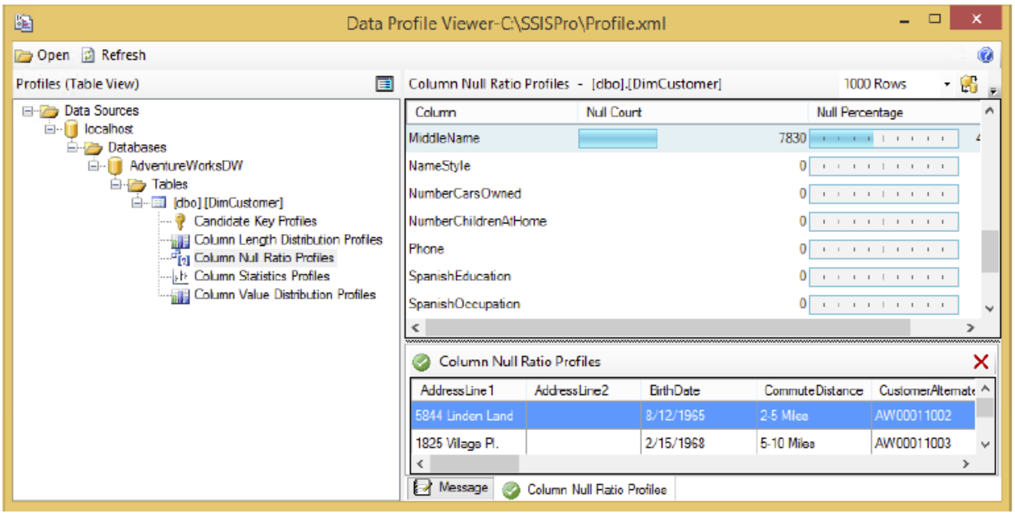
Data Tools Tasks

Contents

1. Data Profiling

- Data profiling is the process of examining data and collecting metadata about the quality of the data, about frequency of statistical patterns, interdependencies, uniqueness, and redundancy.
- This type of analytical activity is important for the overall quality and health of an operational data store (ODS) or data warehouse.
- The Data Profiling Task is located in the SSIS Toolbox, but you probably shouldn't attempt to use the results to make an automated workflow decision in the SSIS package Control Flow.
- The profiler can only report on statistics in the data; you still need to make judgments about these statistics. For example, a column may contain an overwhelming amount of NULL values, but the profiler doesn't know whether this reflects a valid business scenario.


Data Profile Viewer-C:\SSISPro\Profile.xml
Open
Refresh
Profiles (Table View)
Data Sources
localhost
Databases
AdventureWorksDW
Tables
[dbo].[DimCustomer]
Candidate Key Profiles
Column Length Distribution Profiles
Column Null Ratio Profiles
Column Statistics Profiles
Column Value Distribution Profiles
Column Null Ratio Profiles - [dbo].[DimCustomer]
1000 Rows
Message
Column Null Ratio Profiles

Column	Null Count	Null Percentage
MiddleName	7830	
NameStyle	0	
NumberCarsOwned	0	
NumberChildrenAtHome	0	
Phone	0	
SpanishEducation	0	
SpanishOccupation	0	

Column Null Ratio Profiles

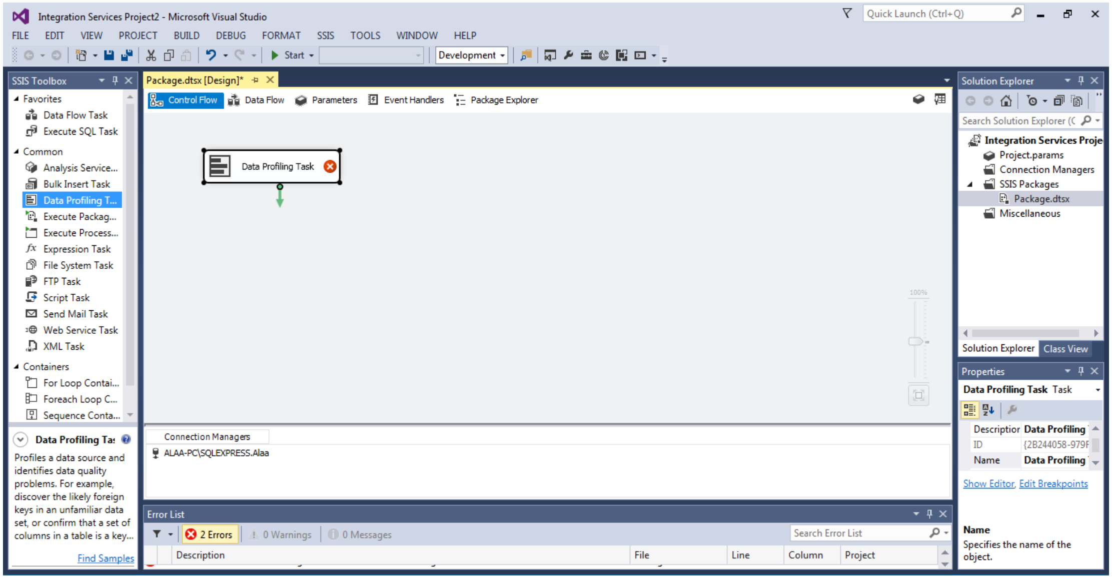
AddressLine1	AddressLine2	BirthDate	CommuteDistance	CustomerAlternate
5844 Linden Land		8/12/1965	2-5 Miles	AW00011002
1825 Village Pl.		2/15/1968	5-10 Miles	AW00011003

- Candidate Key Profile Request: The profile request will examine a column or set of columns to determine the likelihood of there being a unique candidate key for the data set. Use this to determine whether you have duplicate key values or whether it is possible to build a natural key with the data.
- Column Length Distribution Profile: This profile request enables you to analyze the statistical profile of all the data in a column, with the percentage of incidence for each length. You can use this to help you determine whether your data column length settings are set correctly or to look for bad data in attributes that are known to be one fixed size.
- Column Null Ratio Profile Request: This profile request looks at the ratio of NULL values in a column. Use this to determine whether you have a data quality problem in your source system for critical data elements.
- Column Pattern Profile Request: This profile request enables you to apply regular expressions to a string column to determine the pass/fail ratio across all the rows. Use this to evaluate business data using business formatting rules.
- Column Statistics Profile Request: This profile request can analyze all the rows and provide statistical information about the unique values

across the entire source. This can help you find low incidence values that may indicate bad data. For example, a finding of only one color type in a set of 1 million rows may indicate that you have a bad color attribute value.

- Functional Dependency Profile Request: This is one of two profile requests that enable you to examine relationships between tables and columns to look for discrepancies within a known dependency. For example, you can use this request to find countries with incorrect currency codes.
- Value Inclusion Profile Request: This profile request tests to determine whether the values in one column are all included in a separate lookup or dimension table. Use this to test foreign key relationships.

- To make data profiling for stored tables on our SQL Server:



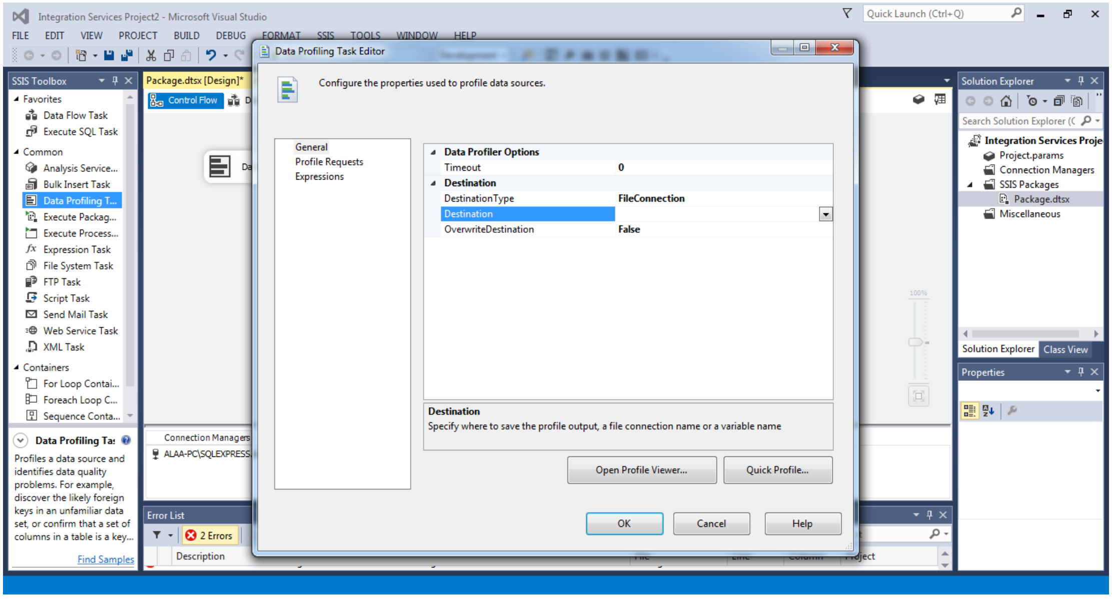
Profiles a data source and identifies data quality problems. For example, discover the likely foreign keys in an unfamiliar data set, or confirm that a set of columns in a table is a key...

ALAA-PC\SQLEXPRESS.Alaa

Descriptior Data Profiling
ID {2B244058-979F
Name Data Profiling

Name
Specifies the name of the object.

Double click on data profiling:



Data Profiling Task Editor

Configure the properties used to profile data sources.

General
Profile Requests
Expressions

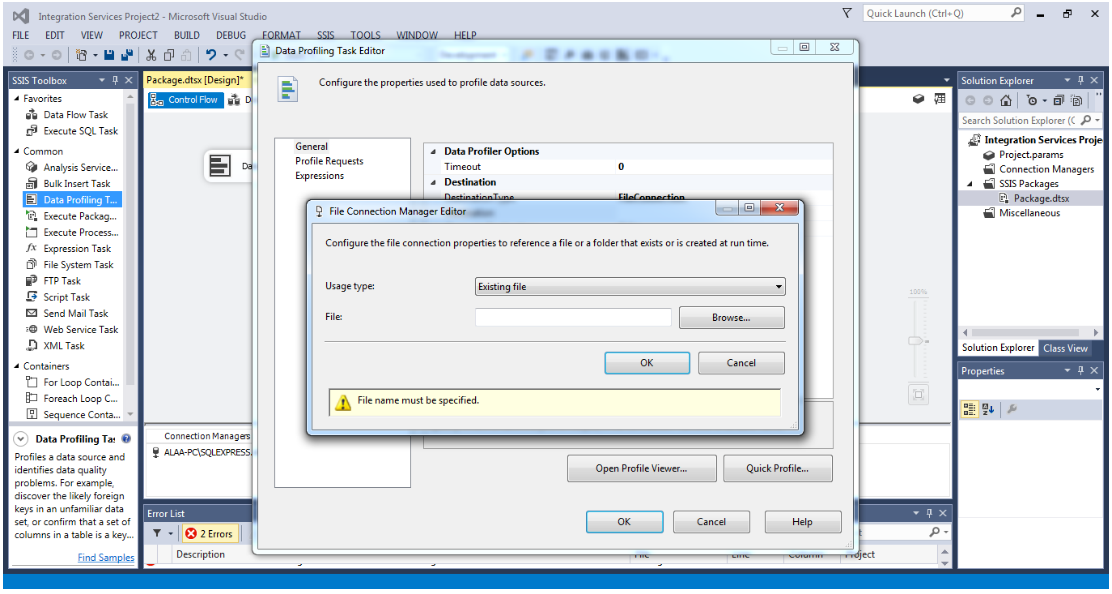
Data Profiler Options
- Timeout: 0

Destination
- DestinationType: FileConnection
- Destination:
- OverwriteDestination: False

Destination

Specify where to save the profile output, a file connection name or a variable name

Select Destination Type and Destination, in our case we will select Destination Type as File Connection and making the destination file to store the result:



File Connection Manager Editor

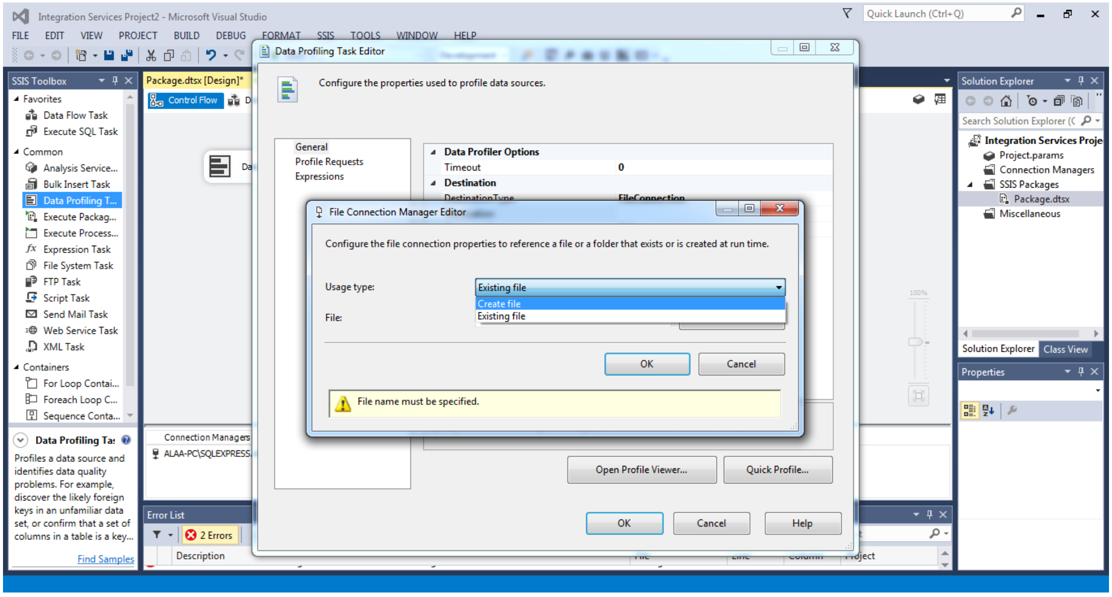
Configure the file connection properties to reference a file or a folder that exists or is created at run time.

Usage type: Existing file

File:

File name must be specified.

If we select Existing file from Usage type, we should select a pre-created file, if we select create file, the prompt will help you to create the file:



Configure the properties used to profile data sources.

File Connection Manager Editor

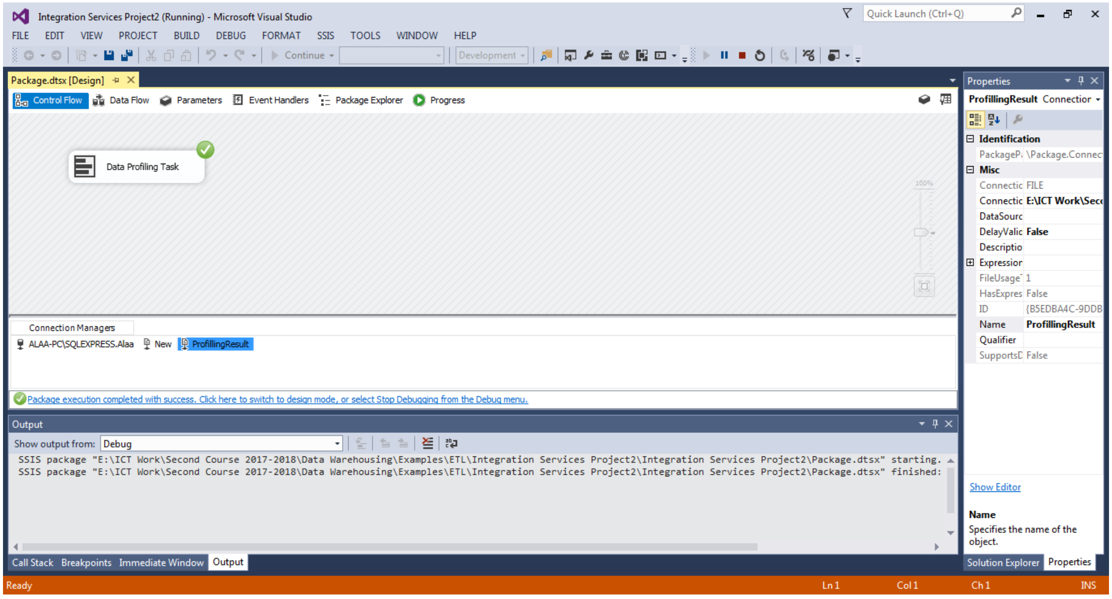
Configure the file connection properties to reference a file or a folder that exists or is created at run time.

Usage type: Existing file

File:

File name must be specified.

Select the file or create it on any place and click OK then execute the package.

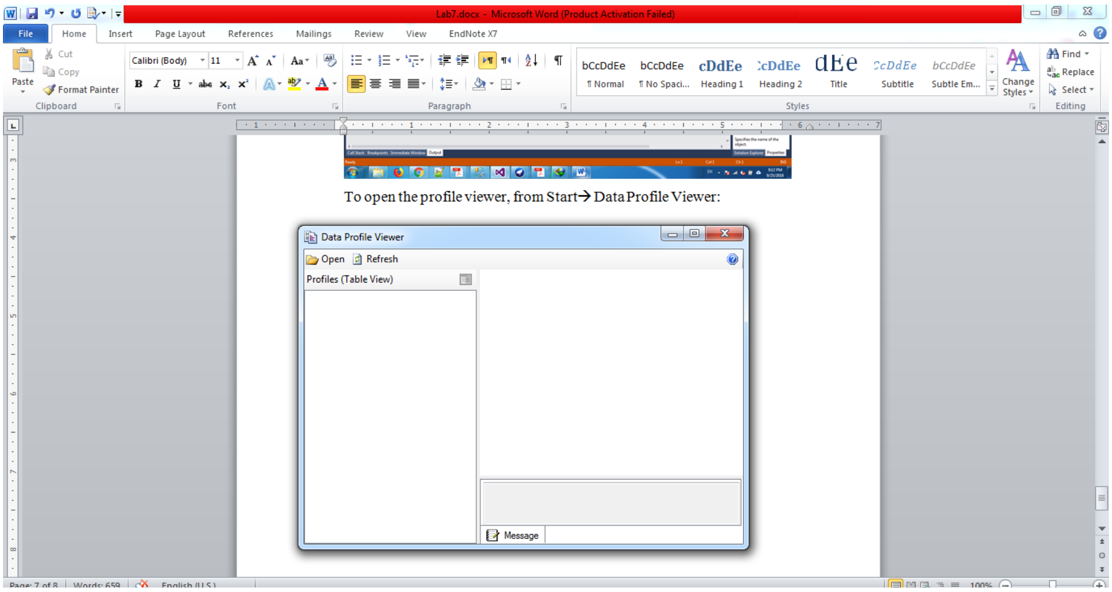


Package execution completed with success. Click here to switch to design mode, or select Stop Debugging from the Debug menu.

Show output from: Debug

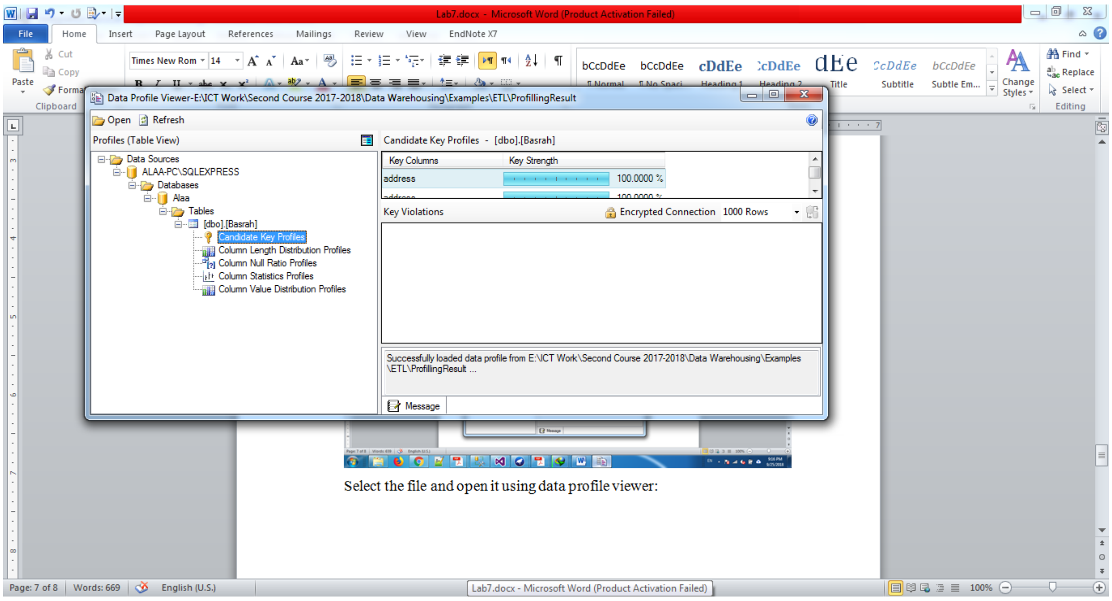
```
SSIS package "E:\ICT Work\Second Course 2017-2018\Data Warehousing\Examples\ETL\Integration Services Project2\Integration Services Project2\Package.dtsx" starting.
SSIS package "E:\ICT Work\Second Course 2017-2018\Data Warehousing\Examples\ETL\Integration Services Project2\Integration Services Project2\Package.dtsx" finished:
```

To open the profile viewer, from Start→ Data Profile Viewer:



To open the profile viewer, from Start→ Data Profile Viewer:

Select the file and open it using data profile viewer:



Data Profile Viewer-E:\ICT Work\Second Course 2017-2018\Data Warehousing\Examples\ETL\ProfilingResult

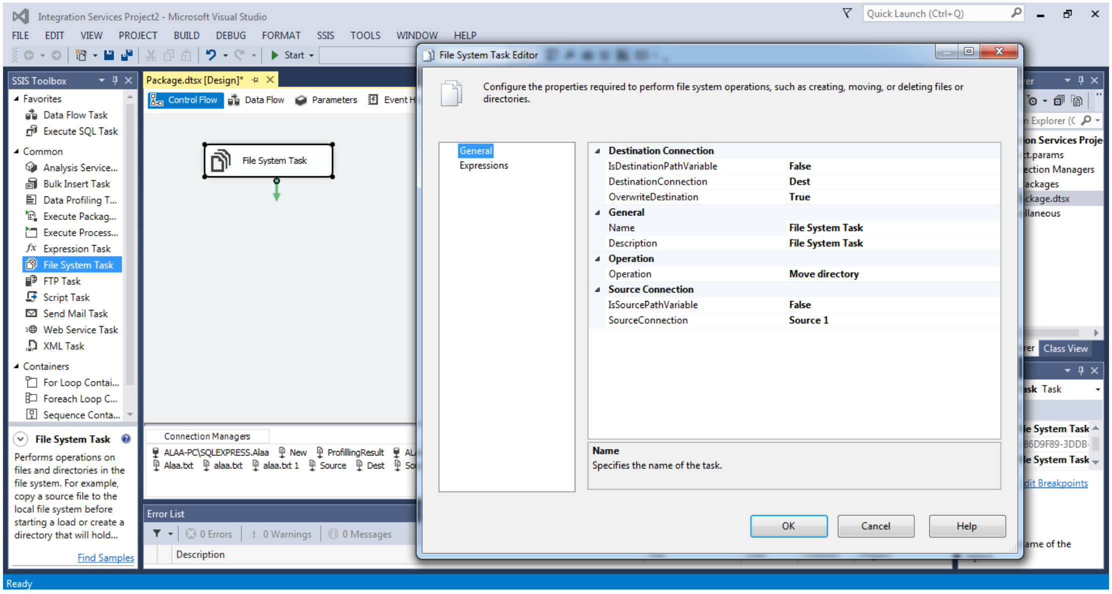
Successfully loaded data profile from E:\ICT Work\Second Course 2017-2018\Data Warehousing\Examples\ETL\ProfilingResult ...

Select the file and open it using data profile viewer:

2. File System Task

- The File System Task is a configurable GUI component that performs file operations available in the System.IO.File .NET class. If you are used to coding in VBScript, this is an out-of-the-box replacement for the VBScript utility classes that you used to write using the COM-based FileSystemObject.
- In either case, the File System Task can perform basic file operations such as the following:

  - Copy Directory: Copies all files from one directory to another. You must provide the source and destination directories.
  - Copy File: Copies a specific file. You must provide the source and destination filename.
  - Create Directory: Creates a directory. You must provide the source directory name and indicate whether the task should fail if the destination directory already exists.
  - Delete Directory: Deletes a directory. You must provide the source directory to delete.
  - Delete Directory Content: Deletes all files in a source directory
  - Delete File: Deletes a specifically provided source file
  - Move Directory: Moves a provided source directory to a destination directory. You must indicate whether the task should fail if the destination directory already exists.
  - Move File: Moves a specific provided source file to a destination. You must indicate whether the task should fail if the destination file already exists.
  - Rename File: Moves a specific provided source file to a destination by changing the name. You must indicate whether the task should fail if the destination file already exists.
  - Set Attributes: Sets Hidden, Read-Only, Archive, or System attributes on a provided source file.
- To move directory content to another directory, select file system task and drop it in data flow task and configure it:



Configure the properties required to perform file system operations, such as creating, moving, or deleting files or directories.

Destination Connection	
IsDestinationPathVariable	**False**
DestinationConnection	**Dest**
OverwriteDestination	**True**
General	
Name	**File System Task**
Description	**File System Task**
Operation	
Operation	**Move directory**
Source Connection	
IsSourcePathVariable	**False**
SourceConnection	**Source 1**

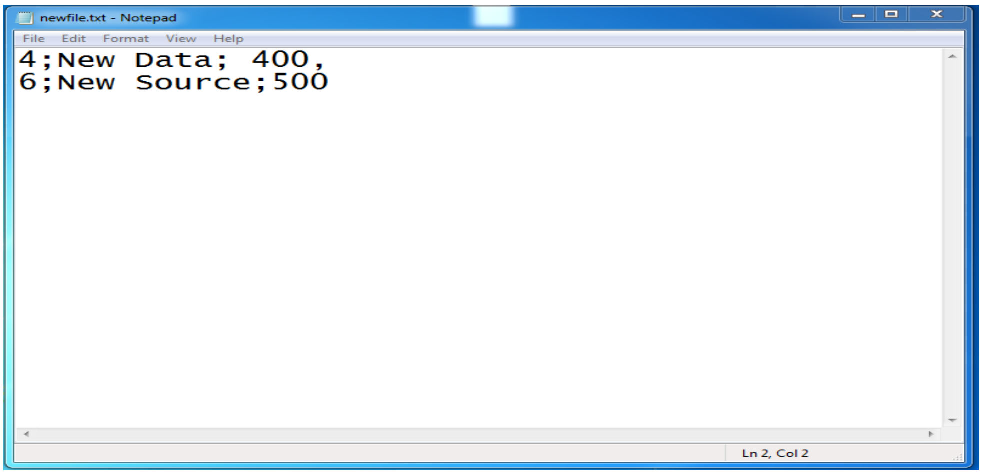
Name
Specifies the name of the task.

- Select the source connection to the directory which you want to move it's content, select the destination to select the destination connection and set the folder and click OK, then execute it.

- Example: move a file to archive directory.

3. Bulk Insert Task

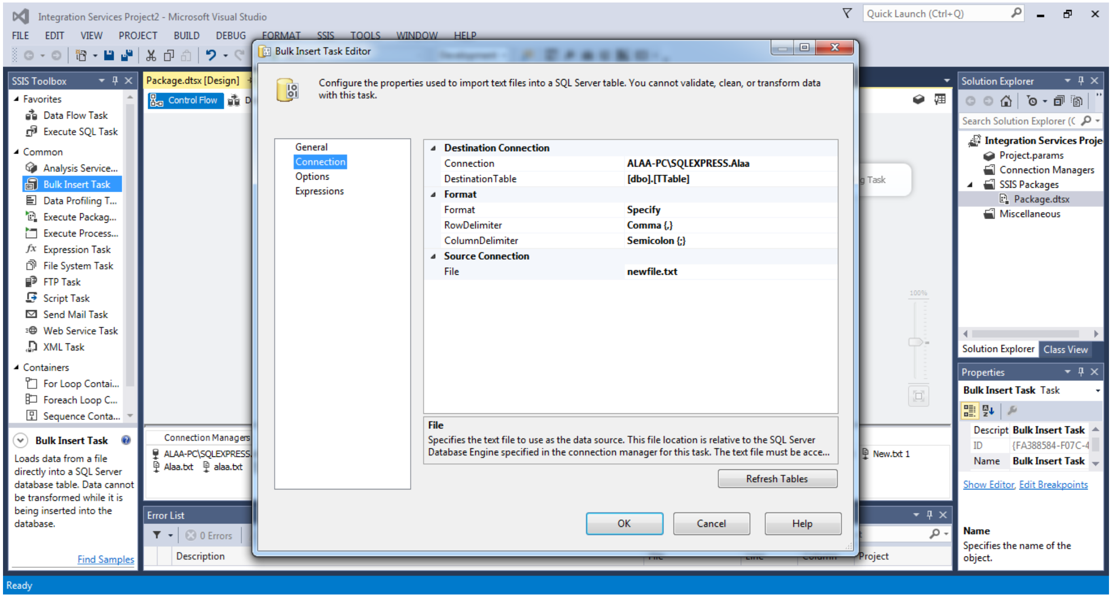
- The Bulk Insert Task enables you to insert data from a text or flat file into a SQL Server database table in the same high-octane manner as using a BULK INSERT statement or the bcp.exe command line tool.
- In fact, the task is basically just a wizard to store the information needed to create and execute a bulk copying command at runtime (similar to BCP from a command line). If you aren't familiar with using BCP, you can research the topic in detail in Books Online. The downside of the Bulk Insert Task is its strict data format, and it precludes being able to work with data in a Data Flow within one action.
- To insert data from file to existing table, first the file should be formatted to structure that can be configured by Bulk Insert Task: suppose you have the following file with content:



newfile.txt - Notepad

```
4;New Data; 400,
6;New Source;500
```

- The file content should be configured depending on the destination table, so based on the table the first column should be integer data type, the second on should be string data type and the third column should be integer data type in order to load the file content to the table with (int, string, int) data types.
- Drag and drop Bulk Insert Task and configure the source and destination table and click OK.



Bulk Insert Task Editor

Configure the properties used to import text files into a SQL Server table. You cannot validate, clean, or transform data with this task.

Destination Connection
- Connection: ALAA-PC\SQLEXPRESS.Alaa
- DestinationTable: [dbo].[TTable]

Format
- Format: Specify
- RowDelimiter: Comma {,}
- ColumnDelimiter: Semicolon {;}

Source Connection
- File: newfile.txt

File
Specifies the text file to use as the data source. This file location is relative to the SQL Server Database Engine specified in the connection manager for this task. The text file must be acce...

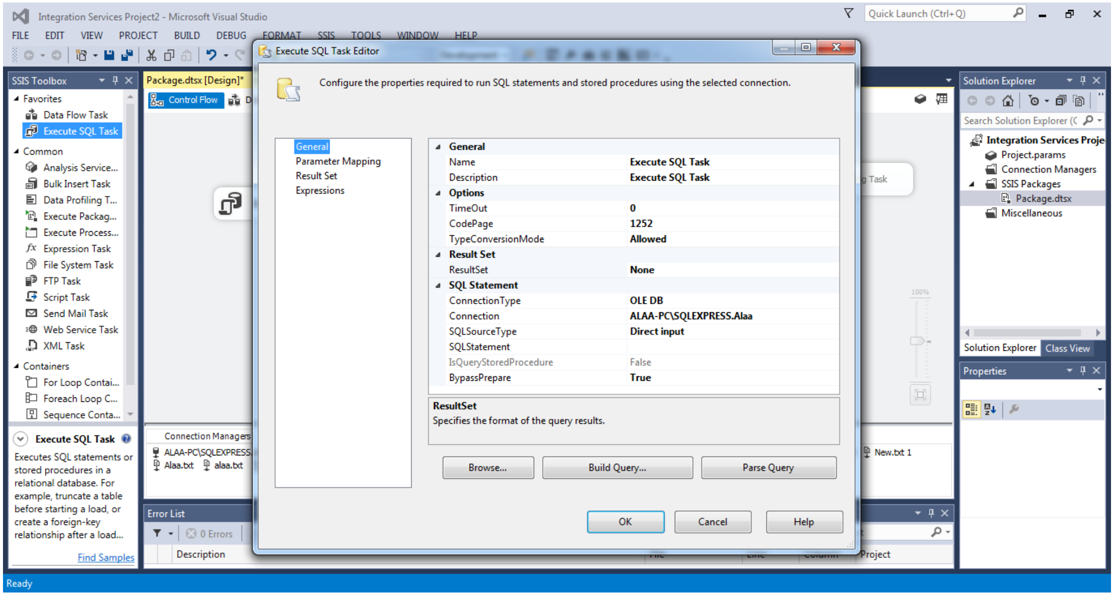
Bulk Insert Task
Loads data from a file directly into a SQL Server database table. Data cannot be transformed while it is being inserted into the database.

4. Execute SQL Task

- The Execute SQL Task is one of the most widely used tasks in SSIS for interacting with an RDBMS Data Source.
- The Execute SQL Task is used for all sorts of things, including truncating a staging data table prior to importing, retrieving row counts to determine the

next step in a workflow, or calling stored procedures to perform business logic against sets of staged data. This task is also used to retrieve information from a database repository.

- The Execute SQL Task is also found in the legacy DTS product, but the SSIS version provides a better configuration editor and methods to map stored procedure parameters to read back result and output values.
- This section introduces you to all the possible ways to configure this task by working through the different ways you can use it. You'll work through how to execute parameterized SQL statements or execute batches of SQL statements, how to capture single-row and multiple-row results, and how to execute stored procedures.
- To execute stored procedure using Execute SQL Task, drop the tool and configure the SQL code:



Execute SQL Task Editor

Configure the properties required to run SQL statements and stored procedures using the selected connection.

General	
Name	Execute SQL Task
Description	Execute SQL Task
Options	
TimeOut	0
CodePage	1252
TypeConversionMode	Allowed
Result Set	
ResultSet	None
SQL Statement	
ConnectionType	OLE DB
Connection	ALAA-PC\SQLEXPRESS.Alaa
SQLSourceType	Direct input
SQLStatement	
IsQueryStoredProcedure	False
BypassPrepare	True

ResultSet
Specifies the format of the query results.

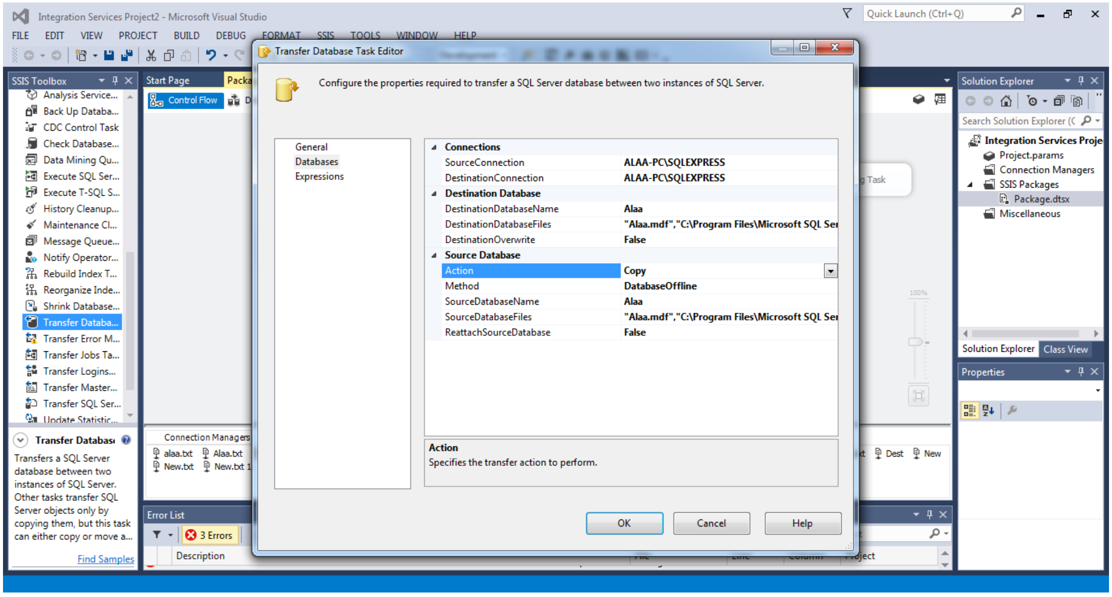
Execute SQL Task: Executes SQL statements or stored procedures in a relational database. For example, truncate a table before starting a load, or create a foreign-key relationship after a load...

- Set the connection to the database from Connection, select the SQL statement to 'EXEC sp_dbremove 'Test' to remove 'Test' database.

5. Transfer Database Task

- The Transfer Database Task has, as you would expect, a source and destination connection and a database property. The other properties address how the transfer should take place. Figure 3-37 is an example of the Transfer Database Task filled out to copy a development database on the same server as a QA instance.

- Notice that the destination and source are set to the same server. For this copy to work, the DestinationDatabaseFiles property has to be set to new mdf and ldf filenames. The property is set by default to the SourceDatabaseFiles property. To set the new destination database filenames, click the ellipsis, and then change the Destination File or Destination Folder properties.
- You can set the Method property to DatabaseOnline or DatabaseOffline. If the option is set to DatabaseOffline, the database is detached copied over and then reattached to both systems. This is a much faster process than with DatabaseOnline, but it comes at a cost of making the database inaccessible.
- The Action property controls whether the task should copy or move the source database. The Method property controls whether the database should be copied while the source database is kept online, using SQL Server Management Objects (SMO), or by detaching the database, moving the files, and then reattaching the database. The DestinationOverwrite property controls whether the creation of the destination database should be allowed to overwrite.
- This includes deleting the database in the destination if it is found. This is useful in cases where you want to copy a database from production into a quality-control or production test environment, and the new database should replace any existing similar database. The last property is the ReattachSourceDatabase, which specifies what action should be taken upon failure of the copy. Use this property if you have a package running on a schedule that takes a production database offline to copy it, and you need to guarantee that the database goes back online even if the copy fails.



Transfers a SQL Server database between two instances of SQL Server. Other tasks transfer SQL Server objects only by copying them, but this task can either copy or move a...

Transfer Database Task Editor

Configure the properties required to transfer a SQL Server database between two instances of SQL Server.

General
Databases
Expressions

Connections	
SourceConnection	**ALAA-PC\SQLEXPRESS**
DestinationConnection	**ALAA-PC\SQLEXPRESS**
Destination Database	
DestinationDatabaseName	**Alaa**
DestinationDatabaseFiles	**"Alaa.mdf","C:\Program Files\Microsoft SQL Se**
DestinationOverwrite	**False**
Source Database	
Action	**Copy**
Method	**DatabaseOffline**
SourceDatabaseName	**Alaa**
SourceDatabaseFiles	**"Alaa.mdf","C:\Program Files\Microsoft SQL Se**
ReattachSourceDatabase	**False**

Action
Specifies the transfer action to perform.

OK Cancel Help

6. Data Flow Task

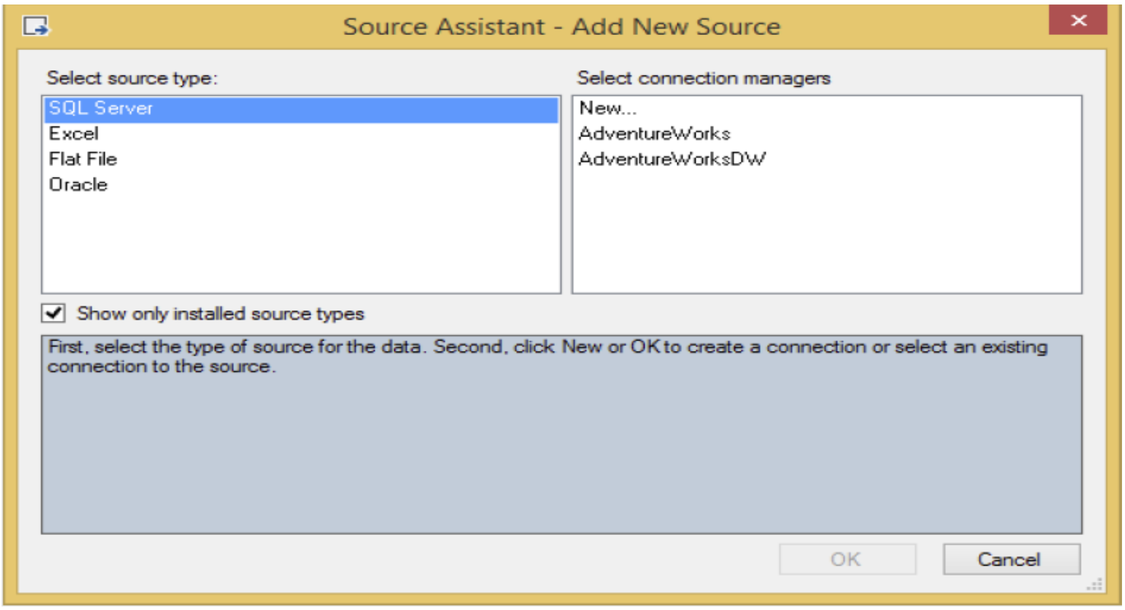
- The SSIS Data Flow is implemented as a logical pipeline, where data flows from one or more sources, through whatever transformations are needed to cleanse and reshape it for its new purpose, and into one or more destinations. The Data Flow does its work primarily in memory, which gives SSIS its strength, allowing the Data Flow to perform faster than any ELT packages.
- One of the toughest concepts to understand for a new SSIS developer is the difference between the Control Flow and the Data Flow tabs.
- The Control Flow tab controls the workflow of the package and the order in which each task will execute. Each task in the Control Flow has a user interface to configure the task, with the exception of the Data Flow Task.
- The Data Flow Task is configured in the Data Flow tab. Once you drag a Data Flow Task onto the Control Flow tab and double-click it to configure it, you're immediately taken to the Data Flow tab.
- Data viewers are a very important feature in SSIS for debugging your Data Flow pipeline. They enable you to view data at points in time at runtime. If you place a data viewer before and after the Aggregate Transformation, for example, you

can see the data flowing into the transformation at runtime and what it looks like after the transformation happens.

- To place a data viewer in your pipeline, right-click one of the paths (red or blue arrows leaving a transformation or source) and select Enable Data Viewer.

7. Source

- ○ A source in the SSIS Data Flow is where you specify the location of your source data. Most sources will point to a Connection Manager in SSIS. By pointing to a Connection Manager, you can reuse connections throughout your package, because you need only change the connection in one place.
- ○ The Source Assistant and Destination Assistant are two components designed to remove the complexity of configuring a source or a destination in the Data Flow.



Source Assistant - Add New Source

Select source type:

- SQL Server
- Excel
- Flat File
- Oracle

Select connection managers

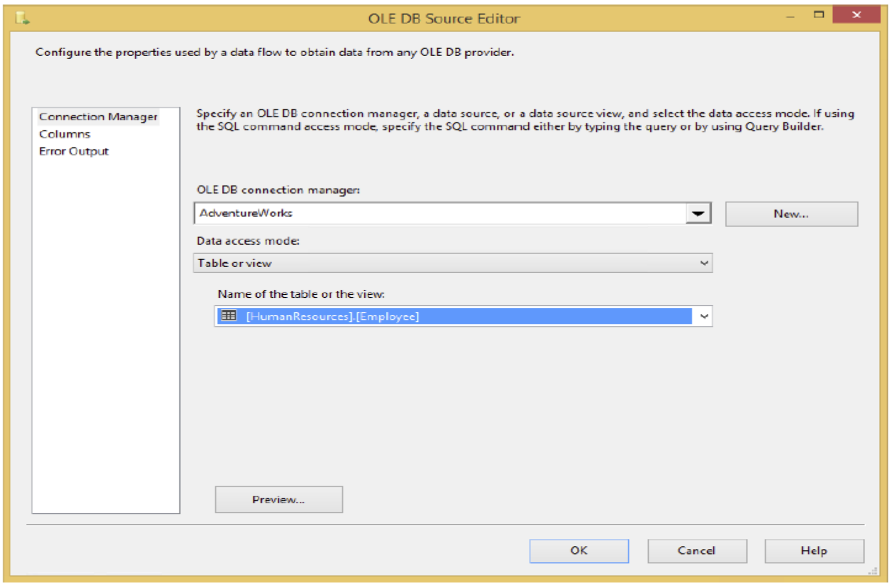
- New...
- AdventureWorks
- AdventureWorksDW

☑ Show only installed source types

First, select the type of source for the data. Second, click New or OK to create a connection or select an existing connection to the source.

- ○ OLE DB Source
- ○ The OLE DB Source is the most common type of source, and it can point to any OLE DB–compliant Data Source such as SQL Server, Oracle, or DB2. To configure the OLE DB Source, double-click the source once you have added it to the design pane in the Data Flow tab. In the Connection Manager page of the OLE DB Source Editor, select the Connection Manager of your

OLE DB Source from the OLE DB Connection Manager dropdown box. You can also add a new Connection Manager in the editor by clicking the New button.



OLE DB Source Editor

Configure the properties used by a data flow to obtain data from any OLE DB provider.

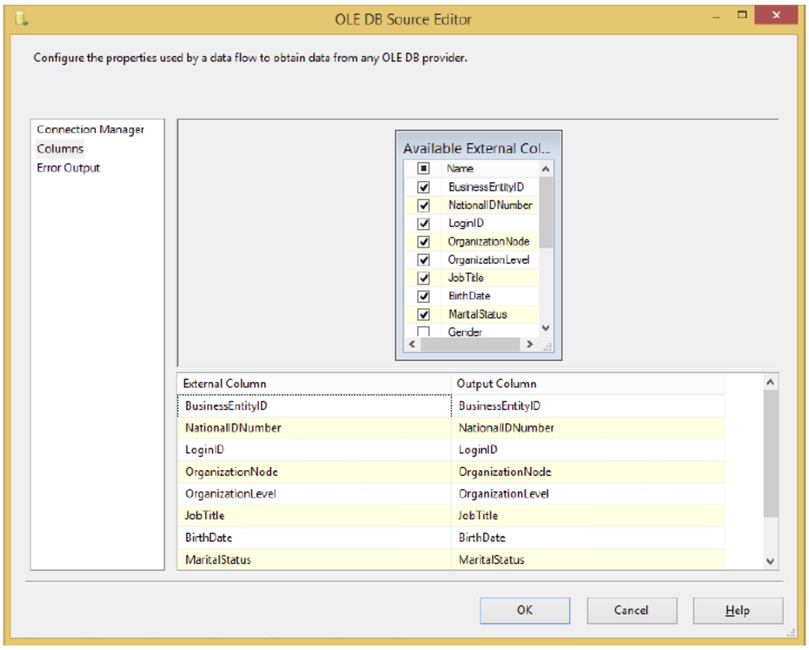
Specify an OLE DB connection manager, a data source, or a data source view, and select the data access mode. If using the SQL command access mode, specify the SQL command either by typing the query or by using Query Builder.

OLE DB connection manager: AdventureWorks

Data access mode: Table or view

Name of the table or the view: [HumanResources].[Employee]

- As with most sources, you can go to the Columns page to set columns that you wish to output to the Data Flow, as shown below. Simply check the columns you wish to output, and you can then assign the name you want to send down the Data Flow in the Output column. Select only the columns that you want to use, because the smaller the data set, the better the performance you will get.



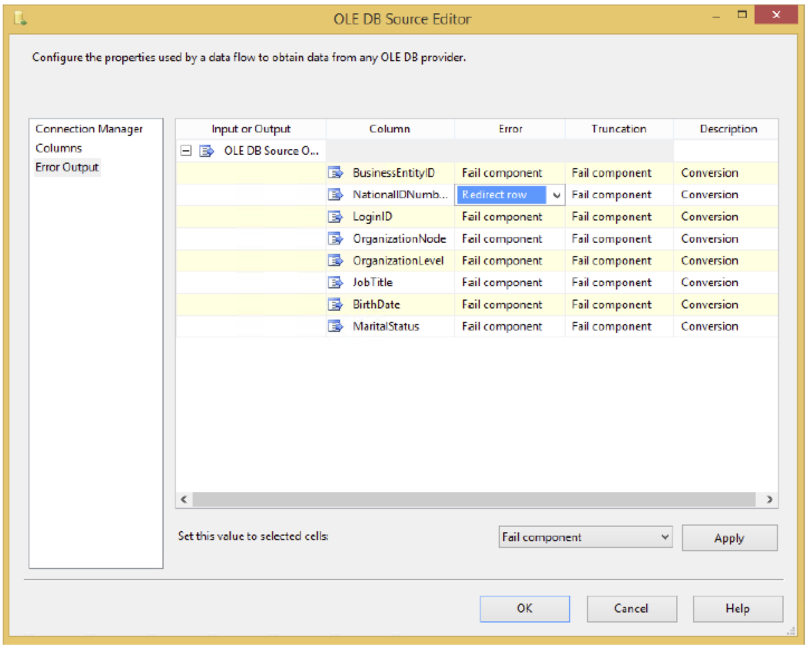
OLE DB Source Editor

Configure the properties used by a data flow to obtain data from any OLE DB provider.

■	Name
☑	BusinessEntityID
☑	NationalIDNumber
☑	LoginID
☑	OrganizationNode
☑	OrganizationLevel
☑	JobTitle
☑	BirthDate
☑	MaritalStatus
□	Gender

External Column	Output Column
BusinessEntityID	BusinessEntityID
NationalIDNumber	NationalIDNumber
LoginID	LoginID
OrganizationNode	OrganizationNode
OrganizationLevel	OrganizationLevel
JobTitle	JobTitle
BirthDate	BirthDate
MaritalStatus	MaritalStatus

- Optionally, you can go to the Error Output page (shown in Figure 4-4) and specify how you wish to handle rows that have errors. For example, you may wish to output any rows that have a data type conversion issue to a different path in the Data Flow. On each column, you can specify that if an error occurs, you wish the row to be ignored, be redirected, or fail. If you choose to ignore failures, the column for that row will be set to NULL. If you redirect the row, the row will be sent down the red path in the Data Flow coming out of the OLE DB Source.



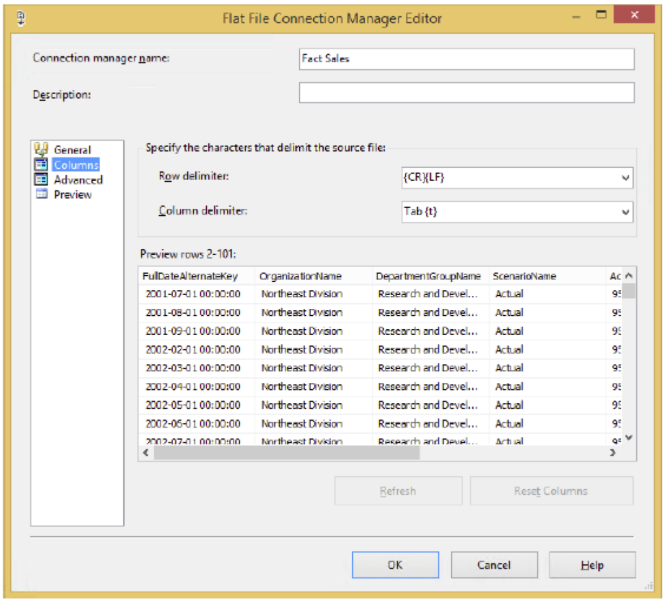
Configure the properties used by a data flow to obtain data from any OLE DB provider.

Input or Output	Column	Error	Truncation	Description
OLE DB Source O...				
	BusinessEntityID	Fail component	Fail component	Conversion
	NationalIDNumb...	Redirect row	Fail component	Conversion
	LoginID	Fail component	Fail component	Conversion
	OrganizationNode	Fail component	Fail component	Conversion
	OrganizationLevel	Fail component	Fail component	Conversion
	JobTitle	Fail component	Fail component	Conversion
	BirthDate	Fail component	Fail component	Conversion
	MaritalStatus	Fail component	Fail component	Conversion

Set this value to selected cells: Fail component

- Excel Source
- The Excel Source is a source component that points to an Excel spreadsheet, just like it sounds. Once you point to an Excel Connection Manager, you can select the sheet from the “Name of the Excel sheet” dropdown box, or you can run a query by changing the Data Access Mode. This source treats Excel just like a database, where an Excel sheet is the table and the workbook is the database. If you do not see a list of sheets in the dropdown box, you may have a 64-bit machine that needs the ACE driver installed or you need to run the package in 32-bit mode.

- Flat File Source
- The Flat File Source provides a data source for connections such as text files or data that's delimited. Flat File Sources are typically comma- or tab-delimited files, or they could be fixed-width or ragged right.
- A fixed-width file is typically received from the mainframe or government entities and has fixed start and stop points for each column. This method enables a fast load, but it takes longer at design time for the developer to map each column. You specify a Flat File Source the same way you specify an OLE DB Source. Once you add it to your Data Flow pane, you point it to a Connection Manager connection that is a flat file or a multi-flat file. Next, from the Columns tab, you specify which columns you want to be presented to the Data Flow. All the specifications for the flat file, such as delimiter type, were previously set in the Flat File Connection Manager.



Connection manager name: Fact Sales

Description:

Specify the characters that delimit the source file:

Row delimiter: {CR}{LF}

Column delimiter: Tab {t}

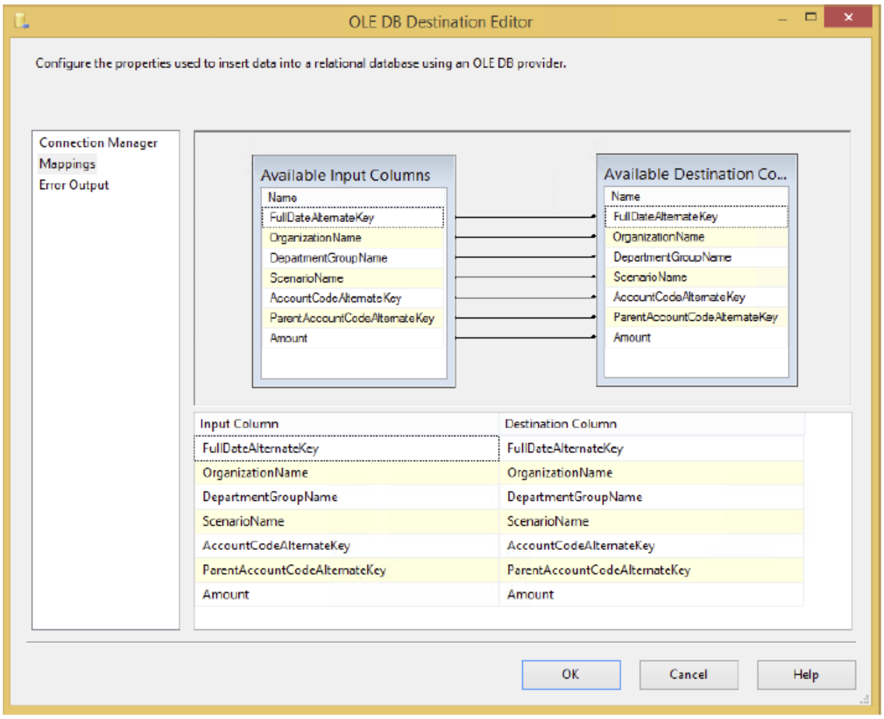
Preview rows 2-101:

FullDateAlternateKey	OrganizationName	DepartmentGroupName	ScenarioName	Ac
2001-07-01 00:00:00	Northeast Division	Research and Devel...	Actual	9
2001-08-01 00:00:00	Northeast Division	Research and Devel...	Actual	9
2001-09-01 00:00:00	Northeast Division	Research and Devel...	Actual	9
2002-02-01 00:00:00	Northeast Division	Research and Devel...	Actual	9
2002-03-01 00:00:00	Northeast Division	Research and Devel...	Actual	9
2002-04-01 00:00:00	Northeast Division	Research and Devel...	Actual	9
2002-05-01 00:00:00	Northeast Division	Research and Devel...	Actual	9
2002-06-01 00:00:00	Northeast Division	Research and Devel...	Actual	9
2002-07-01 00:00:00	Northeast Division	Research and Devel...	Actual	9

8. Destination

- Inside the Data Flow, destinations accept the data from the Data Sources and from the transformations. The architecture can send the data to nearly any OLE DB–compliant Data Source, a flat file, or Analysis Services, to name just a few. Like sources, destinations are managed through Connection

Managers. The configuration difference between sources and destinations is that in destinations, you have a Mappings page (shown in Figure below), where you specify how the inputted data from the Data Flow maps to the destination. As shown in the Mappings page in this figure, the columns are automatically mapped based on column names, but they don't necessarily have to be exactly lined up. You can also choose to ignore given columns, such as when you're inserting into a table that has an identity column, and you don't want to inherit the value from the source table.



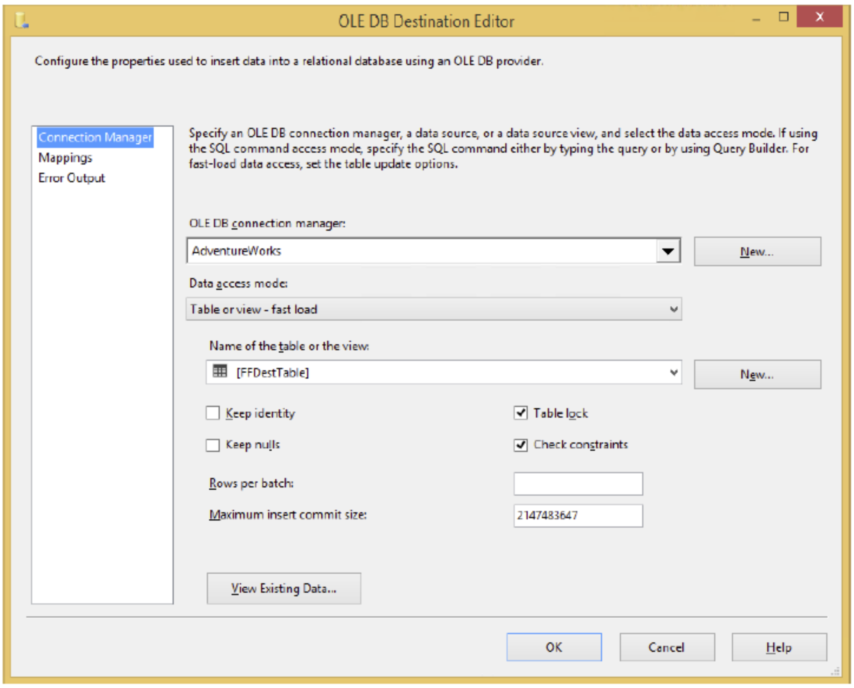
Configure the properties used to insert data into a relational database using an OLE DB provider.

Input Column	Destination Column
FullDateAlternateKey	FullDateAlternateKey
OrganizationName	OrganizationName
DepartmentGroupName	DepartmentGroupName
ScenarioName	ScenarioName
AccountCodeAlternateKey	AccountCodeAlternateKey
ParentAccountCodeAlternateKey	ParentAccountCodeAlternateKey
Amount	Amount

- Excel Destination
  - The Excel Destination is identical to the Excel Source except that it accepts data rather than sends data. To use it, first select the Excel Connection Manager from the Connection Manager page, and then specify the worksheet into which you wish to load data.
- Flat File Destination
  - The commonly used Flat File Destination sends data to a flat file, and it can be fixed-width or delimited based on the Connection Manager. The destination uses a Flat File Connection Manager.

- You can also add a custom header to the file by typing it into the Header option in the Connection Manager page. Lastly, you can specify on this page that the destination file should be overwritten each time the Data Flow is run.

- OLE DB Destination
  - Your most commonly used destination will probably be the OLE DB Destination (see Figure below).
  - It can write data from the source or transformation to OLE DB–compliant Data Sources such as Oracle, DB2, Access, and SQL Server. It is configured like any other destination and source, using OLE DB Connection Managers. A dynamic option it has is the Data Access Mode.
  - If you select Table or View - Fast Load, or its variable equivalent, several options will be available, such as Table Lock. This Fast Load option is available only for SQL Server database instances and turns on a bulk load option in SQL Server instead of a row-by-row operation.



OLE DB Destination Editor

Configure the properties used to insert data into a relational database using an OLE DB provider.

Specify an OLE DB connection manager, a data source, or a data source view, and select the data access mode. If using the SQL command access mode, specify the SQL command either by typing the query or by using Query Builder. For fast-load data access, set the table update options.

OLE DB connection manager: AdventureWorks

Data access mode: Table or view - fast load

Name of the table or the view: [FFDestTable]

□ Keep identity ☑ Table lock

□ Keep nulls ☑ Check constraints

Rows per batch:

Maximum insert commit size: 2147483647

- Raw File Destination
  - The Raw File Destination is an especially speedy Data Destination that does not use a Connection Manager to configure. Instead, you point to the file on the server in the editor. This destination is written to typically as an

intermediate point for partially transformed data. Once written to, other packages can read the data in by using the Raw File Source. The file is written in native format, so it is very fast.