemini 2015. All Rights Reserved 46

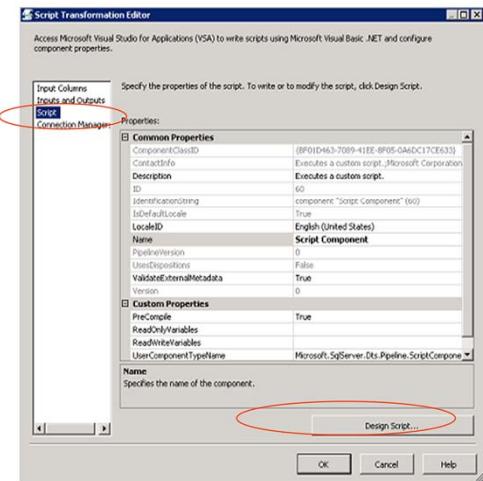
## Script Component Transformation

- Drag and drop the script component into work area.
- Select the use for the script and press OK.



## Script Component Transformation

- Right click on the script component and edit script that will open the script transformation editor.
- Go to script then as we can see there is a design script button click on it.
- Design script will open the Vbscript and we can write script in that for particular task.



## **Slowly Changing Dimension Transformation**

Lesson 28:

## Slowly Changing Dimension Transformation

- “The Slowly Changing Dimension transformation coordinates the updating and inserting of records in data warehouse dimension tables.”
- The Slowly Changing Dimension transformation supports four types of changes:

Changing Attribute  
Historical Attribute  
Fixed Attribute  
Inferred Member.



Copyright © Capgemini 2015. All Rights Reserved 50

## Slowly Changing Dimension Transformation

The Slowly Changing Dimension transformation provides the following functionality for managing slowly changing dimensions:

- Matching incoming rows with rows in the lookup table to identify new and existing rows. (New)
- Identifying incoming rows that contain changes when changes are not permitted. (Type 0)
- Identifying incoming rows that contain historical changes that require insertion of new records and the updating of expired records.(Type2)
- Detecting incoming rows that contain changes that require the updating of existing records. (Type1)
- Identifying inferred member records that require updating.



Copyright © Capgemini 2015. All Rights Reserved 51

## Slowly Changing Dimension Transformation

<b>Changing Attributes Updates Output</b>	The record in the lookup table is updated. This output is used for changing attribute rows.
<b>Fixed Attribute Output</b>	The values in rows that must not change do not match values in the lookup table. This output is used for fixed attribute rows.
<b>Historical Attributes Inserts Output</b>	The lookup table contains no matching rows. The row is inserted. This output is used for historical attribute rows.
<b>Inferred Member Updates Output</b>	Rows for inferred dimension members are inserted. This output is used for inferred member rows.
<b>New Output</b>	The lookup table contains no matching rows. The row is added to the dimension table. This output is used for new rows and changes to historical attributes rows.
<b>Unchanged Output</b>	The values in the lookup table match the row values. This output is used for unchanged rows.



## Slowly Changing Dimension Transformation

- At run time, the Slowly Changing Dimension transformation first tries to match the incoming row to a record in the lookup table. If no match is found, the incoming row is a new record; therefore, the Slowly Changing Dimension transformation performs no additional work, and directs the row to New Output.
- If a match is found, the Slowly Changing Dimension transformation detects whether the row contains changes. If the row contains changes, the Slowly Changing Dimension transformation identifies the update type for each column and directs the row to the Changing Attributes Updates Output, Fixed Attribute Output, Historical Attributes Inserts Output, or Inferred Member Updates Output.
- If the row is unchanged, the Slowly Changing Dimension transformation directs the row to the Unchanged Output.



Copyright © Capgemini 2015. All Rights Reserved 53

## Slowly Changing Dimension Transformation

- EXAMPLE: The table shows records existing in Employee Details table.
- Now, we are inserting few records, amongst which some records.
- Slowly changing Dimension Transformation will update those records which are existing.
- The records which doesn't exist are inserted as new records.

Already Existing Records

Employee Id	Employee Name	Department	Salary
5457	John Mathew	Ins	\$4,000
5466	Pat Underson	Mfg	\$2,500
5367	Tom Hudson	Prod	\$3,000

Records to be Inserted

Employee Id	Employee Name	Department	Salary
5457	John Mathew	Ins	\$4,000
5459	Khris Haris	Prod	\$3,700
5466	Pat Underson	Mfg	\$4,200
5367	Tom Hudson	Prod	\$3,000

Records After Insertion

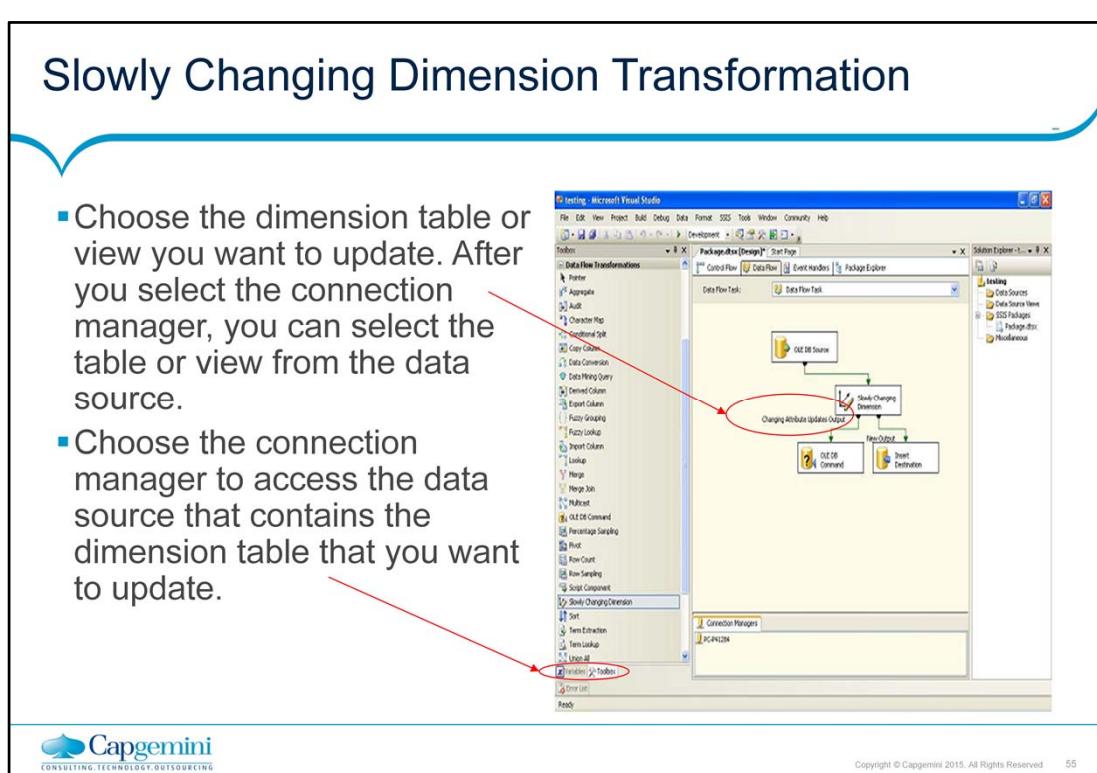
Employee Id	Employee Name	Department	Salary
5457	John Mathew	Ins	\$4,000
5459	Khris Haris	Prod	\$3,700
5466	Pat Underson	Mfg	\$4,200
5367	Tom Hudson	Prod	\$3,000



Copyright © Capgemini 2015. All Rights Reserved 54

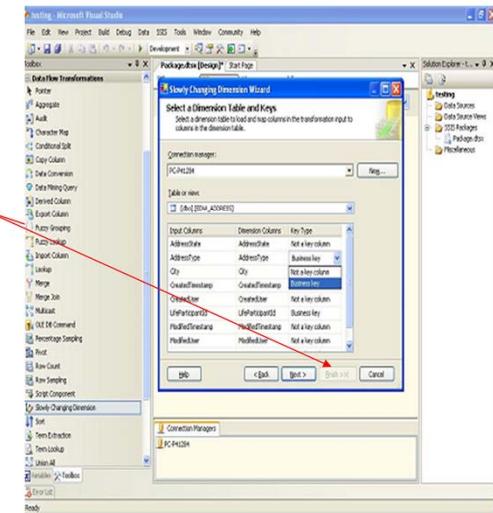
## Slowly Changing Dimension Transformation

- Choose the dimension table or view you want to update. After you select the connection manager, you can select the table or view from the data source.
- Choose the connection manager to access the data source that contains the dimension table that you want to update.



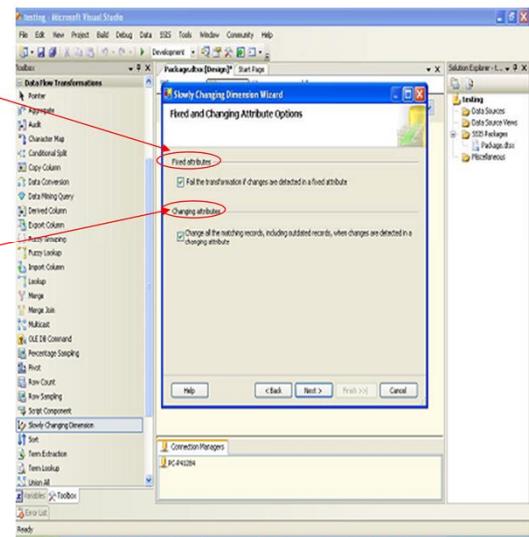
## Slowly Changing Dimension Transformation

- Set key attributes on columns and map input columns to columns in the dimension table. You must choose at least one business key column in the dimension table and map it to an input column. Other input columns can be mapped to columns in the dimension table as non-key mappings.
- Choose the change type for each column.



## Slowly Changing Dimension Transformation

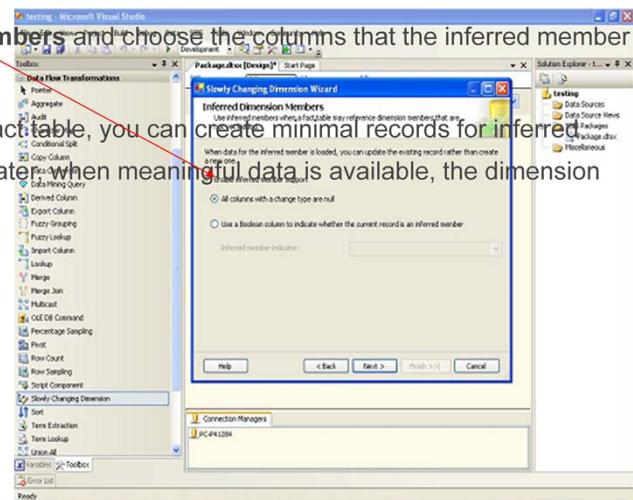
- Set fixed and changing attribute options. For the Fixed attribute change type, you can specify whether the Slowly Changing Dimension transformation fails when changes are detected in these columns. For the Changing attribute change type, you can specify whether all matching records, including outdated records, are updated.
- Set historical attribute options.



## Slowly Changing Dimension Transformation

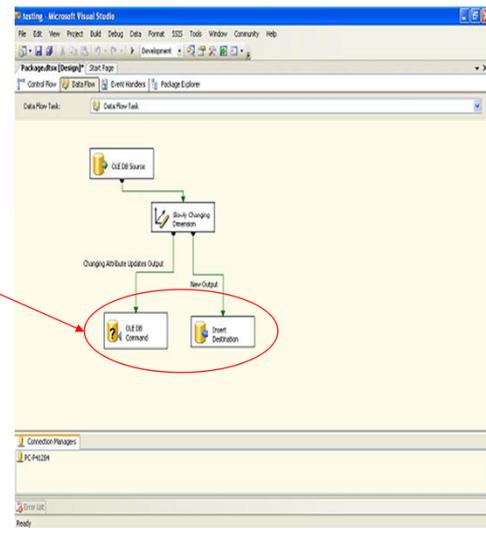
- Specify support for **inferred members** and choose the columns that the inferred member record contains.

When loading measures into a fact table, you can create minimal records for inferred members that do not yet exist. Later, when meaningful data is available, the dimension records can be updated.



## Slowly Changing Dimension Transformation

- Review the configurations that the Slowly Changing Dimension Wizard builds and click 'Finish'.
- Depending on which change types are supported, different sets of data flow components are added to the package.



## Sort Transformation

Lesson 29:

## Sort Transformation

- The Sort transformation sorts input data in ascending or descending order and copies the sorted data to the transformation output.
- You can apply multiple sorts to an input; each sort is identified by a numeral that determines the sort order. The column with the lowest number is sorted first, the sort column with the second lowest number is sorted next, and so on.
- For example, if a column named CountryRegion has a sort order of 1 and a column named City has a sort order of 2, the output is sorted by country/region and then by city.
- A positive number denotes that the sort is ascending, and a negative number denotes that the sort is descending. Columns that are not sorted have a sort order of 0. Columns that are not selected for sorting are automatically copied to the transformation output together with the sorted columns.



Copyright © Capgemini 2015. All Rights Reserved 61

Traditional tabulated presentation v/s graphical representation.  
Tables don't determine trends quickly as line charts do. Other

## Sort Transformation

- “The Sort transformation includes a set of comparison options to define how the transformation handles the string data in a column”.
- “The Sort transformation can also remove duplicate rows as part of its sort. Duplicate rows are rows with the same sort key values. The sort key value is generated based on the string comparison options being used, which means that different literal strings may have the same sort key values. The transformation identifies rows in the input columns that have different values but the same sort key as duplicates”.



Copyright © Capgemini 2015. All Rights Reserved 62

## Sort Transformation

There are options available in editor for configuration of sort component.

- Available Input Columns
- Using the check boxes, specify the columns to sort.
- Name
- View the name of each available input column.
- Pass through
- Indicate whether to include the column in the sorted output.
- Input Column
- Select from the list of available input columns for each row. Your selections are reflected in the check box selections in the Available Input Columns table.
- Output Alias
- Type an alias for each output column. The default is the name of the input column; however, you can choose any unique, descriptive name.



Copyright © Capgemini 2015. All Rights Reserved 63

## Sort Transformation

- Sort Type
  - Indicate whether to sort in ascending or descending order.
- Sort Order
  - Indicate the order in which to sort columns. This must be set manually for each column.
- Comparison Flags
  - For information about the string comparison options, see Comparing String Data.
- Remove rows with duplicate sort values
  - Indicate whether the transformation copies duplicate rows to the transformation output, or creates a single entry for all duplicates, based on the specified string comparison options.



Copyright © Capgemini 2015. All Rights Reserved 64

## Sort Transformation

Example::

- The following example is showing a data set comprising of multiple rows.
- Load only those records meeting the following criteria:
- Department & Salary should be in the ascending order.
- Output data set should only contain Department-wise Salary.

"Account No"	"First Name"	"Last Name"	"Company"	"Address"	"City"	"State"	"Zip"	"Department"	"Salary"
"10019"	"Bobbii"	"Arndt"	"Market Place"	"1000 S Nicolet..."	"Sametown"	"TX"	"99136-1221"	"100.00"	"218.00"
"10023"	"Bruce"	"Beecher"	"Madson & Hut..."	"1037 W Wisco..."	"Smithville"	"AK"	"99140-1599"	"120.00"	"222.00"
"10024"	"Bruce"	"Beyer"	"La Salle Clinic"	"108 E Wisconsin..."	"Jonestown"	"AK"	"99143-1803"	"125.00"	"223.00"
"10025"	"Butch"	"Bobbi"	"Town & Count..."	"108 Hilltop Ct"	"Smithville"	"AK"	"99166-3208"	"130.00"	"224.00"
"10026"	"Callie"	"Bosherz"	"Saturn of App..."	"110 Fall River..."	"Smithville"	"AK"	"99166-3425"	"135.00"	"225.00"
"10028"	"Carol"	"Brauer"	"Eagle Corp."	"1100 Monroe..."	"Jonestown"	"AK"	"99140-1754"	"140.00"	"226.00"
"10029"	"Caro"	"Braun"	"AAL Member ..."	"1115 E Grand...	"Smithville"	"AK"	"99111-5715"	"145.00"	"227.00"
"10029"	"Cheri"	"Buoyk"	"Office Support"	"1122 Milwaukee..."	"Jonestown"	"AK"	"99401-4175"	"150.00"	"228.00"
"10030"	"Chuck"	"Buss"	"EAA"	"1134 S Franklin..."	"Overton"	"AK"	"99401-9903"	"155.00"	"229.00"
"10031"	"Chuck"	"Carpenter"	"Kura Electric"	"115 S Drew St"	"Smithville"	"AK"	"99433-9418"	"160.00"	"230.00"
"10032"	"Chuck"	"Can"	"Alpha 1"	"1151 Valley F...	"Jonestown"	"AK"	"99451"	"165.00"	"231.00"
"10034"	"Colleen"	"Casperon"	"Valley Trust C...	"120 N Morriso..."	"Smithville"	"AK"	"99501"	"175.00"	"233.00"
"10035"	"Connie"	"Catterton"	"Fox Commun..."	"1200 N Perkins..."	"Smithville"	"AK"	"99901"	"180.00"	"234.00"
"10036"	"Connie"	"Clay"	"Valley Lawn C..."	"121 N Douglas..."	"Smithville"	"AK"	"99901"	"185.00"	"235.00"
"10037"	"Craig"	"Collar"	"Network Healt..."	"1216 W Wisco..."	"Moretown"	"AK"	"99901"	"190.00"	"236.00"
"10038"	"Dan"	"Coppenger"	"Kolossal Toyota"	"1221 N Lawe St"	"Smithville"	"AK"	"99901-2104"	"195.00"	"237.00"
"10039"	"Dan"	"Dog"	"AAL"	"1222 N Superi..."	"Smithville"	"AK"	"99901-2990"	"200.00"	"238.00"
"10040"	"Darlene"	"Dantzier"	"WDFP"	"124 W Wiscon..."	"Dime Box"	"AK"	"99901-4848"	"205.00"	"239.00"
"10041"	"Darlene"	"Dellenmann"	"AAL"	"130 E Franklin..."	"Smithville"	"AK"	"99901-7140"	"210.00"	"240.00"
"10042"	"Darren"	"Dinkl"	"Novus Health ..."	"1300 E Calum..."	"Smithville"	"AK"	"99903-2099"	"215.00"	"241.00"
"10043"	"David"	"Dockry"	"AAL"	"1302 S Ritter..."	"Smithville"	"AK"	"99903-2692"	"220.00"	"242.00"



## Sort Transformation

Output::

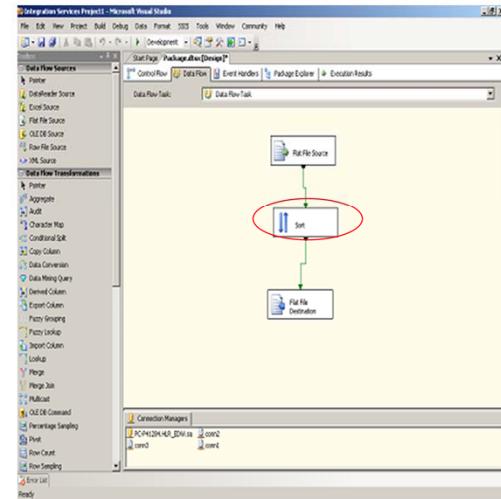
- The following is the output records meeting up the criteria set earlier.

"Account No"	"First Name"	"Last Name"	"Company"	"Addre...	"City"	"State"	"Zip"	"Department"	"Salary"
					"Jonestown"			"125.00"	"223.00"
					"Smithville"			"130.00"	"224.00"
					"Jonestown"			"135.00"	"225.00"
					"Jonestown"			"140.00"	"226.00"
					"Smithville"			"145.00"	"227.00"
					"Jonestown"			"150.00"	"228.00"
					"Overton"			"155.00"	"229.00"
					"Smithville"			"160.00"	"230.00"
					"Jonestown"			"165.00"	"231.00"
					"Smithville"			"175.00"	"233.00"
					"Smithville"			"180.00"	"234.00"
					"Smithville"			"185.00"	"235.00"
					"Moretown"			"190.00"	"236.00"
					"Smithville"			"195.00"	"237.00"
					"Smithville"			"200.00"	"238.00"
					"Dime Box"			"205.00"	"239.00"
					"Smithville"			"210.00"	"240.00"
					"Smithville"			"215.00"	"241.00"



## Sort Transformation

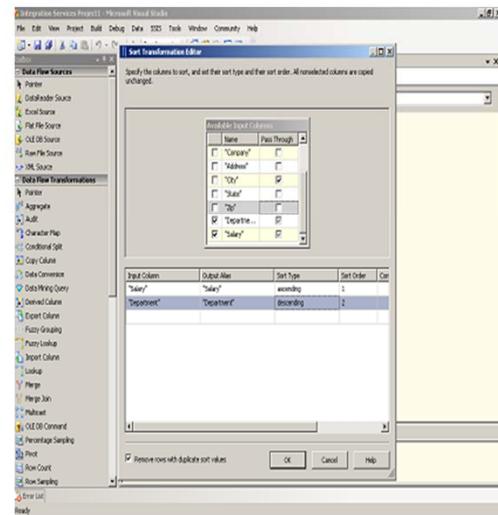
- Drag flat file Source, Sort Transformation and flat file Destination as shown
- Configure the FF\_Source with Src\_Conn\_Employee connection.
- Link output of FF\_Source to sort transformation.
- Double click to open transformation editor as shown



Copyright © Capgemini 2015. All Rights Reserved 67

## Sort Transformation

- Double click to open transformation editor as shown.
- Select columns, selected column are automatically added below.
- Select the City column in the Pass Through as this column is needed in the sorted output.
- Select the Sort type. Select Salary & Department both in ascending order.
- Select the Sort order.



Copyright © Capgemini 2015. All Rights Reserved 68

## **Term Extraction Transformation**

Lesson 30:

## Term Extraction Transformation

- The Term Extraction transformation extracts terms from text in a transformation input column, and then writes the terms to a transformation output column.
- The transformation works only with English text and it uses its own English dictionary and linguistic information about English.
- The Term Extraction transformation can extract nouns only, noun phrases only, or both nouns and noun phrases.
- A noun is a single noun; a noun phrases is at least two words, of which one is a noun and the other is a noun or an adjective. For example, if the transformation uses the nouns-only option, it extracts terms like bicycle and landscape; if the transformation uses the noun phrase option, it extracts terms like new blue bicycle, bicycle helmet, and boxed bicycles.



Copyright © Capgemini 2015. All Rights Reserved 70

## Term Extraction Transformation

### EXAMPLE

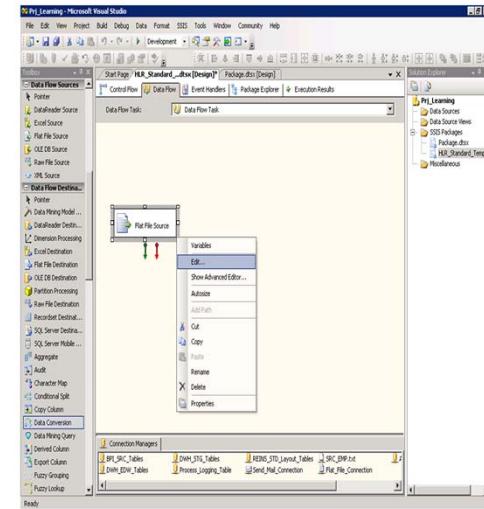
- Articles and pronouns are not extracted. I.e. the Term Extraction transformation extracts the term bicycle from the text the bicycle, my bicycle, and that bicycle.
- The Term Extraction transformation also stems nouns to extract only the singular form of a noun. For example, the transformation extracts man from men, mouse from mice, and bicycle from bicycles etc.



Copyright © Capgemini 2015. All Rights Reserved 71

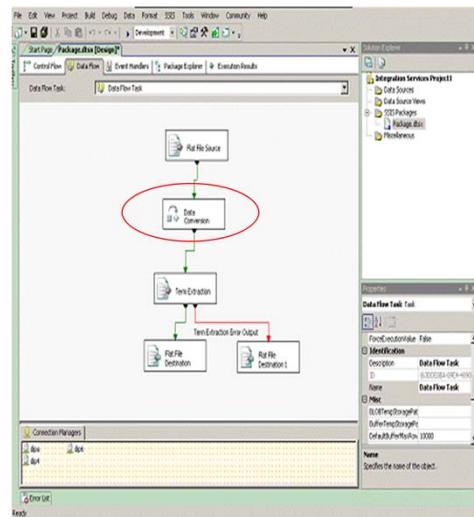
## Term Extraction Transformation

- Drag a ‘Flat File Source’ from Data Flow Source having repeated specific information. Say for eg. In employee table ‘CITY’ is often repeated.
- Double click on ‘Flat File Source’ or Right Click on ‘Flat File Source’ and select ‘Edit’
- Make appropriate change in Flat File.(e.g. Choose Flat File Connection, Select a File, Rename Column Name...etc.)



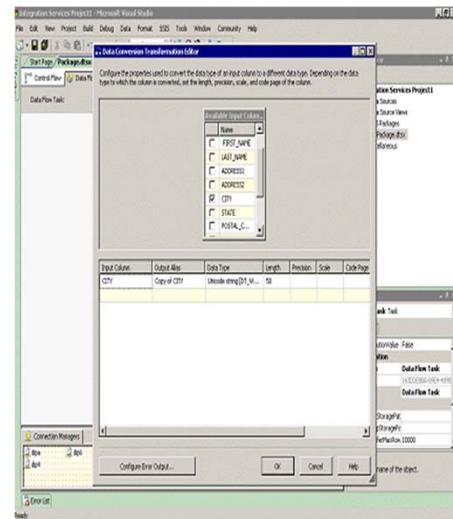
## Term Extraction Transformation

- Data conversion transformation is connected to the flat file source.
- Double click the Data conversion transformation & open up the Editor



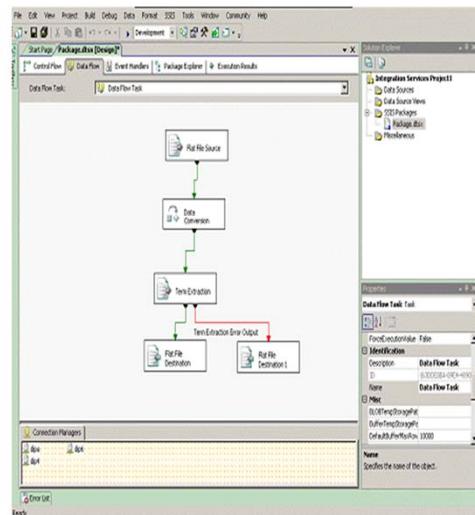
## Term Extraction Transformation

- Data type should be converted to either DT\_WSTR or DT\_NTEXT data type before passing to the ‘Term Extraction transformation’. One copy will get created of the selected item.



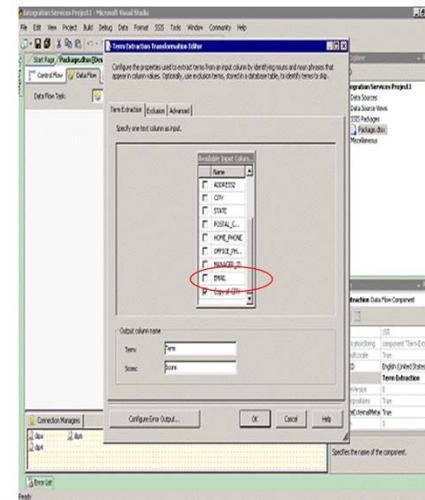
## Term Extraction Transformation

- Drag & drop the Term Extraction transformation and connect it with Data conversion transformation
- Double click the Term Extraction transformation & open up the editor.



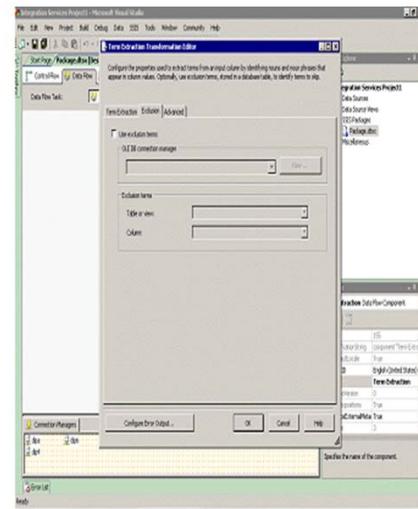
## Term Extraction Transformation

- Choose the 'Term extraction tab' and Select the check box containing the option "copy of column"(here it is 'CITY') which you have selected earlier in the Data conversion transformation editor.
- The output of the Term Extraction transformation includes only two columns. One column contains the extracted terms and the other column contains the score. The default names of the columns are Term and Score .You can change the OUTPUT TERM & SCORE name.



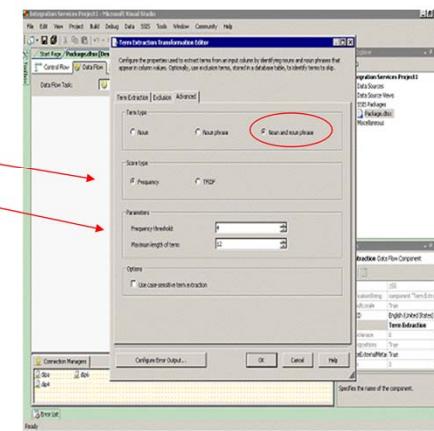
## Term Extraction Transformation

- Now you can chose the exclusion tab & use exclusion terms stored in a database table to identify terms to skip. This is optional.



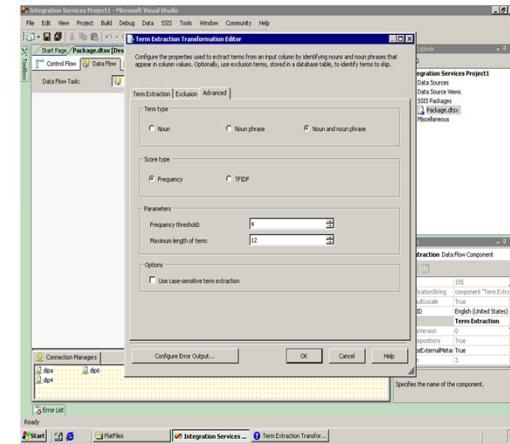
## Term Extraction Transformation

- Now click on the advanced tab you can choose various 'TERM' type options such as 'only noun', 'only phrase' or both.
- 'SCORE' type can be specified as 'frequency' or 'TFIDF' of the term.
- Depending upon the 'TERM' type selected you can choose parameters.
- Note: The TFIDF score is the product of Term Frequency and Inverse Document Frequency, defined as: TFIDF of a Term T =  $(\text{frequency of } T) * \log(\frac{\#\text{rows in Input}}{\#\text{rows having } T})$



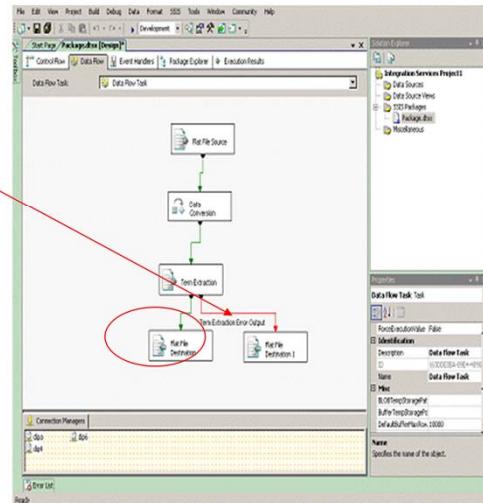
## Term Extraction Transformation

- Frequency threshold
- Specify the number of times a word or phrase must occur before extracting it. The default value is 2.
- Maximum length of term
- Specify the maximum length of a phrase in words. This option affects noun phrases only. The default value is 12.
- Use case-sensitive term
- Specify whether extraction to make the extraction case-sensitive. The default is False.



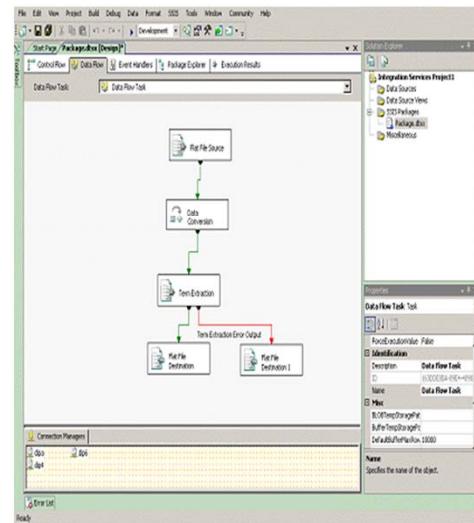
## Term Extraction Transformation

- Drag & drop the 'Flat file Destination' and connect it with the 'Term Extraction transformation' & make necessary connection in the connection manager.
- You can connect another 'Flat file Destination' for 'Term extraction Error output' which is optional.



## Term Extraction Transformation

- Click on 'Debug' > 'start Debugging' option.
- Now check out the output & compare it with input flat file.



## Term Extraction Transformation

- The Term Extraction transformation can work only with text in a column that has either the DT\_WSTR or the DT\_NTEXT data type.
- If a column contains text but does not have one of these data types, the Data Conversion transformation can be used to add a column with the DT\_WSTR or DT\_NTEXT data type to the data flow and copy the column values to the new column. The output from the Data Conversion transformation can then be used as the input to the Term Extraction transformation.



Copyright © Capgemini 2015. All Rights Reserved 82

## **Term Lookup Transformation**

Lesson 31:

## Term Lookup Transformation

- “The Term Lookup transformation matches terms extracted from text in a transformation input column with terms in a reference table”
- It counts the number of times a term in the lookup table occurs in the input data set, and writes the count together with the term from the reference table to columns in the transformation output.
- This transformation is useful for creating a custom word list based on the input text, complete with word frequency statistics.



Copyright © Capgemini 2015. All Rights Reserved 84

## Term Lookup Transformation

### EXAMPLE

- The Term Lookup performs a lookup and returns a value using the following rules:
- If the transformation is configured to perform case-sensitive matches, matches that fail a case-sensitive comparison are discarded. Example, student and STUDENT are treated as separate words.
- If a plural form of the noun or noun phrase exists in the reference table, the lookup matches only the plural form of the noun or noun phrase. Example, all instances of students would be counted separately from the instances of student.



Copyright © Capgemini 2015. All Rights Reserved 85

## Term Lookup Transformation

- If only the singular form of the word is found in the reference table, both the singular and the plural forms of the word or phrase are matched to the singular form.
- Example, if the lookup table contains student, and the transformation finds the words student and students, both words would be counted as a match for the lookup term student.



Copyright © Capgemini 2015. All Rights Reserved 86

## Term Lookup Transformation

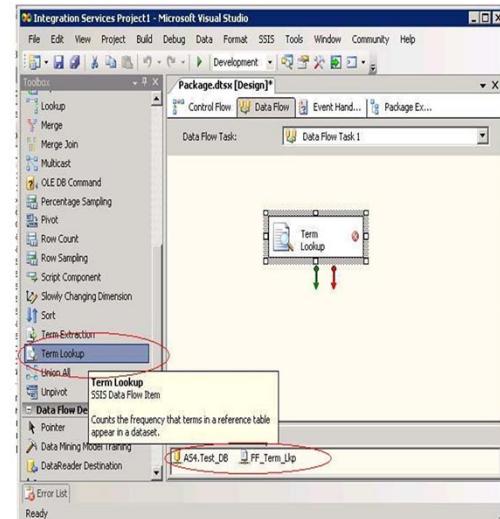
- Before the Term Lookup transformation performs a lookup, it extracts words from the text in an input column using the same method as the Term Extraction transformation:
  - Text is broken into sentences.
  - Sentences are broken into words.
  - Words are normalized.
- To further customize which terms to match, the Term Lookup transformation can be configured to perform a case-sensitive match.
- The Term Lookup transformation can match nouns and noun phrases that contain special characters, and the data in the reference table may include these characters.



Copyright © Capgemini 2015. All Rights Reserved 87

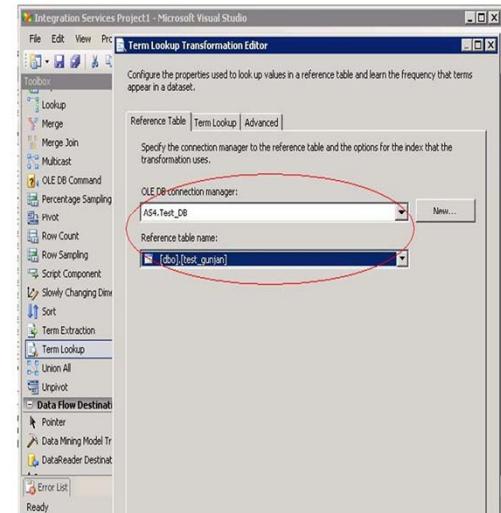
## Term Lookup Transformation

- Drag the OLE DB Command transformation to the design surface.
- Create respective connection managers.



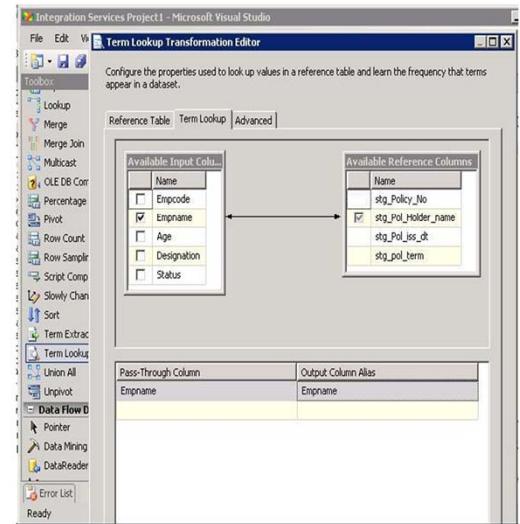
## Term Lookup Transformation

- Right click on Term Lookup and go to 'Edit' in order to change settings
- On 'Reference Table' Tab, give details regarding Reference table for lookup and other details ( like connection manager etc.)



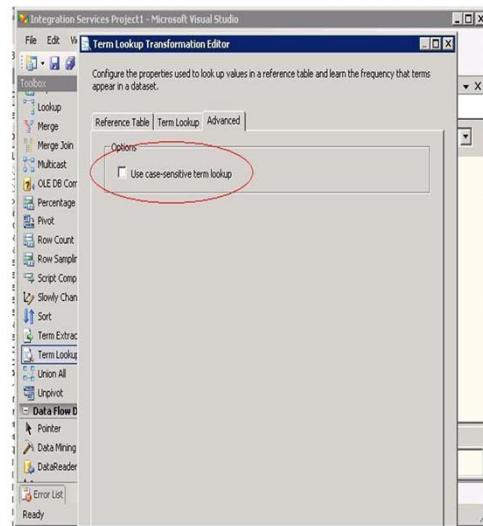
## Term Lookup Transformation

- In 'Term Lookup' tab, match available input columns with Reference columns.



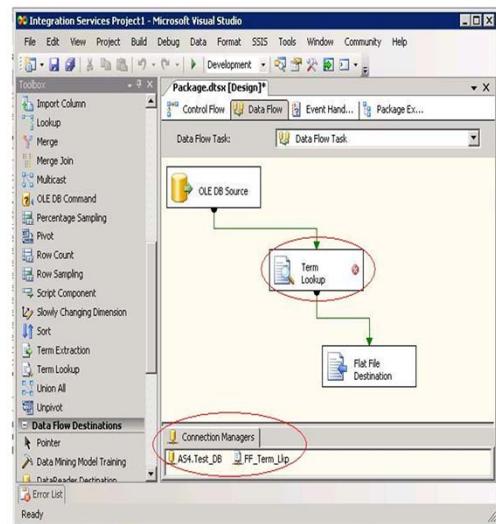
## Term Lookup Transformation

- ‘Advance Setting’ tab allows you to enable case sensitive lookup.



## Term Lookup Transformation

- Click OK.
- Finally, To save the updated package, click Save on the File menu.



Copyright © Capgemini 2015. All Rights Reserved 92

## Term Lookup Transformation

- Input Source

Global_Table_Structure	
Table_Name	Constraint_Name
Tbl_Registration	UNQ_Reg_EnolD
Tbl_Registration	PK_REG_ID
Tbl_Symptom	UNQ_ICD_SymID
Tbl_Registration	FK_EnolD

- Lookup TERM 'UNQ'

- Term Lookup Output

Tbl_Symptom	UNQ_ICD_SymID
Tbl_Registration	UNQ_Reg_EnolD

- Lookup TERM '\_I'

- Term Lookup Output

Tbl_Registration	PK_REG_ID
Tbl_Symptom	UNQ_ICD_SymID



Term Lookup Output

93

## **Union All Transformation**

Lesson 32:

## Union All Transformation

- To add and configure a Union All transformation, the package must already include at least one Data Flow task and two data sources.
- The Union All transformation combines multiple inputs. It can also be used to combine two or more heterogeneous sources like Flat File source and OLE DB source. The first input that is connected to the transformation is the reference input, and the inputs connected subsequently are the secondary inputs. The output includes the columns in the reference input.
- Using Union All Transformation you can merge data from multiple data flows, create complex datasets by nesting Union All transformations, and re-merge rows after you correct errors in the data.



Copyright © Capgemini 2015. All Rights Reserved 95

## Union All Transformation

Example::

- The following example is showing two data sets comprising multiple number of rows of 4 rows. Suppose now we have to combine the two data sets. We can achieve this by using Union All Transformation.

"Account No"	"First Name"	"Last Name"	"Company"	"Address"
"10027"	"Carol"	"Brauer"	"Benniss Corp."	"110 W North Wat...
"10028"	"Carol"	"Braun"	"AAL Member ..."	"1115 E Glendale A...
"10029"	"Cheri"	"Buksyk"	"Office Support"	"1122 Milwaukee S...
"10030"	"Chuck"	"Buss"	"EAA"	"1134 S Franklin St...
"10031"	"Chuck"	"Carpenter"	"Kurz Electric"	"115 S Drew St","S...

"Account No"	"First Name"	"Last Name"	"Company"	"Address"
"10019"	"Bobbi"	"Arndt"	"Market Place"	"1000 S Nicolet...
"10023"	"Bruce"	"Beecher"	"Madson & Hut..."	"1037 W Wisco...
"10024"	"Bruce"	"Beyer"	"La Salle Clinic"	"108 E Wiscon...
"10025"	"Butch"	"Bobbi"	"Town & Count..."	"108 Hillock Ct"...



## Union All Transformation

Output::

- The following is the output i.e the union of the two data sets.

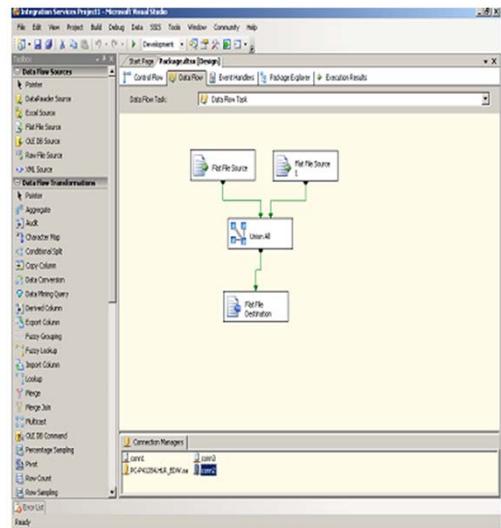
Account No	First Name	Last Name	Company	Address
"10027"	"Carol"	"Brauer"	"Bemiss Corp."	"110 W North ...
"10028"	"Carol"	"Braun"	"AAL Member ..."	"1115 E Glend...
"10029"	"Cheri"	"Buksyk"	"Office Support"	"1122 Milwauk...
"10030"	"Chuck"	"Buss"	"EAA"	"1134 S Frankli...
"10031"	"Chuck"	"Carpenter"	"Kurz Electric"	"115 S Drew St...
"10019"	"Bobbi"	"Arndt"	"Market Place"	"1000 S Nicolet...
"10023"	"Bruce"	"Beecher"	"Madson & Hut..."	"1037 W Wisco...
"10024"	"Bruce"	"Beyer"	"La Salle Clinic"	"108 E Wiscon...
"10025"	"Butch"	"Bobbi"	"Town & Count..."	"108 Hillock Ct"...



Copyright © Capgemini 2015. All Rights Reserved 97

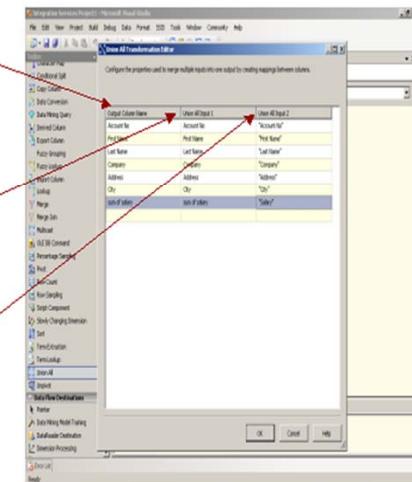
## Union All Transformation

- Open Integration Services project.
- Click the Data Flow tab, and then, from the Toolbox, drag the Union All transformation to the design surface.
- Connect the Union All transformation to the data flow by dragging a connector—the green or red arrow—from the data source or a previous transformation to the Union All transformation.
- Double-click the Union All transformation.



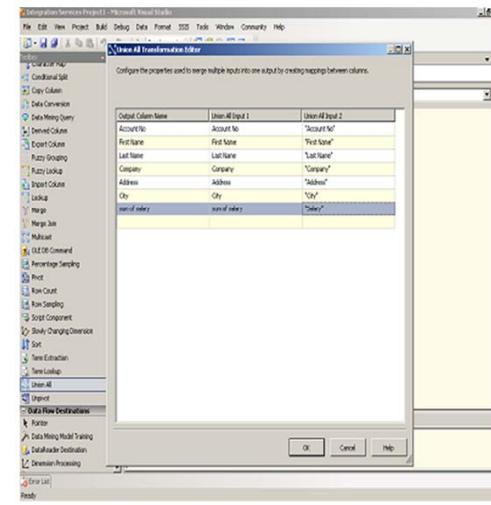
## Union All Transformation

- Output Column Name :- Type an alias for each column. The default is the name of the input column from the first (reference) input; however, you can choose any unique, descriptive name.
- Union All Input 1 :- Select from the list of available input columns in the first (reference) input. The metadata of mapped columns must match.
- Union All Input n :- Select from the list of available input columns in the second and additional inputs. The metadata of mapped columns must match



## Union All Transformation

- In the Union All Transformation Editor, map a column from an input to a column in the Output Column Name list by clicking a row and then selecting a column in the input list. Select <ignore> in the input list to skip mapping the column.
- Optionally, modify the names of columns in the Output Column Name column.
- Repeat steps 5 and 6 for each column in each input.
- Click OK.



## **Unpivot Transformation**

Lesson 33:

## Unpivot Transformation

- The Unpivot transformation makes an unnormalized dataset into a more normalized version by expanding values from multiple columns in a single record into multiple records with the same values in a single column. For example, a dataset that lists customer names has one row for each customer, with the products and the quantity purchased shown in columns in the row. After the Unpivot transformation normalizes the data set, the data set contains a different row for each product that the customer purchased.
- The following diagram shows a data set before the data is unpivoted on the Product column.

Destination Column	Cust	Qty	Qty	Qty	Qty	Qty
Pivot Key Value		Ham	Soda	Milk	Beer	Chips
Column Name	Cust	HAM	Soda	Milk	Beer	Chips
Data Records	Kate	2	6	1	12	
	Fred			3	24	2

## Unpivot Transformation

- The following diagram shows a data set after it has been unpivoted on the Product column

Pivot Key Col Name Data Records	False <b>Cust</b>	True <b>Product</b>	False <b>Qty</b>
Kate		Ham	2
Kate		Soda	6
Kate		Milk	1
Kate		Beer	12
Fred		Milk	3
Fred		Beer	24
Fred		Chips	2

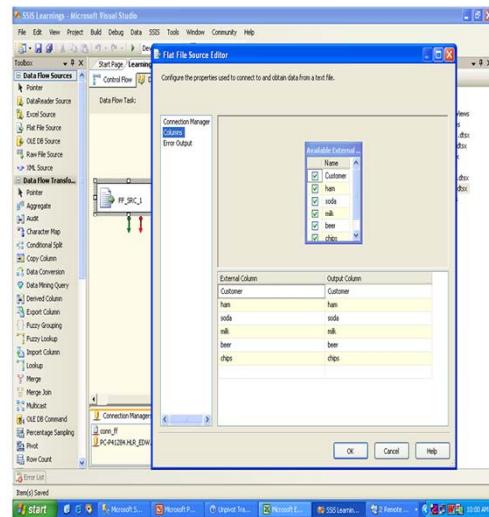
**Note:** This transformation has one in put, one output and an error output.



Copyright © Capgemini 2015. All Rights Reserved 103

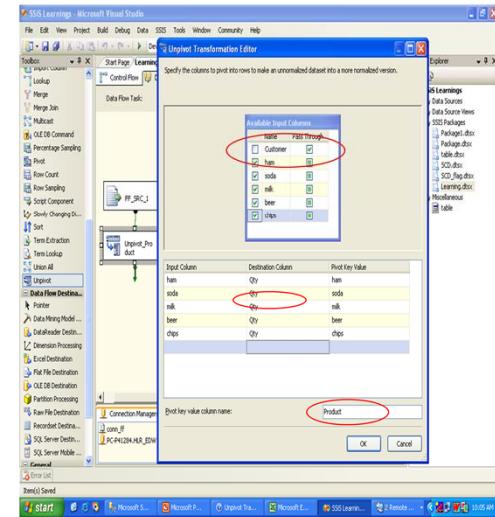
## Unpivot Transformation

- Drag a ‘Flat File Source’ from Data Flow Task Navigator to Data Flow Task Window.
- Select the flat file connection pointing to the Pivoted flat file created under the Pivot Transformation.
- Drag ‘Unpivot’ transformation from Data Flow Transformation Navigation Toolbox to the data flow window.
- Link ‘Unpivot’ Transformation with ‘OLE DB Source’.
- Double Click on ‘Unpivot’ transformation or Right click and select ‘Edit’.



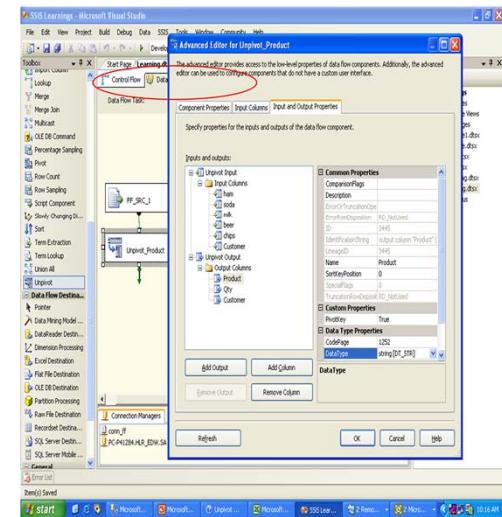
## Unpivot Transformation

- In the Unpivot Transformation Editor, Select the input columns, in this example it will be ham, soda, milk,beer,chips.
- The destination column will be set to 'Qty'.
- The Pivot key value same as the Input column names.
- the Pivot Key Value Column Name is Product and designates the new Product column into which the Ham, Soda, Milk, Beer, and Chips columns are unpivoted.
- Check Customer column pass through property.



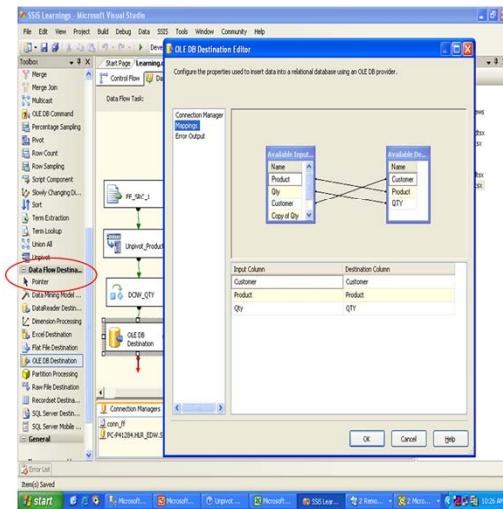
## Unpivot Transformation

- The Advanced Editor for unpivot transformation will look like:
- Check whether the input column Destination column property is set to the proper lineage id of the Output column. e.g. If Product column has lineage id = 3450 then input columns, ham, soda,etc. will have the destination column set to 3450.
- Check whether the output columns data type is properly set.e.g. Product output column will have data type as string[DT\_STR].



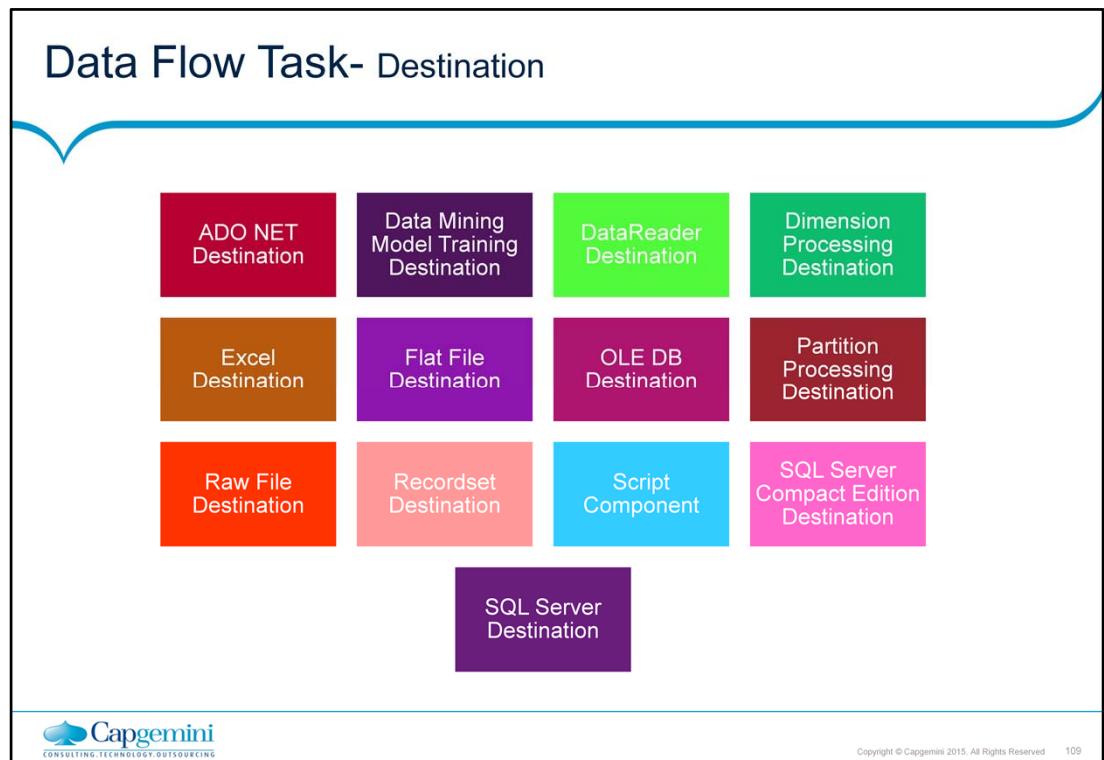
## Unpivot Transformation

- Use the data conversion transformation to convert the datatype of Qty from string to target datatype.
- Drag a ‘OLE DB Destination’ from Data Flow Task Navigator to Data Flow Task Window and link it to the ‘Data conversion’ transformation. Double click it or right click it and then select ‘Edit’ to open the ‘OLE DB Destination editor’. Edit the ‘OLE DB Destination editor’ properly to ensure the target table and mappings.



## Data Flow Task-Destination

Lesson 34:



## Data Flow Task- Destination.....

- ADO NET - The ADO NET destination loads data into a variety of ADO.NET-compliant databases that use a database table or view. You have the option of loading this data into an existing table or view, or you can create a new table and load the data into the new table.
- Excel - The Excel destination loads data into worksheets or ranges in Microsoft Excel workbooks.
- Flat File - The Flat File destination writes data to a text file. The text file can be in delimited, fixed width, fixed width with row delimiter, or ragged right format.
- OLEDB - The OLE DB destination loads data into a variety of OLE DB-compliant databases using a database table or view or an SQL command. For example, the OLE DB source can load data into tables in Microsoft Office Access and SQL Server databases.
- Data Mining Model Training - The Data Mining Model Training destination trains data mining models by passing the data that the destination receives through the data mining model algorithms. Multiple data mining models can be trained by one destination if the models are built on the same data mining structure.
- DataReader - The DataReader destination exposes the data in a data flow by using the ADO.NET DataReader interface. The data can then be consumed by other applications.
- Dimension Processing - The Dimension Processing destination loads and processes an SQL Server Analysis Services dimension.



Copyright © Capgemini 2015. All Rights Reserved 110

## Data Flow Task- Destination.....

- Partition Processing - The Partition Processing destination loads and processes an SQL Server Analysis Services partition
- Raw File - The Raw File destination writes raw data to a file. Because the format of the data is native to the destination, the data requires no translation and little parsing. This means that the Raw File destination can write data more quickly than other destinations such as the Flat File and the OLE DB destinations.
- Recordset - The Recordset destination creates and populates an in-memory ADO recordset. The shape of the recordset is defined by the input to the Recordset destination at design time.
- SQL Server compact edition - The SQL Server Compact destination writes data to SQL Server Compact databases.
- SAP BW - The SAP BW destination is the destination component of the Microsoft Connector 1.1 for SAP BW. Thus, the SAP BW destination loads data from the data flow in an Integration Services package into an SAP Netweaver BW version 7 system.
  - This destination has one input and one error output.
- SQL Server - The SQL Server destination connects to a local SQL Server database and bulk loads data into SQL Server tables and views. You cannot use the SQL Server destination in packages that access a SQL Server database on a remote server. Instead, the packages should use the OLE DB destination



Copyright © Capgemini 2015. All Rights Reserved 111