

SQL Server Integration Services (SSIS)

Lesson 00: Training Kit(Part 1)

People matter, results count.



Document History

Date	Course Version No.	Software Version No.	Developer / SME	Reviewer(s)	Approver	Change Record Remarks
01-Feb-12	1.0	1.0	Ajit jog			Initial Document
June 2016	2.0	NA	Sarita Kaloya	Pramodh Daniel	Mahima Sharma	Material Revamp as per Integrated ToC for I & D LoT

Copyright © Capgemini 2016. All Rights Reserved.

Course Audience and Course Goals

- This course is designed for
 - Analysts, Designers and Developers involved in Design, Development, and Maintenance of Data Warehousing Applications
- Course Goals
 - To understand SSIS as an ETL tool

Copyright © Capgemini 2010. All Rights Reserved.

Pre-requisites

- Good knowledge of RDBMS
- Fair knowledge of Data Warehousing Concepts



Copyright © Capgemini 2014. All Rights Reserved.

Intended Audience

- Professional Data warehousing experts, or anyone requiring a foundation knowledge in ETL process



Day Wise Schedule

■ Day 1

- Lesson 1: Introduction to ETL and Integration Services
- Lesson 2: SSIS Background
- Lesson 3: Typical Uses of Integration Services
- Lesson 4: Tools

■ Day 2

- Lesson 5: SSIS Objects
- Lesson 6: Data Flow Task
- Lesson 7: Data Flow Task -Source

■ Day 3

- Lesson 8: Data Flow Tasks-Transformation
- Lesson 9: Data Flow Tasks-Destination

Copyright © Capgemini 2010. All Rights Reserved

Day Wise Schedule

■ Day 4

- Lesson 10: Operating system-level tasks
- Lesson 11: SQL Server Tasks
- Lesson 12: Scripting Tasks

■ Day 5

- Lesson 13: Profiling Task
- Lesson 14: Workflow Tasks
- Lesson 15: Processing XML
- Lesson 16: Analysis Services Tasks
- Lesson 17: Maintenance Tasks

■ Day 6

- Lesson 18: Container
- Lesson 19: Precedence Constraint
- Lesson 20: Variables and Expressions

Copyright © Capgemini 2010. All Rights Reserved

Day Wise Schedule

■ Day 7

- Lesson 21: SSIS Architecture
- Lesson 22: Error Handling, Logging and Transactions
- Lesson 23: Execution , debugging and Protection

■ Day 8

- Lesson 24: Deployment
- Lesson 25: Performance Tuning
- Lesson 26: Best Practices

Copyright © Capgemini 2010. All Rights Reserved.

Course Content

- Lesson 1:Introduction to ETL and Integration Services
 - Overview of BI
 - ETL Concept
 - Different ETL Tool providers
 - Defining SQL Server Integration Services (SSIS)
- Lesson 2: SSIS Background
 - DTS Limitation
 - Difference between DTS , 2005, 2008 R2 ,2012,2014 and 2016
 - Lesson 3: Typical Uses of Integration Services
 - Merging Data from Heterogeneous Data Stores
 - Populating Data Warehouses and Data Marts Cleaning and Standardizing Data
 - Automating Administrative Functions and Data Loading
- Lesson 4: Tools
 - Business Intelligence Development Studio and SQL Server data Tool
 - SQL Server Management Studio
 - Import and Export Wizard
 - Package Configuration Wizard
 - Package Deployment Wizard

Copyright © Capgemini 2016. All Rights Reserved

Course Content

- Lesson 5: SSIS Objects
 - Package
 - Control Flow
 - Data Flow
 - Connection Manager
- Lesson 6: Data Flow Task
 - Data Flow Task
- Lesson 7: Data Flow Task -Source
 - ADO.NET Source
 - Excel Source
 - Flat File Source
 - OLE DB Source
 - Raw File Source
 - Script Component
 - XML Source
- Lesson 8: Data Flow Tasks-Transformation
 - Aggregate
 - Audit
 - Cache Transform
 - Character Map

Copyright © Capgemini 2010. All Rights Reserved.

Course Content

- Conditional Split
- Copy Column
- Data Conversion
- Data Mining Query
- Derived Column
- Export Column
- Fuzzy Grouping
- Fuzzy Lookup
- Import Column
- Lookup
- Merge
- Merge Join
- Multicast
- OLE DB Command
- Percentage Sampling
- Pivot
- Row Count
- Row Sampling
- Script Component
- Slowly Changing Dimension

Copyright © Capgemini 2010. All Rights Reserved.11

Course Content

- Sort
- Term Extraction
- Term Lookup
- Union All
- Unpivot

■ Lesson 9: Data Flow Tasks-Destination

- ADO NET Destination
- Data Mining Model Training Destination
- DataReader Destination
- Dimension Processing Destination
- Excel Destination
- Flat File Destination
- OLE DB Destination
- Partition Processing Destination
- Raw File Destination
- Recordset Destination
- Script Component
- SQL Server Compact Edition Destination
- SQL Server Destination

Copyright © Capgemini 2010. All Rights Reserved. 12

Course Content

- Lesson 10: Operating system-level tasks
 - File System Task
 - FTP Task
 - Send Mail Task
- Lesson 11: SQL Server Tasks
 - Bulk Insert Task
 - Execute SQL Task
 - Transfer Database Task
 - Transfer Error Messages Task
 - Transfer Jobs Task
 - Transfer Logins Task
 - Transfer Master Stored Procedures Task
 - Transfer SQL Server Objects Task
- Lesson 12: Scripting Tasks
 - Script Task
- Lesson 13: Profiling Task
 - Data Profiling Task
 - Lesson 14: Workflow Tasks
 - Execute Package Task
 - Execute Process Task
 - Message Queue Task

Copyright © Capgemini 2010. All Rights Reserved. 11

Course Content

- Lesson 15: Processing XML
 - XML Task
- Lesson 16: Analysis Services Tasks
 - Analysis Services Processing Task
 - Analysis Services Execute DDL Task
 - Data Mining Query Task
- Lesson 17: Maintenance Tasks
 - Back Up Database Task
 - Check Database Integrity Task
 - Execute SQL Server Agent Job Task
 - Execute T-SQL Statement Task
 - History Cleanup Task
 - Notify Operator Task
 - Rebuild Index Task
 - Reorganize Index Task
 - Shrink Database Task
 - Update Statistics Task
- Lesson 18: Container
 - For each loop
 - For loop
 - Sequence

Copyright © Capgemini 2010. All Rights Reserved. 14

Course Content

- **Lesson 19:** Precedence Constraint
 - Configuring the Precedence Constraint
- **Lesson 20:** Variables and Expressions
 - System Variables
 - User Defined Variables
 - Configuring Variables
- **Lesson 21:** SSIS Architecture
 - SSIS Architecture overview
 - Distinguishing between data flow pipeline and package runtime
 - Executing packages on the client side or hosted in the SSIS service
- **Lesson 22:** Error Handling, Logging and Transactions
 - Event Handler
 - SSIS Logging
 - Check point
- **Lesson 23:** Execution , debugging and Protection
 - Package execution
 - data Viewer
 - Breakpoints
 - Package Protection

Copyright © Capgemini 2010. All Rights Reserved. 11

Course Content

- Lesson 24: Deployment
 - Switching between project and package deployment modes
 - Deploying packages to the SSISDB
 - Running packages from SQL Server
 - Leveraging Project/package parameters
- Lesson 25: Performance Tuning
 - Designing SSIS Packages for High Performance
 - Monitoring the SSIS Performance with Performance Counters
 - Tuning Tips
 - Configure Rows per Batch and Maximum Insert Commit Size in OLEDB destination
- Lesson 26: Best Practices
 - Best Practices for SSIS

Copyright © Capgemini 2010. All Rights Reserved.

Introduction to ETL and Integration Services

Lesson 1

Overview of BI and Data Warehousing

Business Intelligence

- Business Intelligence covers different technologies for gathering, storing, analyzing and providing access to data.
- BI is a decision support because it simplifies information discovery and analysis, making .

Data Warehousing

- DW contains historical data derived from transaction data.
- Data warehouse is a relational database that is designed for query and analysis rather than for transaction processing
- DW includes ETL,OLAP client analysis tools, and other application



Copyright © Capgemini 2010. All Rights Reserved. 10

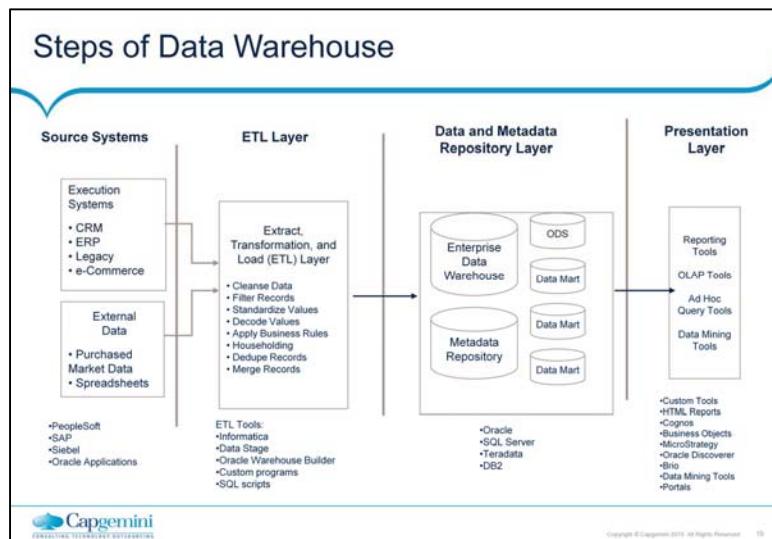
Business Intelligence is an area that covers a number of different technologies for gathering, storing, analyzing and providing access to data that will help an large company make better business decisions.

BI is a decision support system because it simplifies information discovery and analysis, making it possible for decision-makers at all levels of an organization to more easily access, understand, analyze, collaborate, and act on information

DW contains historical data derived from transaction data.

Data warehouse is a relational database that is designed for query and analysis rather than for transaction processing

Data warehouse environment includes an extraction, transportation, transformation, and loading (ETL) solution, an online analytical processing (OLAP) engine, client analysis tools, and other applications that manage the process of gathering data and delivering it to business users.



➤ **Source System:** From where data is coming to DW system

➤ **ETL Layer:** Clean ,Validate, Transform data

➤ **Data and Metadata Repository Layer:** Load data into table

➤ **Presentation Layer:** Report generation for Business user

ETL Concept

- ETL stands for extraction, transformation and loading. Etl is a process that involves the following tasks:

Extract

- **extracting data** from source operational or archive systems which are the primary source of data for the data warehouse

Transform

- **transforming the data** - which may involve cleaning, filtering, validating and applying business rules

Load

- **loading the data** into a data warehouse or any other database or application that houses data



Copyright © Capgemini 2010. All Rights Reserved. 30

➤ **ETL** stands for extraction, transformation and loading. Etl is a process that involves the following tasks:

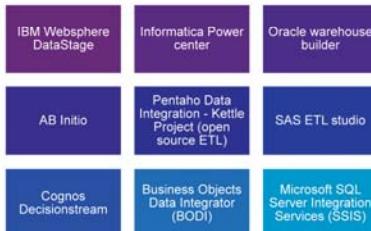
➤ **extracting data** from source operational or archive systems which are the primary source of data for the data warehouse

➤ **transforming the data** - which may involve cleaning, filtering, validating and applying business rules

➤ **loading the data** into a data warehouse or any other database or application that houses data

ETL Tools & Different providers

- ETL Tools:
- Extract, transform, load tools are software packages that facilitate the performing of ETL tasks.
- At present the most popular and widely used ETL tools and applications on the market are:

Copyright © Capgemini 2010. All rights reserved. 21

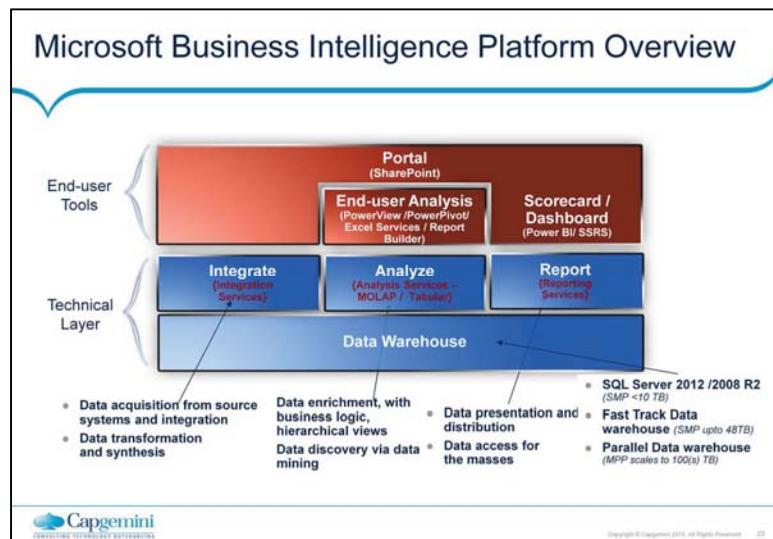
IBM Websphere DataStage (Formerly known as Ascential DataStage and Ardent DataStage)
Informatica Power center
Oracle warehouse builder
AB Initio
Pentaho Data Integration - Kettle Project (open source ETL)
SAS ETL studio
Cognos Decisionstream
Business Objects Data Integrator (BODI)
Microsoft SQL Server Integration Services (SSIS)

SSIS AND INFORMATICA		
Informatica	SSIS	Description
Aggregator Transformation Expression Transformation	Aggregate Transformation Derived Column Transformation	Collects data with alike attributes. <i>Manipulates data in the transformation pipeline using a data expression language.</i>
Filter Transformation Joiner Transformation Lookup Transformation Transformation Mapping	Conditional Split Transformation Merge Join Transformation Lookup Transformation	Filters data in the transformation pipeline. <i>Amalgamates two streams of data.</i>
Mapping	Data flow	Fetches data from a source external to SSIS. <i>Defines how data moves from one place to another. A mapping contains a definition of a source and target, but doesn't care where they are.</i>
Router Transformation	n/a	A mapping is a collection of transformations that operate as a single module of work.
Session	A session is an instantiation of a mapping. There is no instantiation of a session or instantiation of a data flow in SSIS. Sort Transformation	Sends data in the transformation pipeline to a target location according on attributes of that data. <i>A session is the execution of a mapping and is where the source & target locations are defined. Workflows control sessions, not mappings.</i>
Source Transformation Source Source Qualifier	Connection Source Adapter	Represents data. <i>The component used to fetch data for the transformation pipeline.</i>
Target	Destination Adapter	Represents where data is sent in the transformation pipeline will eventually reside.
Transformation	Transformation	A component that applies an operation to some data.
Workflow	Control flow or package	Groups related designed objects into a single, executable, unit of work.
Worklet	Sequence Container	Logically group related components into a single unit of work.

Source – MS whitepaper - Jamie



Copyright © Capgemini 2010. All Rights Reserved.



Integration Services: An Introduction

- Integration Services is a platform for building high performance data integration and workflow solutions, including extraction, transformation, and loading (ETL) operations for data warehousing.
- Integration Services includes graphical tools and wizards for building and debugging packages; tasks for performing workflow functions such as FTP operations, SQL statement execution, and e-mail messaging; data sources and destinations for extracting and loading data; transformations for cleaning, aggregating, merging, and copying data; a management service, the Integration Services service, for administering Integration Services packages; and application programming interfaces (APIs) for programming the Integration Services object model.
- Integration Services=>
- Provides functionality commonly referred to as Extract, Transform, Load (ETL).
- Moves and transforms data between sources and destinations, regardless of format.
- Cleanses data and ensures data integrity.
- Integrates heterogeneous data sources.

Copyright © Capgemini 2010. All Rights Reserved. 20

SSIS Background

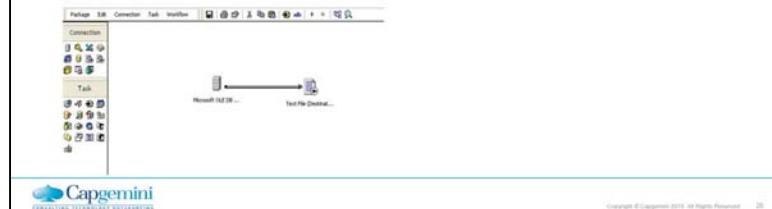
Lesson 2:

SQL Server 2000 DTS Limitations

Offered a limited set of tasks and transformations.

The graphical designer combined the workflow and data flow on a single design surface and offered limited control flow options.

- In DTS, the designer consists of a single pane: the Connections and the Workflow task
- SSIS Data Flow engine is significant faster than DTS Data Pump



DTS provided only a limited set of tasks and transformations. The graphical designer that was used to create packages combined the workflow and data flow on a single design surface and offered limited control flow options.

The DTS and SSIS are both used in SQL Server Transformations. The DTS was used for transformation purpose upto SQL Server 2000 and SSIS (Including SSAS and SSRS) are the advanced to DTS and are included in SQL Server 2005 onwards. The capability of SSIS tool in comparison with DTS is tremendous. This SSIS tool is in competition with other ETL Tools in the present. The SSIS has lot added features as per the current requirements.

Advantages of SSIS 2005, 2008, R2 and 2012,2014,2016 over DTS

The SSIS has lot added features as per the current requirements as compared to DTS

- SSIS, the designer is split into 5 design panes



- SSIS has new many built in tasks which can save significant amount of time compare to manual coding in DTS (e.g. Looping through files in a folder is inbuilt in SSIS - See Foreach Loop Container)
- Making package dynamic at runtime is easier using expression and configurations
- DataViewer support to debug DataFlow
- Event Handlers support with separate control flow for each event
- ScriptTask in SSIS uses VB.net language with rich .net framework support. Script Task code is compiled so run much faster than DTS ActiveX script.
- Expression based control flow along with Success, Failure and Complete

The Evolution of Microsoft Data Platform



Copyright © Capgemini 2014. All Rights Reserved. 27

Added features of SSIS 2016

- Better deployment:
- Better debugging
- Better package management
- Expanded connectivity on premises
- Usability and productivity

SQL Server 2016 editions – what's new

SQL Server 2016 is the biggest leap forward in the Microsoft data platform history with real-time operational analytics, rich visualizations on mobile devices, built-in advanced analytics, new advanced security technology, and new hybrid cloud scenarios.

Enterprise	Standard	Express	Developer
Mission critical performance	Security	Data warehousing	Business intelligence
<ul style="list-style-type: none"> ▪ Clustering, System Health ▪ Always On ▪ DC DR failover ▪ Enhanced AlwaysOn (with 100% HA) ▪ QuantumLink ▪ Framework 	<ul style="list-style-type: none"> ▪ Always Encrypted ▪ Dynamic data masking ▪ Column level encryption ▪ Enhanced DBX server ▪ Transparent data ▪ Policy based management 	<ul style="list-style-type: none"> ▪ Enhanced security monitoring ▪ Improved security audit ▪ Policy-based management ▪ Prohibited ▪ Support for mode (only) ▪ Support for OLEDB 	<ul style="list-style-type: none"> ▪ EMEA for cloud including BI on demand ▪ BI environment ▪ BI environment analysis ▪ BI environment configuration ▪ Advanced validation ▪ Advanced validation ▪ Model validation ▪ Model validation report ▪ Enhanced reporting ▪ Brain valuator 1.1000 ▪ Brain valuator 1.1000 ▪ BI reports for Project BI ▪ BI environment analysis ▪ Brain reporting and visualization ▪ Advanced reports
Advanced analytics			Hybrid cloud
			<ul style="list-style-type: none"> ▪ In-database, advanced analytics ▪ Predictive analysis ▪ Enhanced risk and compliance ▪ Models with time dimension ▪ Brain Analytics ▪ Brain Analytics ▪ Azure Data Factory and Stream Analytics ▪ Stream Analytics ▪ Machine learning

The product names displayed on this page are the trademarks or registered trademarks of their respective owners and are used, as required, under license or with permission of the owner or the registered trademark owner or for publicity purposes only. Microsoft is a registered trademark of Microsoft Corporation in the United States and/or other countries.

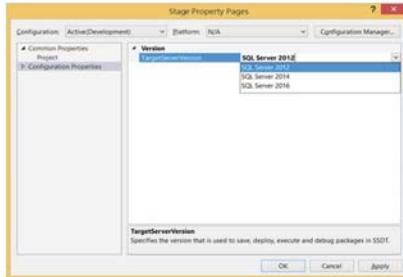
 Capgemini
PERFORMING BUSINESS BETTER

Microsoft

Copyright © Capgemini 2016. All rights reserved. 20

Added features of SSIS 2016

- SSDT for VS2015 supports SSIS 2012, 2014 and 2016



The screenshot shows the 'TargetServerVersion' property page in the 'Properties' window of the SSDT interface. The 'Configuration Manager' dropdown is set to 'SQL Server 2016'. Other options listed are 'SQL Server 2017', 'SQL Server 2014', and 'SQL Server 2012'. The 'OK' button is highlighted.

*Not in current preview/screenshot photoshopped

 Capgemini
PERFECT YOUR BUSINESS

When you upgrade SSIS projects from previous versions to the current version, the project-level connection managers continue to work as expected and the package layout and annotations are retained.

Added features of SSIS 2016 - Error column

- Error column

- ErrorCode (integer)

-

- ErrorCode can be translated to a description with a Script Component

- ErrorColumnName?

- Custom Transformation

- Looping through copy of package

- Now similar solution as for the Error description

Copyright © Capgemini 2016. All Rights Reserved.

<http://microsoft-ssis.blogspot.com/2015/09/sql-server-2016-ctp-23-get-error.html>

```
// C# Code
public override void Input0_ProcessInputRow(Input0Buffer Row)
{
    Row.ErrorDescription =
this.ComponentMetaData.GetErrorDescription(Row.ErrorCode);
    Row.ErrorColumnName =
this.ComponentMetaData.GetIdentificationStringByLineageID(Row.ErrorColumn);
}
```

Added features of SSIS 2016- OData source and connection manager

- Open Data protocol initiated by Microsoft in 2007 (v1 to v3)
- Standardized in 2014 (v4)
- Available as separate download for SSIS 2012 and 2014
- Included in SSDT CTP 2.3
- Supports v4 protocol

Copyright © Capgemini 2014. All Rights Reserved. 31

oData protocol: allows the creation and consumption of queryable and interoperable RESTful APIs in a simple and standard way

<http://microsoft-ssis.blogspot.com/2015/09/sql-server-2016-ctp-23-odata-v4-protocol.html>

Added features of SSIS 2016-New database roles for SSIS

- ssis_monitor - used for AlwaysOn, not designed to use directly
- ssis_logreader - Allows you to see all executions on catalog reports.
Alternative for ssis_admin which gives too much rights

Copyright © Capgemini 2014. All Rights Reserved. 30

<http://microsoft-ssis.blogspot.com/2015/09/sql-server-2016-ctp-23-new-database.html>

Added features of SSIS 2016-Logging Levels

- 2008 => developers decided what to log
- 2012 => introduction of 4 logging levels None, Performance, Basic and Verbose
- 2016 => Extending the 4 logging levels with your own customized logging level.

Microsoft® SQL Server® 2016. All Rights Reserved. © 2016

<http://microsoft-ssis.blogspot.com/2015/08/sql-server-2016-ctp-23-custom-logging.html>

Added features of SSIS 2016 - Logging level RuntimeLineage

- Is intended to support a lineage library, which is not yet released.
- Lineage library will probably not included in SSIS, but in other tools.
Probably in Azure Data Factory.

Copyright © Capgemini 2016. All Rights Reserved. 30

Azure Data Factory is a guess

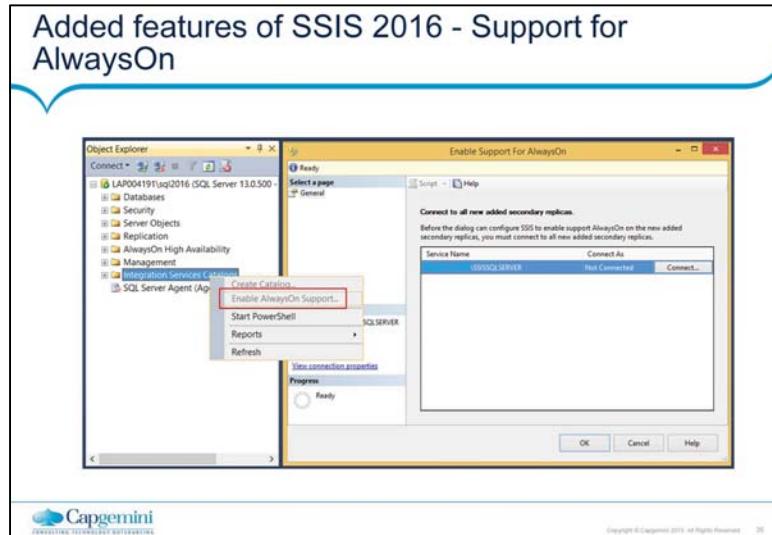
Added features of SSIS 2016 –Deployment model

- 2008 => Package Deployment Model
- 2012 => Project Deployment Model
- 2016 => Introduction of Incremental Package Deployment for the Project Deployment Model

Copyright © Capgemini 2014. All Rights Reserved. 30

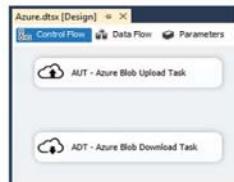
<http://microsoft-ssis.blogspot.com/2015/06/ssis-2016-ctp2-incremental-deployment.html>

Note: can't deploy project connection managers or parameters



Added features of SSIS 2016 - Azure Pack

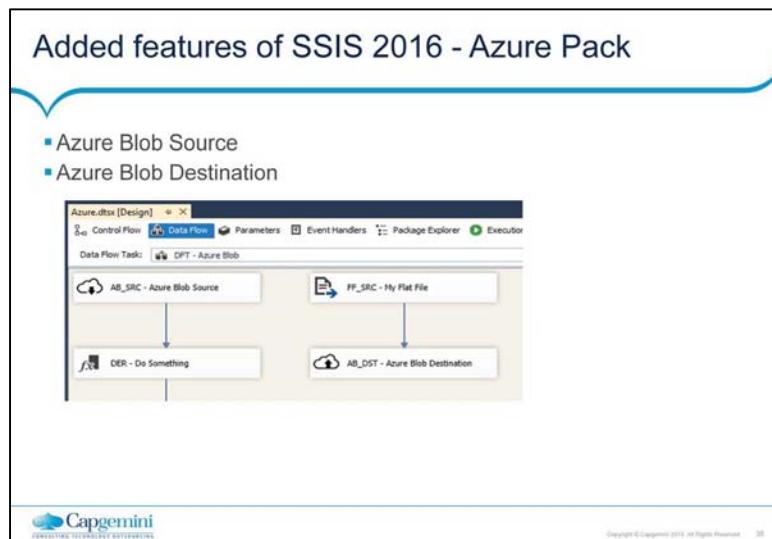
- Azure Blob Upload Task
- Azure Blob Download Task
- Azure File System Task (still missing)



 Capgemini
PERFORMANTES INNOVATRICES

Copyright © Capgemini 2014. All Rights Reserved. 31

<http://microsoft-ssis.blogspot.com/2015/06/azure-upload-and-download-tasks.html>



<http://microsoft-ssis.blogspot.com/2015/06/azure-blob-source-and-destination.html>

Added features of SSIS 2016 - Azure Pack

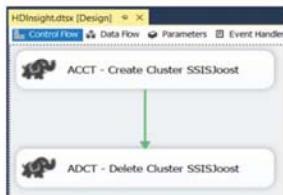
- Foreach Azure Blob Enumerator

The screenshot shows the 'Foreach Loop Editor' dialog box. The title bar says 'Foreach Loop Editor'. The main area has a tab labeled 'General' which is selected, and another tab labeled 'Expressions'. The content area is titled 'Foreach Azure Blob Enumerator'. It includes sections for 'Enumerator' (set to 'Foreach Azure Blob Enumerator'), 'Storage configuration' (set to 'SSIS Connection Manager for Azure Storage'), 'Location' (set to 'Container1'), and 'Filter' (set to 'blob1.txt')). At the bottom are 'OK', 'Cancel', and 'Help' buttons.

<http://microsoft-ssis.blogspot.com/2015/06/azure-blob-enumerator.html>

Added features of SSIS 2016 - Azure Pack

- Azure HDInsight Create Cluster Tasks
- Azure HDInsight Delete Cluster Tasks

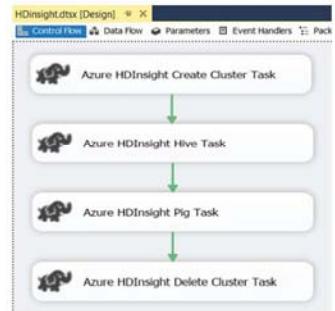


Copyright © Capgemini 2014. All Rights Reserved. 40

<http://microsoft-ssis.blogspot.com/2015/06/azure-hdinsight-create-delete-cluster.html>

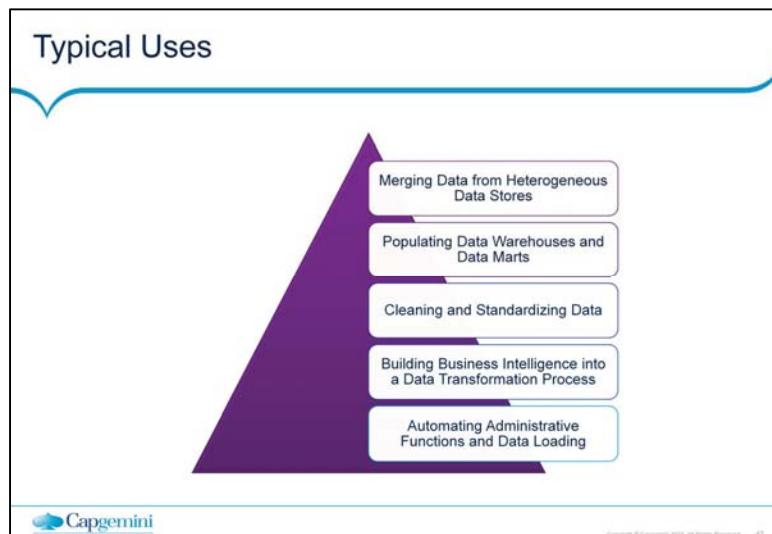
Added features of SSIS 2016 - Azure Pack

- Azure HDInsight Hive Tasks
- Azure HDInsight Pig Tasks



Typical Uses of Integration Services

Lesson 3



Merging Data from Heterogeneous Data Stores

Data is typically stored in many different data storage systems, and extracting data from all sources and merging the data into a single, consistent dataset is challenging. This situation can occur for a number of reasons. For example:

Many organizations archive information that is stored in legacy data storage systems. This data may not be important to daily operations, but it may be valuable for trend analysis that requires data collected over a long period of time.

Branches of an organization may use different data storage technologies to store the operational data. The package may need to extract data from spreadsheets as well as relational databases before it can merge the data.

Data may be stored in databases that use different schemas for the same data. The package may need to change the data type of a column or combine data from multiple columns into one column before it can merge the data.

Populating Data Warehouses and Data Marts

The data in data warehouses and data marts is usually updated frequently, and the data loads are typically very large.

Integration Services includes a task that bulk loads data directly from a flat file into SQL Server tables and views, and a destination component that bulk loads data into a SQL Server database as the last step in a data transformation process.

An SSIS package can be configured to be restartable. This means you can rerun the package from a predetermined checkpoint, either a task or container in the package. The ability to restart a package can save a lot of time, especially if the package processes data from a large number of sources.

Cleaning and Standardizing Data

Whether data is loaded into an online transaction processing (OLTP) or online analytic processing (OLAP) database, an Excel spreadsheet, or a file, it needs to be cleaned and standardized before it is loaded. Data may need to be updated for the following reasons:

Data is contributed from multiple branches of an organization, each using different conventions and standards. Before the data can be used, it may need to be formatted differently. For example, you may need to combine the first name and the last name into one column.

Data is rented or purchased. Before it can be used, the data may need to be standardized and cleaned to meet business standards. For example, an organization wants to verify that all the records use the same set of state abbreviations or the same set of product names.

Data is locale-specific. For example, the data may use varied date/time and numeric formats. If data from different locales is merged, it must be converted to one locale before it is loaded to avoid corruption of data.

Building Business Intelligence into a Data Transformation Process

A data transformation process requires built-in logic to respond dynamically to the data it accesses and processes.

The data may need to be summarized, converted, and distributed based on data values. The process may even need to reject data, based on an assessment of column values.

To address this requirement, the logic in the SSIS package may need to perform the following types of tasks:

Merging data from multiple data sources.

Evaluating data and applying data conversions.

Splitting a dataset into multiple datasets based on data values.

Applying different aggregations to different subsets of a dataset.

Loading subsets of the data into different or multiple destinations.

Integration Services provides containers, tasks, and transformations for building business intelligence into SSIS packages.

Automating Administrative Functions and Data Loading

Administrators frequently want to automate administrative functions such as backing up and restoring databases, copying SQL Server databases and the objects they contain, copying SQL Server objects, and loading data. Integration Services packages can perform these functions.

Integration Services includes tasks that are specifically designed to copy SQL Server database objects such as tables, views, and stored procedures; copy SQL Server objects such as databases, logins, and statistics; and add, change, and delete SQL Server objects and data by using Transact-SQL statements.

Administration of an OLTP or OLAP database environment frequently includes the loading of data. Integration Services includes several tasks that facilitate the bulk loading of data. You can use a task to load data from text files directly into SQL Server tables and views, or you can use a destination component to load data into SQL Server tables and views after applying transformations to the column data.

Tools

Lesson 4

Tools

- SQL Server Data Tools
- BIDS can perform the following tasks:

SQL Server Data Tools
creates packages that include complex control flow, data flow, event-driven logic, and logging.

Test and debug packages

Create configurations

Create a deployment utility that can install packages

Save copies of packages to the SQL Server msdb database, the SSIS Package Store,



Capgemini
PERFORMANT TECHNICAL INNOVATION

Copyright © Capgemini 2018. All Rights Reserved. A1

Microsoft SQL Server includes two studios for working with Integration Services:
SQL Server Data Tools, for developing the Integration Services packages that a business solution requires,
and SQL Server Management Studio, for managing packages in a production environment.

SQL Server Data Tools provides the Integration Services project in which you create packages, their data sources, and data source views.

Working in you can perform the following tasks:

- Run the SQL Server Import and Export Wizard to create basic packages that copy data from a source to a destination.
- Create packages that include complex control flow, data flow, event-driven logic, and logging.
- Test and debug packages by using the troubleshooting and monitoring features in SSIS Designer, and the debugging features in Business Intelligence Development Studio.
- Create configurations that update the properties of packages and package objects at run time.
- Create a deployment utility that can install packages and their dependencies on other computers.
- Save copies of packages to the SQL Server **msdb** database, the SSIS Package Store, and the file system.

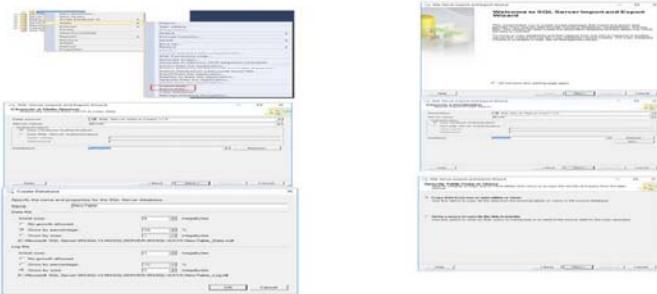
Tools...

- SQL Server Management Studio provides the Integration Services service to manage packages and monitor running packages.
- It performs following tasks



TOOLS.....

- Import and Export Wizard
 - The SQL Server Import and Export Wizard provides the simplest method of copying data between data sources and of constructing basic packages



Copyright © Capgemini 2010. All Rights Reserved.

Capgemini
PERFECTING TECHNOLOGY FOR BUSINESS

Third tools is Import and Export Wizard . The SQL Server Import and Export Wizard provides the simplest method of copying data between data sources and of constructing basic packages.

Here we will see that how can we load data using Import export wizard.

To start the SQL Server Import and Export Wizard

On the Start menu, point to All Programs, point to Microsoft SQL Server 2008, and then click Import and Export Data.

—Or—

In Business Intelligence Development Studio, right-click the SSIS Packages folder, and then click SSIS Import and Export Wizard.

—Or—

In Business Intelligence Development Studio, on the Project menu, click SSIS Import and Export Wizard.

—Or—

In SQL Server Management Studio, connect to the Database Engine server type, expand Databases, right-click a database, point to Tasks, and then click Import Data or Export data.

—Or—

In a command prompt window, run DTSWizard.exe, located in C:\Program Files\Microsoft SQL Server\100\DTSP\Binn.

To use the SQL Server Import and Export Wizard for importing and exporting data

Start the SQL Server Import and Export Wizard.

On the corresponding wizard pages, select a data source and a data destination.

In our example Browse and choose the Customers.xls which is located in the Sample Data Folder Verify that the Excel version box contains Microsoft Excel 97-2005 and the First Row has column names check box is selected

On the Choose Destination page, in the Destination list, Select SQL Native Client, and in the Server name box, type localhost.

In the Database list, select AdventureWorks.

On the Specify Table Copy or Query page, click Write a query to specify the data to transfer and click Next If you want to copy it from a table or views then click the other option.

On the Provide a Source Query page, in the SQL statement box,type the query.

Click Next

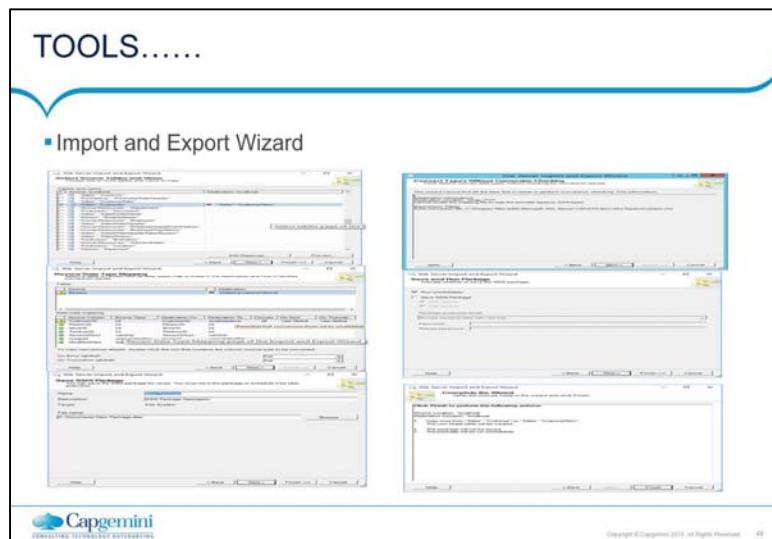
On the Select Source Tables and Views page, click [AdventureWorks].[dbo].[Query] in the Destination list, and change the table name, Query, to Prospective Customers.

To edit column metadata and table options, click Edit Mappings

On the Columns Mappings page, verify that the Create Destination table option is selected, select the Drop and re-create destination table check box, and modify the metadata of the destination columns

On the Complete the Wizard page, review information about the new package and click Finish

On the Performing Operations page, view the actions that the wizard performs. When finished, the Status column for each action should display Success



Third tools is Import and Export Wizard . The SQL Server Import and Export Wizard provides the simplest method of copying data between data sources and of constructing basic packages.

Here we will see that how can we load data using Import export wizard.

To start the SQL Server Import and Export Wizard

On the Start menu, point to All Programs, point to Microsoft SQL Server 2008, and then click Import and Export Data.

—or—

In Business Intelligence Development Studio, right-click the SSIS Packages folder, and then click SSIS Import and Export Wizard.

—or—

In Business Intelligence Development Studio, on the Project menu, click SSIS Import and Export Wizard.

—or—

In SQL Server Management Studio, connect to the Database Engine server type, expand Databases, right-click a database, point to Tasks, and then click Import Data or Export Data.

—or—

In a command prompt window, run DTSWizard.exe, located in C:\Program Files\Microsoft SQL Server\100\DTSP\Binn.

To use the SQL Server Import and Export Wizard for importing and exporting data

Start the SQL Server Import and Export Wizard.

On the corresponding wizard pages, select a data source and a data destination.

In our example Browse and choose the Customers.xls which is located in the Sample Data Folder

Verify that the Excel version box contains Microsoft Excel 97-2005 and the First Row has column names check box is selected

On the Choose Destination page, in the Destination list, Select SQL Native Client, and in the Server name box, type localhost.

In the Database list, select AdventureWorks.

On the Specify Table Copy or Query page, click Write a query to specify the data to transfer and click Next If you want to copy it from a table or views then click the other option.

On the Provide a Source Query page, in the SQL statement box,type the query.

Click Next

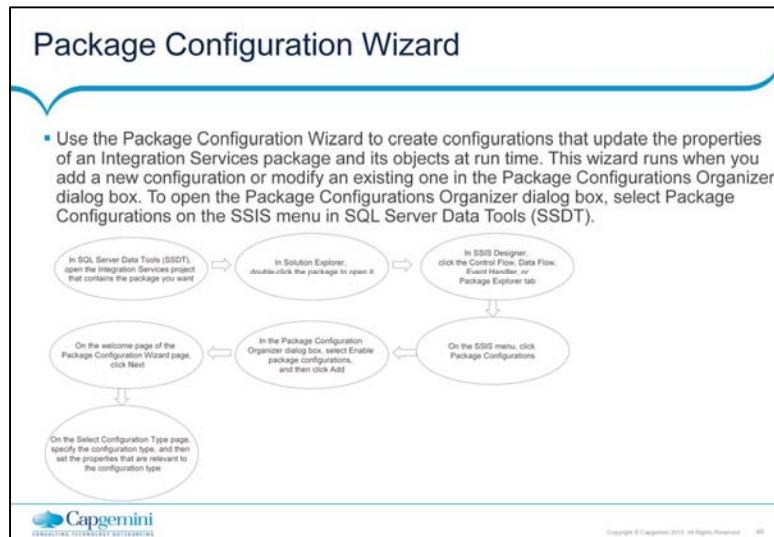
On the Select Source Tables and Views page, click [AdventureWorks].[dbo].[Query] in the Destination list, and change the table name, Query, to Prospective_Customers.

To edit column metadata and table options, click Edit Mappings

On the Columns Mappings page, verify that the Create Destination table option is selected, select the Drop and re-create destination table check box, and modify the metadata of the destination columns

On the Complete the Wizard page, review information about the new package and click Finish

On the Performing Operations page, view the actions that the wizard performs. When finished, the Status column for each action should display Success



SSIS Objects

Lesson 5:

Objects and Concepts

Topic	Description
Integration Services Packages	Describes packages.
Control Flow Elements	Describes containers, tasks, and precedence constraints.
Data Flow Elements	Describes sources, transformations, destinations, and paths.
Integration Services Connections	Describes connection managers.
Integration Services Variables	Describes user-defined variables and the system variables that Integration Services provides.
Integration Services Event Handlers	Describes the run-time events and the event handlers you can build for the events.
Integration Services Log Providers	Describes the log providers that Integration Services includes.

```
graph TD; Package[Package] --> CFE[Control Flow Elements]; CFE --> DFE[Data Flow Elements]; DFE --> ISC[Integration Services Connections]; ISC --> V[Variables]; V --> EH[Event Handlers]; EH --> LP[Log Providers];
```

 Capgemini
PERFORMANTES INNOVATRICES RELIABLES

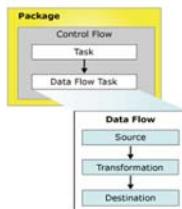
Copyright © Capgemini 2014. All Rights Reserved. 01

To start SSIS in details like how to create SSIS project, first you should be familiar with some Integration Services concepts to help you create Integration Services packages successfully from the very beginning. These objects and concepts are relevant to the use of the Integration Services tools. Those concepts include the following:

- Package
- Control Flow Elements
- Data Flow Elements
- Integration Services Connections
- Variables
- Event Handlers
- Log Providers

Package..

- A package is an organized collection of connections, control flow elements, data flow elements, event handlers, variables, and configurations.,
- The package is the unit of work that is retrieved, executed, and saved.
- The following diagram shows a simple package that contains a control flow with a Data Flow task, which in turn contains a data flow.



 Capgemini
PERFORMING TECHNOLOGY

Copyright © Capgemini 2014. All Rights Reserved. 

A package is an organized collection of connections, control flow elements, data flow elements, event handlers, variables, and configurations, that you assemble using either the graphical design tools that SQL Server Integration Services provides, or build programmatically. We then save the completed package to SQL Server, the SSIS Package Store, or the file system.

When we first create a package, it is an empty object that does nothing. To add functionality to a package, we add a control flow and, optionally, one or more data flows to the package.

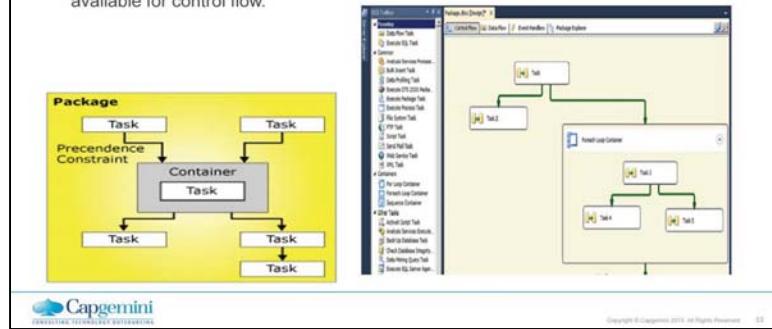
AS diagram shows a simple package with a control flow with a Data Flow task, which in turn contains a data flow.

In next slides we will go through with control flow and data flow.

Control Flow Elements

- Control flow consists of below components:

- Create the control flow in a package by using the Control Flow tab in SSIS Designer.
- When the Control Flow tab is active, the Toolbox lists the tasks and containers that are available for control flow.



➤ A control flow consists of the tasks ,containers and precedence constraints . containers that provide structures in packages.

The tasks perform specific types of work such as executing SQL statements or sending email messages, and the containers define repeating subsets of the control flow or group subsets of the control flow to make the package easier to manage. The tasks and containers are usually connected by precedence constraints that specify the sequence in which tasks and containers are executed and the conditions that must be satisfied to run the next task or container in the control flow.

Tasks

- Task: Tasks do the work in packages. Integration Services includes tasks for performing a variety of functions.
- The Data Flow task

DFT defines and runs data flows that extract data, apply transformations, and load data.

DFT encapsulates the data flow engine that moves data between sources and destinations.

At run time, the DFT builds an execution plan from the data flow, and the data flow engine executes the plan.

A Data Flow task can include multiple data flows.

Copyright © Capgemini 2010. All Rights Reserved.

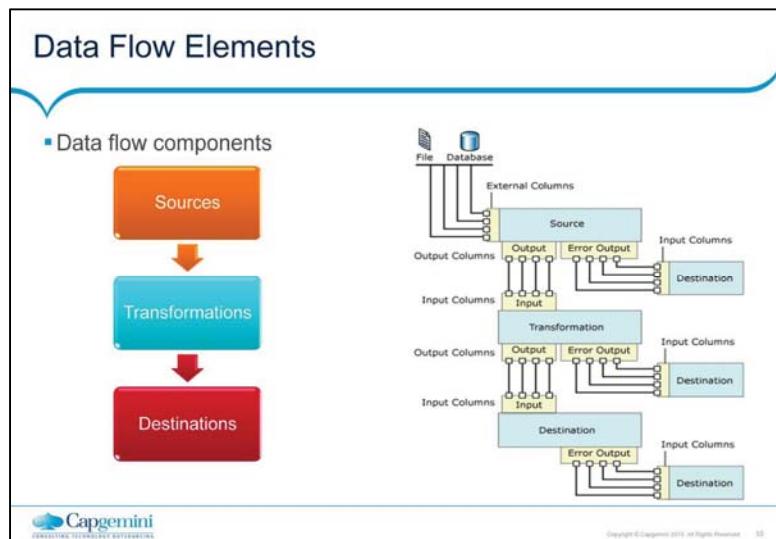
Tasks do the work in packages. Integration Services includes tasks for performing a variety of functions.

The Data Flow task encapsulates the data flow engine that moves data between sources and destinations, and lets the user transform, clean, and modify data as it is moved. Addition of a Data Flow task to a package control flow makes it possible for the package to extract, transform, and load data.

A data flow consists of at least one data flow component, but it is typically a set of connected data flow components: sources that extract data; transformations that modify, route, or summarize data; and destinations that load data. Components are connected in the data flow by paths. Each path specifies the two components that are the start and the end of the path.

A Data Flow task can include multiple data flows. If a task copies several sets of data, and if the order in which the data is copied is not significant, it can be more convenient to include multiple data flows in the Data Flow task. For example, you might create five data flows, each copying data from a flat file into a different dimension table in a data warehouse star schema.

However, the data flow engine determines order of execution when there are multiple data flows within one data flow task. Therefore, when order is important, the package should use multiple Data Flow tasks, each task containing one data flow. You can then apply precedence constraints to control the execution order of the tasks.



Definition of Connection Manager

- "A Connection Manager is a logical representation of a connection". At design time, one sets its properties to describe the physical connection that SSIS creates when the package runs. For instance, a connection manager includes the Connection String property that you set at design time; at run time, a physical connection is created using the value in the connection string property.
- NOTE: A package can use multiple instances of a connection manager type, and you can set the properties on each instance. At run time, each instance of a connection manager type creates a connection that has different attributes.

Copyright © Capgemini 2010. All Rights Reserved. 00

Connection Managers

- The following table(s) show the various connections that can be created using Connection Manager:

Type	Description
ADO	Connects to ActiveX Data Objects (ADO) objects.
ADO.NET	Connects to a data source by using a .NET provider.
EXCEL	Connects to an Excel workbook file.
FILE	Connects to a file or a folder.
FLATFILE	Connect to data in a single flat file.
FTP	Connect to an FTP server.
HTTP	Connects to a web server.
MSMQ	Connects to a message queue.
MSOLAP90	Connects to an instance of SQL Server 2008 Analysis Services (SSAS) or an Analysis Services project.

Type	Description
MULTIFILE	Connects to multiple files and folders.
MULTIFLATFILE	Connects to multiple data files and folders.
OLEDB	Connects to a data source by using an OLE DB provider.
ODBC	Connects to a data source by using ODBC.
SMO Server	Connects to a SQL Management Objects (SMO) server.
SMTP	Connects to an SMTP mail server.
SQLMOBILE	Connects to a SQL Server Mobile database.
WMI	Connects to a server and specifies the scope of Windows Management Instrumentation (WMI) management on the server.

Copyright © Capgemini 2010. All Rights Reserved. E1

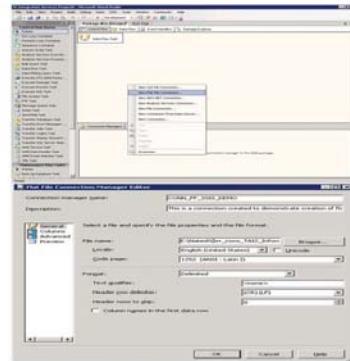
Connection Managers

- The following sections will explain at length, the method(s) to create the below mentioned types of connections:
 - FLAT FILE
 - EXCEL
 - OLEDB
 - ODBC

Copyright © Capgemini 2010. All Rights Reserved.

Flat File Connection

- Right-click in the "Connection Managers" pane and select "New Flat File Connection..."
- Completion of step 1 opens the "Flat File Connection Manager Editor". Select the "General" section if it's not the default.
- Specify the ---
- Name of this Connection Manager,
- Description,
- File Name

Copyright © Capgemini 2010. All Rights Reserved

Flat File Source

- Support for varying number of columns

OrderID,CustomerID,EmployeeID,ShipVia
10248,VINET,5,3
10249,TOMSP,6
10250,HANAR,4,2

OrderID	CustomerID	EmployeeID	ShipVia
10248	VINET	5	3
10249	TOMSP	6	
10250	HANAR	4	2

- Embedded qualifiers

ID,Title
1148,'Can''t Buy a Thrill'
1149,'Echoes, Silence, Patience & Grace'

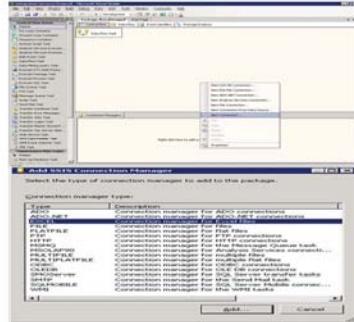
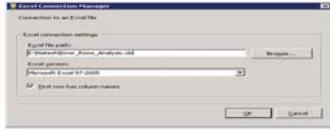
ID	Title
1148	Can't buy a thrill
1149	Echoes, Silence, Patience & Grace



Copyright © Capgemini 2010. All Rights Reserved.

Excel Connection

- Right-click in the "Connection Managers" pane and select "New Connection..."
- Completion of STEP 1 opens the "Add SSIS Connection Manager" dialog box. Select the type as "EXCEL"
- Specify the Excel File Path



Capgemini
PERFECTING SUSTAINABLE BUSINESS

Copyright © Capgemini 2010. All Rights Reserved.

OLE DB Connection

- Right-click in the "Connection Managers" pane and select "New OLE DB Connection..."
- Completion of STEP 1 opens the "Configure OLE DB Connection Manager" dialog box. Select New if you would like to have something different from the existing ones (if any)
- When one selects New, a new dialog box will be opened for Selecting a provider & its respective information
- Select 'OK'

The screenshot displays the 'Configure OLE DB Connection Manager' dialog box. On the left, a list of available providers is shown: OLE DB for Oracle, OLE DB for Performance, and OLE DB for Text. On the right, the 'Data connection properties' section is visible, showing the following settings:

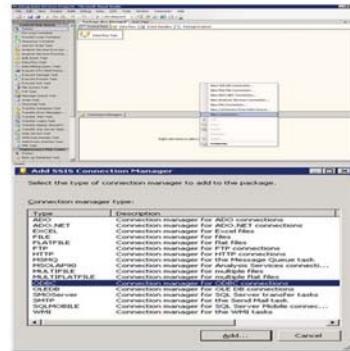
Property	Value
Entry Source	SSPI
Integrated Security	SSPI
Provider	SQLOLEDB

Buttons for 'Next...', 'Delete', 'OK', and 'Cancel' are located at the bottom right of the dialog.

Capgemini
PERFORMANT. INNOVATIVE. SUSTAINABLE.

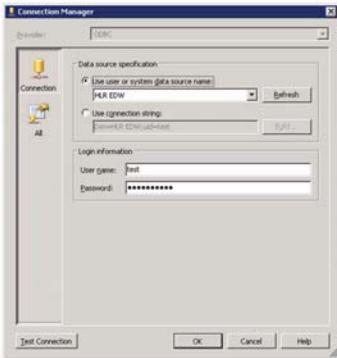
ODBC Connection

- Right-click in the "Connection Managers" pane and select "New Connection..."

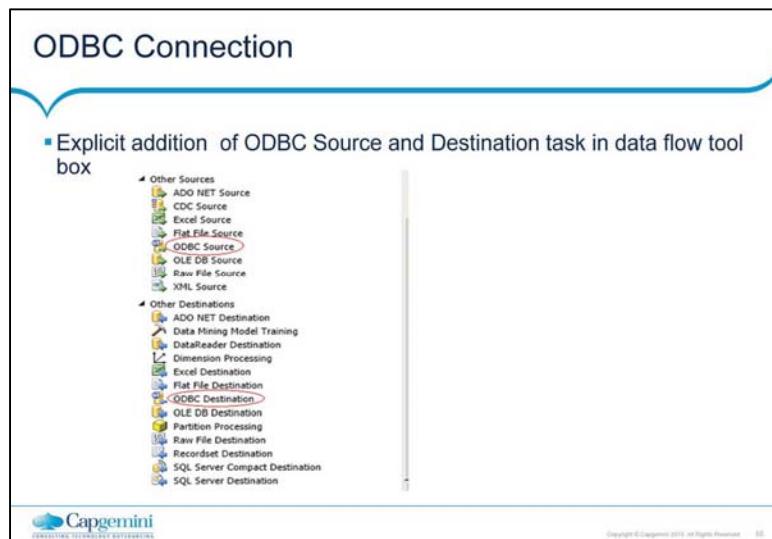
Copyright © Capgemini 2010. All Rights Reserved.

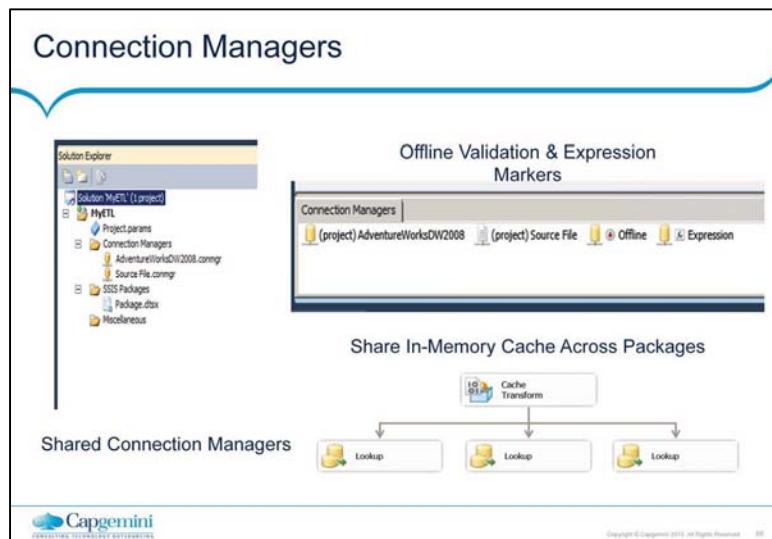
ODBC Connection

- Completion of STEP 2 opens the "Configure ODBC Connection Manager" dialog box. Select New if you would like to have something different from the existing ones (if any).
- When one selects New, a new dialog box will be opened for specifying User or System DSN & Login Information
- Select 'OK' to successfully complete creating the connection



Copyright © Capgemini 2010. All Rights Reserved.



Copyright © Capgemini 2010. All Rights Reserved00-66

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

Lesson 6:Data Flow Tasks

Data Flow Task

- The Data Flow task encapsulates the data flow engine that moves data between sources and destinations, providing the facility to transform, clean, and modify data as it is moved.
- Addition of a Data Flow task to a package control flow makes it possible for the package to extract, transform, and load data.
- A data flow consists of at least one data flow component, but it is typically a set of connected data flow components: sources that extract data; transformations that modify, route, or summarize data; and destinations that load data.
- Components are connected in the data flow by paths. Each path specifies the two components that are the start and the end of the path.

Copyright © Capgemini 2010. All Rights Reserved.

Data Flow Task

- A Data Flow task can include multiple data flows. However, the data flow engine determines order of execution when there are multiple data flows within one data flow task. Therefore, when order is important, the package should use multiple Data Flow tasks, each task containing one data flow. You can then apply precedence constraints to control the execution order of the tasks.
- The following diagram shows a Data Flow task with multiple data flows.



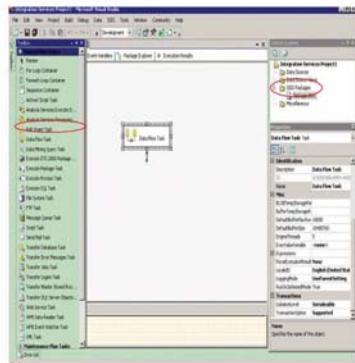
Data Flow Task

- The Data Flow task also manages error flows. At run time, row-level errors may occur when data flow components convert data, perform a lookup, or evaluate expressions. For example, a data column with a string value cannot be converted to an integer, or an expression tries to divide by zero. Both operations result in errors, and the rows containing errors can be processed separately using an error flow.

Copyright © Capgemini 2010. All Rights Reserved.

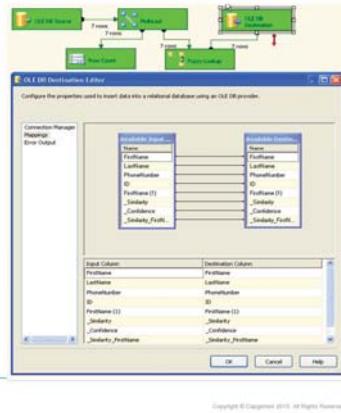
Data Flow Task

- To add a Data Flow task
 - Click the Control Flow tab.
 - In the Toolbox, expand Control Flow Items, and drag a Data Flow Task onto the design surface of the Control Flow tab.
 - On the Control Flow design surface, right-click the newly added Data Flow Task, click Rename, and change the name to Extract Sample Currency Data.
 - It is good practice to provide unique names to all components that you add to a design surface. For ease of use and maintainability, the names should describe the function that each component performs.



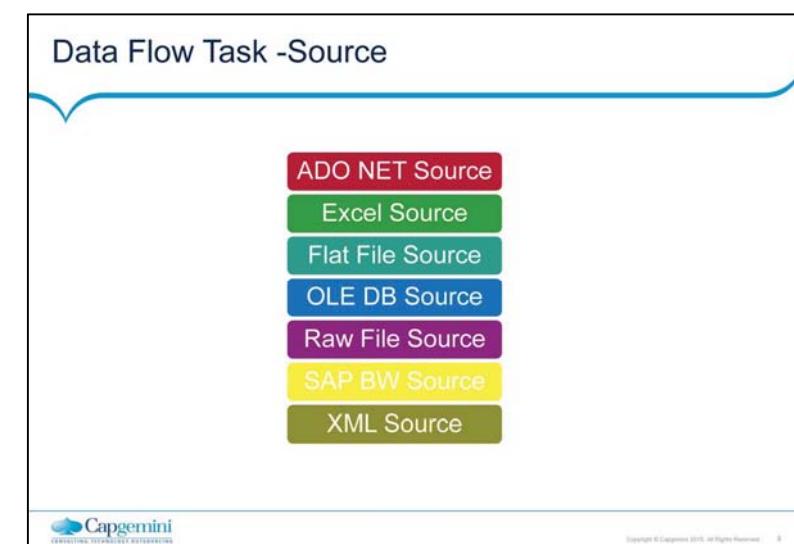
Data Flow Task

- Data Flow Sources
 - Eg.Flat file source,OIedb source
- Data Flow Transformations
 - Eg. Conditional Split ,Aggregate, Lookup
 - May have multiple inputs and outputs
 - Must have at least one input
 - Must have at least one output
 - Process rows by buffer
 - May output more or less rows than take on input
- Data Flow Destinations
 - Eg.Flat file destinations,Pointer,OLEDB Destinations



Data Flow Tasks- Source

Lesson 7:



Data Flow Task –Source....

- ADO NET - The ADO.NET source consumes data from a .NET provider and makes the data available to the data flow.
- Excel - The Excel source extracts data from worksheets or ranges in Microsoft Excel workbooks.
- Flat File - The Flat File source reads data from a text file. The text file can be in delimited, fixed width, or mixed format.
 - Delimited format uses column and row delimiters to define columns and rows.
 - Fixed width format uses width to define columns and rows. This format also includes a character for padding fields to their maximum width.
 - Ragged right format uses width to define all columns, except for the last column, which is delimited by the row delimiter.
- OLEDB - The OLE DB source extracts data from a variety of OLE DB-compliant relational databases by using a database table, a view, or an SQL command. For example, the OLE DB source can extract data from tables in Microsoft Office Access or SQL Server databases

Copyright © Capgemini 2010. All Rights Reserved

Data Flow Task –Source....

- Raw File - The Raw File source reads raw data from a file. Because the representation of the data is native to the source, the data requires no translation and almost no parsing. This means that the Raw File source can read data more quickly than other sources such as the Flat File and the OLE DB sources.
- SAP BW - The SAP BW source is the source component of the Microsoft Connector 1.1 for SAP BW. Thus, the SAP BW source extracts data from an SAP Netweaver BW version 7 system and makes this data available to the data flow in a Microsoft Integration Services package.
- XML - The XML source reads an XML data file and populates the columns in the source output with the data.
- The data in XML files frequently includes hierarchical relationships. For example, an XML data file can represent catalogs and items in catalogs. Before the data can enter the data flow, the relationship of the elements in XML data file must be determined, and an output must be generated for each element in the file.

Copyright © Capgemini 2010. All Rights Reserved

Data Flow Tasks- Transformation

Lesson 8:

Aggregate Transformation

- The Aggregate transformation applies aggregate functions, such as Average, to column values and copies the results to the transformation output. Besides aggregate functions, the transformation provides the GROUP BY clause, which you can use to specify groups to aggregate across.
- The Aggregate transformation is asynchronous, which means that it does not consume and publish data row by row. Instead it consumes the whole row set, performs its groupings and aggregations, and then publishes the results.
- The Aggregate transformation has one input and one or more outputs. It does not support an error output.

Copyright © Capgemini 2010. All Rights Reserved.

Aggregate Transformation

- The Aggregate transformation supports the following operations :
- Group by :- Divides datasets into groups. Columns of any data type can be used for grouping.
- Sum :- Sums the values in a column. Only columns with numeric data types can be summed.
- Average :- Returns the average of the column values in a column. Only columns with numeric data types can be averaged.
- Count :- Returns the number of items in a group.
- Count distinct :- Returns the number of unique nonnull values in a group.

Copyright © Capgemini 2010. All Rights Reserved.

Aggregate Transformation

- At the transformation level, you configure the Aggregate transformation for performance by specifying the number of keys and distinct count keys the transformation is expected to handle and the percentage by which memory can be extended during the aggregation. The Aggregate transformation can also be configured to generate a warning instead of failing when the value of a divisor is zero.
- At the output level, you configure the Aggregate transformation for performance by specifying the number of keys the output is expected to contain. The Aggregate transformation supports multiple outputs, and each can be configured differently.
- At the column level, you specify the aggregation that the column performs and the comparison options of the aggregation. You can also configure the Aggregate transformation for performance by specifying the number of keys and distinct count keys that each column contains, and identifying columns as Is Big if a column contains large numeric values or numeric values with high precision.

Copyright © Capgemini 2010. All Rights Reserved.

Aggregate Transformation

Example::

- The following example is showing a data set comprising of 4 rows. Suppose now we have to calculate the sum of the salary all cities and We can achieve this by using Aggregate Transformation.

"Account No"	"First Name"	"Last Name"	"Company"	"Address"	"City"	"Salary"
"10019"	"bobbi"	"Arnold"	"Market Place"	"1000 S Nicolet St"	"Sometown"	1000
"10023"	"bruce"	"Beecher"	"Madison & Hut"	"1037 W Wisconsin Ave"	"Smithville"	1500
"10024"	"bruce"	"Beyer"	"La Salle Clinic"	"108 E Wisconsin Ave"	"Jonestown"	1800
"10025"	"butch"	"Bobbi"	"Town & Count"	"108 Hillock Ct"	"Smithville"	3000



Copyright © Capgemini 2010. All Rights Reserved.

Aggregate Transformation

Output::

- The following is the output i.e total sum of the salary of all cities after Aggregate Transformation is done.

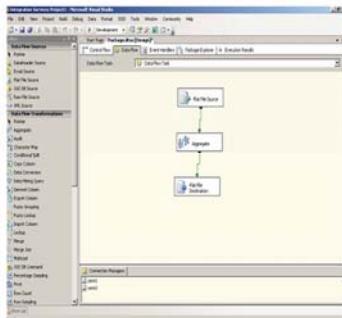
Account No	First Name	Last Name	Company	Address	City	sum of salary
						7300



Copyright © Capgemini 2010. All Rights Reserved

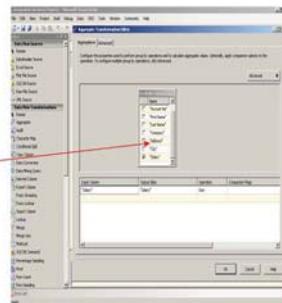
Aggregate Transformation

- Click the Data Flow tab, and then, from the Toolbox, drag the Aggregate transformation to the design surface
- Connect the Aggregate transformation to the data flow by dragging a connector—the green or red arrow—from the source or the previous transformation to the Aggregate transformation.



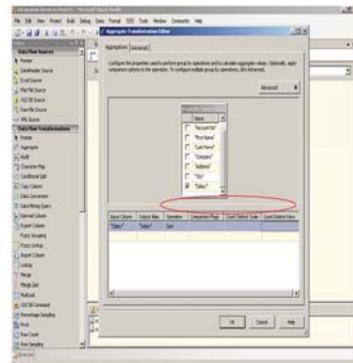
Aggregate Transformation

- Double-click the transformation.
- In the Editor dialog box,
- click the Aggregations tab.
- In the Available Input Columns list, select the 'Salary' check box by the columns on which we want to aggregate values. The selected columns appear in the table.
- Optionally, you can modify the value in the Output Alias columns.
- Default aggregation operation is Group by. choose the Sum operation. From the drop down
- Operation list.

Copyright © Capgemini 2007. All Rights Reserved.

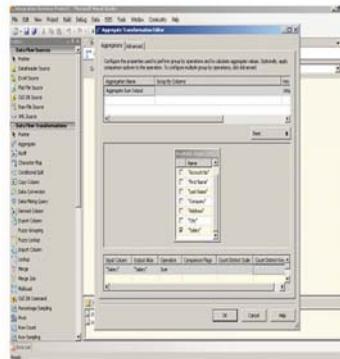
Aggregate Transformation

- To change the default comparison, select the comparison flags to use when defining the groups to aggregate in the Comparison Flags column.
- Optionally, for the Count distinct aggregation, specify an exact count of distinct values in the Count Distinct Keys column, or select an approximate count in the Count Distinct Scale column.

Copyright © Capgemini 2012. All Rights Reserved.

Aggregate Transformation

- Optionally, click Advanced and update the name of the Aggregate transformation output. If the aggregations include a group by operation, you can select an approximate count of grouping key values in the Keys Scale column, or specify an exact number of grouping key values in the Keys column.
- Optionally, click the **Advanced** tab and set the attributes that apply to all the operations that the Aggregate transformation performs.

Copyright © Capgemini 2007. All Rights Reserved

Aggregate Transformation

- The Aggregate transformation includes a set of properties that you can set to enhance the performance of the transformation.
- Set the Keys and Keys Scale properties of the component and the component outputs. Using Keys, you can specify the exact number of keys the transformation is expected to handle, and using Keys Scale, you can specify an approximate number of keys. When you specify a value for Keys, which is the value the transformation will receive when the package runs, the transformation avoids reorganizing cached totals, improving performance.
- Set the Count Distinct Keys and Count Distinct Scale properties of the component. Using Count Distinct Keys, you can specify the exact number of keys the transformation is expected to handle for a count distinct operation. Using Count Distinct Scale, you can specify an approximate number of keys for a count distinct operation. When you specify a value for Count Distinct Scale, which is the value the transformation will receive when the package runs, the transformation also avoids reorganizing cached totals, improving performance.

Copyright © Capgemini 2010. All Rights Reserved.

Audit Transformation

Lesson 9:

Audit Transformation

- "The Audit transformation enables the data flow in a package to include data about the environment in which the package runs. For example, the name of the package, computer, and operator can be added to the data flow. Microsoft SQL Server 2008 Integration Services (SSIS) includes system variables that provide this information".
- "You configure the Audit transformation by providing the name of a new output column to add to the transformation output, and then mapping the system variable to the output column. You can map a single system variable to multiple columns".
- "This transformation has one input and one output. It does not support an "error output"".

Copyright © Capgemini 2010. All Rights Reserved.

Audit Transformation

- EXAMPLE: The audit Fields are used to describe general information such as, Package Name, Machine Name, User Name, Package Id, Version Id etc.
- The output shown here includes few of the audit fields after Audit Field Transformation.

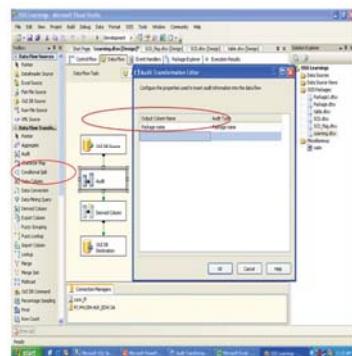
Employee Id	First Name	Package name	Execution insta...	Machine name	User name
10019	Bobb	Package1	(74ED1227-94...	PC-P41213	PATNlgoretouch
10023	Bruce	Package1	(74ED1227-94...	PC-P41213	PATNlgoretouch
10024	Bruce	Package1	(74ED1227-94...	PC-P41213	PATNlgoretouch
10025	Butch	Package1	(74ED1227-94...	PC-P41213	PATNlgoretouch
10026	Calla	Package1	(74ED1227-94...	PC-P41213	PATNlgoretouch
10027	Carol	Package1	(74ED1227-94...	PC-P41213	PATNlgoretouch
10028	Carol	Package1	(74ED1227-94...	PC-P41213	PATNlgoretouch
10029	Cheri	Package1	(74ED1227-94...	PC-P41213	PATNlgoretouch
10030	Chuck	Package1	(74ED1227-94...	PC-P41213	PATNlgoretouch
10031	Chuck	Package1	(74ED1227-94...	PC-P41213	PATNlgoretouch
10032	Chuck	Package1	(74ED1227-94...	PC-P41213	PATNlgoretouch
10034	Coleen	Package1	(74ED1227-94...	PC-P41213	PATNlgoretouch
10035	Conne	Package1	(74ED1227-94...	PC-P41213	PATNlgoretouch
10036	Conne	Package1	(74ED1227-94...	PC-P41213	PATNlgoretouch



Copyright © Capgemini 2012. All Rights Reserved.

Audit Transformation

- Drag a 'OLE DB Source' from Data Flow Task Navigator to Data Flow Task Window.
- Set the 'Connection Manager' by giving appropriate information, e.g. Connection Manager Name, Table or Query, Table Name.
- Drag the Audit transformation from Data Flow Transformation Navigation Toolbox.
- Select the output column name and audit field from and click 'OK' eg. Package Name.



Audit Transformation

- Drag the 'OLE DB Destination' select the proper connection and map the columns properly.
- The Audit column – package name will also be mapped.
- Derived Column transformation is used between Audit and OL DB destination transformation to cast the Package Name, Data type – Unicode String[DT_WSTR] to Data type – Non-Unicode, as mentioned in the target table structure.



Copyright © Capgemini 2010. All Rights Reserved.

Audit Transformation

- SYSTEM VARIABLES:
- ExecutionInstanceGUID:
 - The GUID that identifies the execution instance of the package.
- PackageID:
 - The unique identifier of the package.
- PackageName:
 - The package name.
- VersionID:
 - The version of the package.
- ExecutionStartTime:
 - The time the package started to run.

Copyright © Capgemini 2010. All Rights Reserved

Audit Transformation

- **MachineName:**
The computer name.
- **UserName**
The login name of the person who started the package.
- **TaskName**
The name of the Data Flow task with which the Audit transformation is associated.
- **TaskId**
The unique identifier of the Data Flow task.

Copyright © Capgemini 2010. All Rights Reserved

Cache Transformation

Lesson 10:

Cache Transformation

- "The Cache transformation is a brand new feature in SQL 2008, allowing you to cache the data used in the Lookup transform. The Lookup transform can then utilize this cached data to perform the lookup operation "
- We can configure the component to cache the lookup dataset, rather than retrieving the data on a per row basis. In SSIS 2008, your caching options for performing lookup operations have been extended through the Cache transformation and Cache connection manager .

Copyright © Capgemini 2010. All Rights Reserved

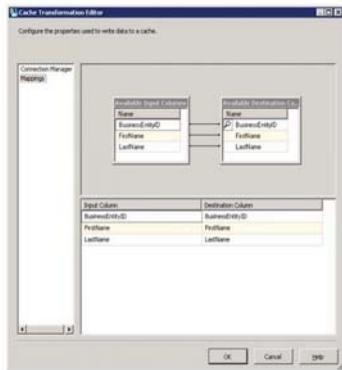
Cache Transformation

- By using the new transformation and connection manager, you can cache lookup data from any type of data source (not only an OLE DB source), persist the data to the memory cache or into a cache file on your hard drive, and use the data in multiple data flows or packages.
- The primary purpose of the Cache transformation is to persist data through a Cache connection manager. When configuring the transformation, you must, in addition to specifying the connection manager, define the column mappings.

Copyright © Capgemini 2010. All Rights Reserved

Cache Transformation

- The following figure shows the Mappings page of the Cache Transformation Editor.

Copyright © Capgemini 2007. All Rights Reserved

Cache Transformation

Cache connection manager

- To support cached lookups, you must, along with configuring the Cache transformation, configure the Cache connection manager. The following figure shows the General tab of the Connection Manager Editor.

Copyright © Capgemini 2007. All Rights Reserved

Cache Transformation

Best practices

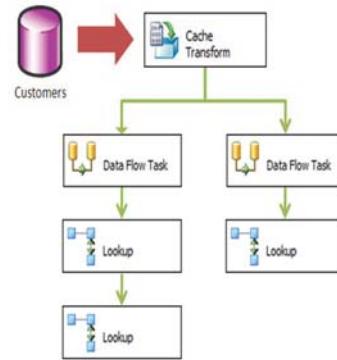
- Reuse the cache to reduce database load
- Share the cache between lookups to reduce memory usage
- Using the CCM is not always faster than OLEDB - the cost of disk access can out weight the benefits of pre-creating the cache
- The cache is essentially clear text - do not store sensitive data inside of the cache
- In terms of Cache Modes(Full,Partial,No Cache) and the best practices that surround them, using a cache connection manager is equivalent to using a Full Cache mode

Copyright © Capgemini 2010. All Rights Reserved

Cache Transformation

Reducing database and memory usage

- If your reference database is remote, or under heavy load, consider using the Cache Connection Manager instead of an OLEDB connection.
- Once a cache is used (or created) in an SSIS package, it will be kept in memory until the package has finished executing. The cache can be reused across multiple data flows, and shared between multiple lookups in the same data flow. It can also be persisted to disk, and reused across package executions.



Character Map Transformation

Lesson 11:

Character Map Transformation

- "The Character Map transformation applies string functions, such as conversion from lowercase to uppercase, to character data. This transformation operates only on column data with a string data type."
- This is a passive transformation.
- You configure the Character Map transformation in the following ways
 - Specify the columns to convert.
 - Specify the operations to apply to each column.
- The Character Map transformation can convert column data in place or add a column to the transformation output and put the converted data in the new column. You can apply different sets of mapping operations to the same input column and put the results in different columns.

Copyright © Capgemini 2010. All Rights Reserved

Character Map Transformation

- The following table describes the mapping operations that the Character Map transformation supports.

Operation	Description
Byte reversal	Reverses byte order.
Full width	Maps half-width characters to full-width characters.
Half width	Maps full-width characters to half-width characters.
Hiragana	Maps katakana characters to hiragana characters.
Katakana	Maps hiragana characters to katakana characters.
Linguistic casing	Applies linguistic casing instead of the system rules. It refers to functionality provided by the Win32 API for Unicode simple case mapping of Turkic and other locales.
Lowercase	Converts characters to lowercase.
Simplified Chinese	Maps traditional Chinese characters to simplified Chinese characters.
Traditional Chinese	Maps simplified Chinese characters to traditional Chinese characters.
Uppercase	Converts characters to uppercase.

Copyright © Capgemini 2010. All Rights Reserved

Character Map Transformation

- More than one operation can be performed in a transformation. However, some mapping operations are mutually exclusive. The following table lists restrictions that apply when you use multiple operations on the same column. Operations in the columns Operation A and Operation B are mutually exclusive.

Operation A	Operation B
Lowercase	Uppercase
Hiragana	Katakana
Half width	Full width
Traditional Chinese	Simplified Chinese
Lowercase	Hiragana, katakana, half width, full width
Uppercase	Hiragana, katakana, half width, full width

- Use the Character Map Transformation Editor dialog box to select the string functions to apply to column data and to specify whether mapping is an in-place change or added as a new column.

Copyright © Capgemini 2010. All Rights Reserved

Character Map Transformation

Example:

- The following example is showing a data set comprising of 10 rows. Suppose now requirement is that all characters of the column 'First Name' should be in Uppercase. For that we have to use Character Map Transformation to convert the characters to uppercase.

Account No	First Name	Last Name	Company	Address	Cty
"10019"	"bobbi"	"Ardi"	"Market Place"	"1000 5 Nickel..."	"Sanetown", "T...
"10023"	"bruce"	"Beecher"	"Madison & Hu..."	"1037 W Walco..."	"Smithville", "A...
"10024"	"bruce"	"Beyer"	"La Salle Clinic"	"108 E Wilson..."	"Jonestown", "...
"10025"	"butch"	"Bobbs"	"Town & Countr..."	"108 Hileck Ct"	"Smithville", "A...
"10026"	"collie"	"Boshers"	"Saturn of App..."	"110 Fox River..."	"Smithville", "A...
"10027"	"carol"	"Braun"	"Bensis Corp."	"110 W North ..."	"Jonestown", "...
"10028"	"cardi"	"Braun"	"AAL Member ..."	"1115 E Glend..."	"Smithville", "A...
"10029"	"chen"	"Buksak"	"Office Support"	"1122 Milwaukee..."	"Jonestown", "...
"10030"	"chuck"	"Buss"	"EAA"	"1134 S Franklin..."	"Oreton", "Ak...
"10031"	"chuck"	"Carpenter"	"Kurz Electric"	"115 S Drew St"	"Smithville", "A...

Copyright © Capgemini 2010. All Rights Reserved

Character Map Transformation

- All the characters of the column 'First Name' is converted from lowercase to uppercase in the O/P data set after Character Map Transformation is done.
- Output::

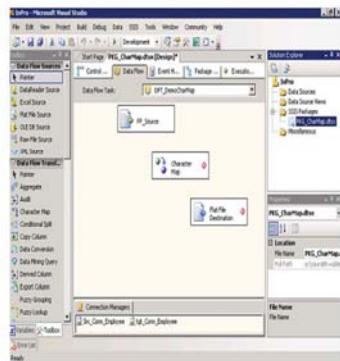
Account No	First Name	Last Name	Company	Address	City
"10019"	"BOBET"	"Andi"	"Market Place"	"1000 S Nicol...	"Sametown", "T...
"10023"	"BRUCE"	"Beecher"	"Madison & Huff...	"1037 W Wilson...	"Smithville", "A...
"10024"	"BRUCE"	"Beyer"	"Le Salle Circ..."	"108 E Wilson...	"Jonestown", "...
"10025"	"BUCHANAN"	"Bobbi"	"Town & Count...	"108 Middle Cr...	"Smithville", "A...
"10026"	"CALLA"	"Bohner"	"Saturn of App..."	"110 Fox River...	"Smithville", "...
"10027"	"CAROL"	"Braser"	"Bensis Corp."	"110 W North ...	"Jonestown", "...
"10028"	"CAROL"	"Brun"	"AAI Member ..."	"1115 E Gind...	"Smithville", "A...
"10029"	"CHERI"	"Buaya"	"Office Support"	"1122 Minnau...	"Jonestown", "...
"10030"	"CHUCK"	"Bush"	"EAA"	"1134 S Franklin...	"Overton", "AK...
"10031"	"CHUCK"	"Carpenter"	"Kurz Electric"	"115 S Drew St"	"Smithville", "A...



Copyright © Capgemini 2007. All Rights Reserved 47

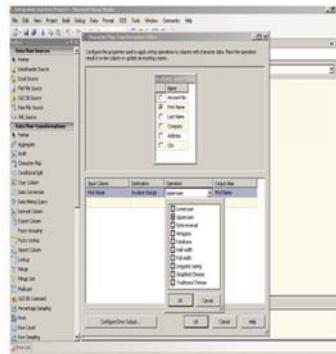
Character Map Transformation

- Drag flat file Source, character map Transformation and flat file Destination as shown
- Configure the FF_Source with Src_Conn_Employee connection.
- Link output of FF_Source to character map transformation.



Character Map Transformation

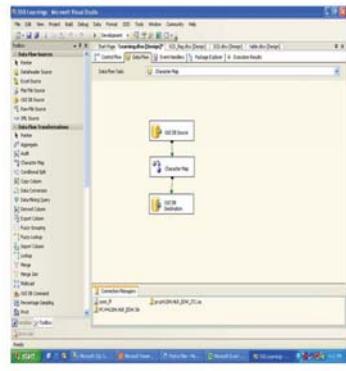
- Double click to open transformation editor as shown.
- Select columns, selected column are automatically added below
- Select the operation you want to be performed.
- Input Column: for selecting column.
- Destination : either a new column or In place change.
- Operation: you can specify operation/s on column.
- Output Alias: for supplying either new column name or
- Alias name to input column.
- Other column redirected as it is.



Copyright © Capgemini 2007. All Rights Reserved.

Character Map Transformation

- Click ok, and connect the output of transformation to Flat File Destination.
- Double Click to configure the Flat file destination.
- Add the connection.
- Map the columns of source and destination.
- Run the data flow and check the output



Conditional Split Transformation

Lesson 12:

Conditional Split Transformation

- The Conditional Split transformation evaluates expressions, and based on the results, directs the data row to the specified output.
- This transformation also provides a default output, so that if a row matches no expression it is directed to the default output.
- Provide an expression that evaluates to a Boolean for each condition you want the transformation to test.



Copyright © Capgemini 2010. All Rights Reserved

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

Conditional Split Transformation

- Specify the order in which the conditions are evaluated. Order is significant, because a row is sent to the output corresponding to the first condition that evaluates to true.
- Specify the default output for the transformation. The transformation requires that a default output be specified.

- **EXAMPLE**

Suppose in our scenario we want the employees working for USA or for IND. Here I will put condition as

case 1 CITY=="USA" And

case 2 CITY=="IND"

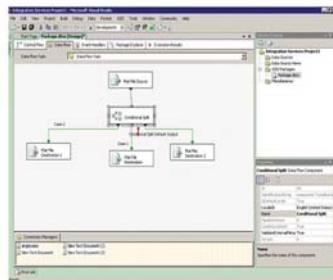
to split the records into two groups. One default group is also created automatically to store unmatched, like CITY other then USA or IND.



Copyright © Capgemini 2010. All Rights Reserved.

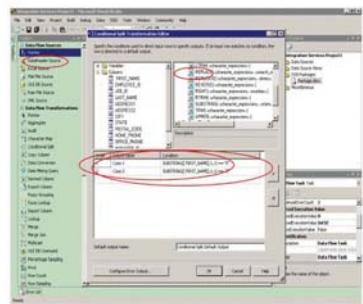
Conditional Split Transformation

- Drag a 'Conditional split' transformation from Data Flow Transformations & drop it in to work area & connect it with the flat file source.
- Double Click or Right Click on Data Flow transformation and select edit to open 'Conditional Split Transformation Editor' window



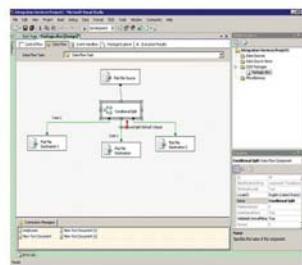
Conditional Split Transformation

- For example, the following conditions direct any rows in the FirstName column that begin with the letter "K" to one output, rows that begin with the letter "J" to a different output, and all other rows to the default output

Copyright © Capgemini 2007. All Rights Reserved.

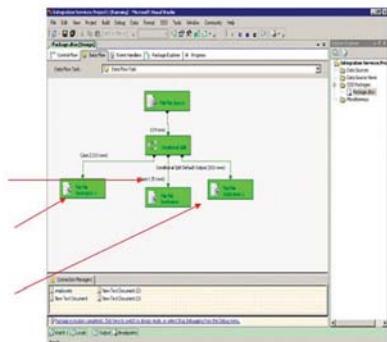
Conditional Split Transformation

- Depending upon the conditions set the conditional split transformation splits the data into 3 different flat files.
- Three Flat file destinations are connected with the conditional split & corresponding connections for the flat files are done in the connection manager.

Copyright © Capgemini 2010. All Rights Reserved

Conditional Split Transformation

- Start the debugging process and carefully notice result. It is seen that conditional split transformation splits the data depending upon the conditions following conditions.
- Case1: First Name column that begin with the letter "K".(5 rows)
- Case2: First Name column that begin with the letter "J".(13 rows)
- Conditional split default output is the o/p having the remaining data.

Copyright © Capgemini 2007. All Rights Reserved.

Conditional Split Transformation

- Input Source

EMPLOYEE		
First_Name	Last_Name	City_Code
Kashav	ganguly	452001
sachin	tendulkar	400045
vaibhav	dravid	300016
Jai	prakash	400456
Jai	chand	700342
Kishor	kumar	676004

- Case 1 Output

Kashav	ganguly	452001
Kishor	kumar	676004

- Case 2 Output

Jai	prakash	400456
Jai	chand	700342

- Default Output

sachin	tendulkar	400045
vaibhav	dravid	300016

Copyright © Capgemini 2012. All Rights Reserved.

Copy Column Transformation

Lesson 13:

Copy Column Transformation

- The copy column transformation creates new columns by copying input columns and adding the new columns to the transformation output. Later in the data flow, different transformations can be applied to the column copies.
- For example, you can use the copy column transformation to create a copy of a column and then convert the copied data to uppercase characters by using the character map transformation, or apply aggregations to the new column by using the aggregate transformation.



Copyright © Capgemini 2010. All Rights Reserved

Copy Column Transformation

Example::

- The following example is showing a data set comprising of 6 columns. Say now requirement is that we have to create a copy of a column and then convert the copied data to uppercase characters by using the Character Map Transformation. We can achieve this by using Copy Column Transformation.

Account No	First Name	Last Name	Company	Address	City
"10019"	"bobbi"	"Andi"	"Market Place"	"1000 S Nicole...	"Smethville"
"10023"	"bruce"	"Beecher"	"Madison & Hull..."	"1037 W Wilson...	"Smethville"
"10024"	"bruce"	"Beyer"	"La Salle Circa"	"108 E Wilson...	"Jonesborough"
"10025"	"butch"	"Bobbi"	"Town & Count...	"108 E Nolock Ct"	"Smethville"



Copyright © Capgemini 2010. All Rights Reserved

Copy Column Transformation

- Output::

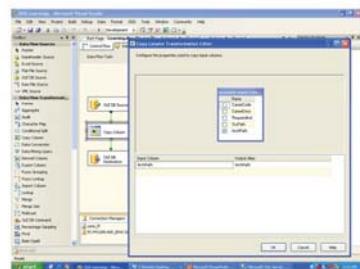
- Output Data set will be as follows comprising of 7 columns which includes newly added company alias column which is nothing but a copy of company column.

Account No	First Name	Last Name	Company	Address	Ctry	Company alias
"10019"	"bobbi"	"Andie"	"Market Place"	"1000 S Nicolet..."	"Sometown"	"Market Place"
"10023"	"bruce"	"Beecher"	"Madison & Huth Communication Co"	"1027 W Wisconsin..."	"Steatville"	"Madison & Huth Communication Co"
"10024"	"bruce"	"Beyer"	"La Sale Clinic"	"108 E Wisconsin..."	"Jonestown"	"La Sale Clinic"
"10025"	"butch"	"Bobby"	"Town & Country Electric Inc."	"108 Hilltop Ct"	"Steatville"	"Town & Country Electric Inc."

Copyright © Capgemini 2010. All Rights Reserved

Copy Column Transformation

- Drag Copy Column Transformation from Data Flow Transformations.
- Link it with source.
- Double click on Copy Column Transformation to open copy column Transformation Editor.
- The column to be copied here is ArchPath. The column will be accordingly mapped in the target table.



Copyright © Capgemini 2010. All Rights Reserved

Copy Column Transformation

Options Available:

- Available Input Columns
- Select columns to copy by using the check boxes. Your selections add input columns to the table below.
- Input Column
- Select columns to copy from the list of available input columns. Your selections are reflected in the check box selections in the Available Input Columns table.
- Output Alias
- Type an alias for each new output column. The default is Copy of, followed by the name of the input column; however, you can choose any unique, descriptive name.

Copyright © Capgemini 2010. All Rights Reserved

Data Conversion Transformation

Lesson 14:

Data Conversion Transformation

- "The Data Conversion transformation converts the data in an input column to a different data type and then copies it to a new output column."
- For example, a package can extract data from multiple sources, and then use this transformation to convert columns to the data type required by the destination data store. You can apply multiple conversions to a single input column.
- You can configure the Data Conversion transformation in the following ways:
 - Specify the columns that contain the data to convert and the types of data conversions to perform.
 - Specify whether the transformation output columns use the quicker, but locale-insensitive, fast parsing routines that Microsoft SQL Server 2008 Integration Services (SSIS) provides or the standard locale-sensitive parsing routines.

Copyright © Capgemini 2007. All Rights Reserved.

Data Conversion Transformation

- Example: The Data before Data Conversion is shown below. The column named Expenses contains decimal values.
- We need to have integer values in Expenses Column. In this case we choose Data Conversion Transformation for converting decimal values into integer values.

Employee ID	Expenses	Description	Account No
44894	001.11	INTL	111
44895	0011.9	ALBANY	119
44896	0018.8	ALBUQUERQUE	120
44897	5.6626	HONOLULU	126
44898	005.27	HILO	127
44899	07.128	KONA	128
44900	89.129	MAUI	129
44901	080.31	IRV	131
44902	7684.8	ANCHORAGE	140
44903	7645.5	APPLETON	150
44904	077.53	CHERRY HILL	153

Copyright © Capgemini 2010. All Rights Reserved

Data Conversion Transformation

- The Data after conversion will be as shown. All the decimal values in the Expenses column are now converted into Integer values.

Employee ID	Rounded Off Expenses	Description	Account No
44894	1	INTL	111
44895	12	ALBANY	119
44896	19	ALBUQUERQUE	120
44897	6	HONOLULU	126
44898	5	HBO	127
44899	7	KONA	128
44900	89	MAUI	129
44901	80	MC	131
44902	7685	ANCHORAGE	140
44903	7646	APPLETON	150
44904	78	CHERRY HILL	153



Copyright © Capgemini 2010. All Rights Reserved

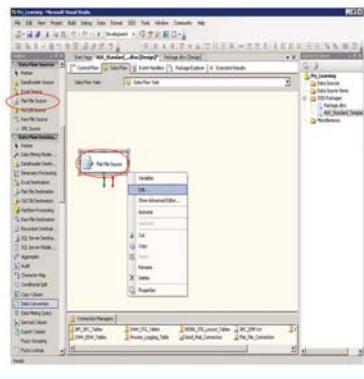
Data Conversion Transformation

- Drag a 'Flat File Source' from Data Flow Source.

Data in Flat File

1	Amit	200
2	Disha	400
3	Mohit	350
4	Xavier	500

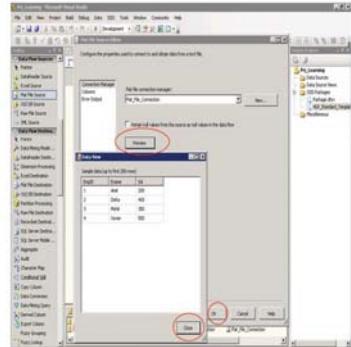
- Double click on 'Flat File Source' or Right Click on 'Flat File Source' and select 'Edit'
- Make appropriate change in Flat File.(e.g. Choose Flat File Connection, Select a File, Rename Column Name...etc.)



Copyright © Capgemini 2010. All Rights Reserved.

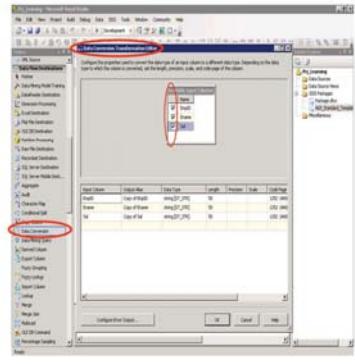
Data Conversion Transformation

- On Clicking Preview button, Data View window will be open.
- Click 'Close and Then 'OK' button of Flat File Source Editor.

Copyright © Capgemini 2007. All Rights Reserved.

Data Conversion Transformation

- Drag a 'Data Conversion' transformation from Data Flow Destinations Navigation & drop in to work area.
- Double Click or Right Click on Data Flow transformation and select edit to open 'Data Conversion Transformation Editor' window.
- Select required columns from 'Available Input Columns'.



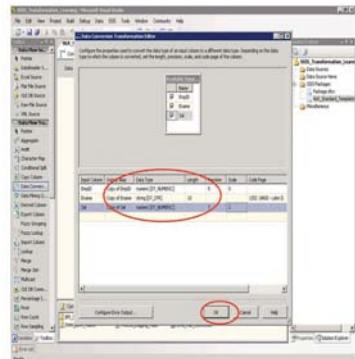
Capgemini
INNOVATING THROUGH INTEGRATION

Copyright © Capgemini 2010. All Rights Reserved

Data Conversion Transformation

- Change 'Data Type', 'Length',....of required columns.

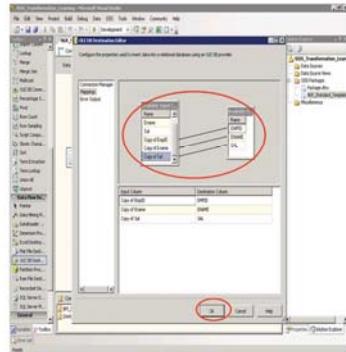
- Click 'OK'



Copyright © Capgemini 2007. All Rights Reserved

Data Conversion Transformation

- Drag a 'OLE DB Destination' from Data Flow Destination and drop it in work area, and make a connection between 'Data Conversion Transformation' and 'OLE DB Destination'.
- Edit 'OLE DB Destination' and make proper connection in 'Connection Manager' (e.g. Set Target Database, Target Table.).
- In 'Mappings' tab make proper column mapping between 'Available Input Columns' and 'Available Output Columns'.
- Click 'OK'.



Derived Column Transformation

Lesson 15:

Derived Column Transformation

"The Derived Column Transformation creates new columns values by applying expressions to transformation input columns."

The Derived Column can perform following tasks :

- Concatenate data from different column into derived column.
- Extract characters from string data by using functions like SUBSTRING, and then store result in a derived column.
- Apply mathematical functions to numeric data and store the result in the derived column.
- Create expressions that compare input columns and variables.
- Extract part of date time value.



Copyright © Capgemini 2010. All Rights Reserved

Derived Column Transformation

- EXAMPLE: The data shown below is in fixed width format. Columns 'First Name' & 'Last Name' both are of length 10.
- Here, we can trim those fields in order to avoid blank spaces to be inserted into target.
- Also we can insert some value incase the input coming is NULL.

Account No	First Name	Last Name	Company
10019	Bobbi	Arndt	Market Place
10023	Bruce	Beecher	Madison & Hult Communication Co
10024	Bruce	Beyer	La Salle Clinic
10025	Butch	Bobbi	Town & Country Electric Inc.
10026	Calla	Bosheris	Sabre of Appleton
10027	Carol	Brauer	Benes Corp.
10028	Carol	Braun	AAL Member Credit Union
10029	Cheri	Bulayik	Office Support
10030	Chuck	Buss	EAA
10031	Chuck	Carpenter	Kurz Electric
10032	Chuck	Carr	Alpha I
10034	Colleen	Casperon	Valley Trust Corporation
10035	Connie	Caterton	Fox Community Credit Union
10036	Connie	Clay	Valley Lawn Care

Copyright © Capgemini 2010. All Rights Reserved

Derived Column Transformation

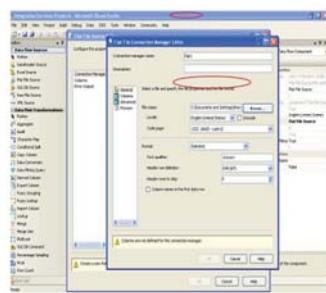
- Modified Date is current system date & Modified User is current system user.
- The following output contains two newly Derived columns called 'Modified Date' & 'Modified User'.
- Fields 'First Name' & 'Last Name' are trimmed where unnecessary spaces are removed.

Account No.	First Name	Last Name	Company	ModifiedDate	Modified User
10011	Babbie	Amit	Market Place	1/8/2007 3:04:03 PM	PAITHognathu
10023	Bruck	Beebe	Madden & Hahn Com...	1/8/2007 3:04:03 PM	PAITHognathu
10024	Bruck	Beyer	Le Salle Cinc...	1/8/2007 3:04:03 PM	PAITHognathu
10025	Bulich	Bobbie	Town & Country Elec...	1/8/2007 3:04:03 PM	PAITHognathu
10026	Cain	Bennie	Wingtip Computer Applian...	1/8/2007 3:04:03 PM	PAITHognathu
10027	Caril	Braun	Bennix Com...	1/8/2007 3:04:03 PM	PAITHognathu
10028	Carol	Braun	AAA Member Credit U...	1/8/2007 3:04:03 PM	PAITHognathu
10029	Chen	Bruayah	Office Support	1/8/2007 3:04:03 PM	PAITHognathu
10030	Chuck	Burns	EAI	1/8/2007 3:04:03 PM	PAITHognathu
10031	Chuck	Carpenter	Kurt Electric	1/8/2007 3:04:03 PM	PAITHognathu
10032	Colleen	Carter	Logan's	1/8/2007 3:04:03 PM	PAITHognathu
10034	Colleen	Casperian	Valley Trout Corporat...	1/8/2007 3:04:03 PM	PAITHognathu
10035	Corine	Catheron	Fox Community Credit...	1/8/2007 3:04:03 PM	PAITHognathu

Copyright © Capgemini 2007. All Rights Reserved

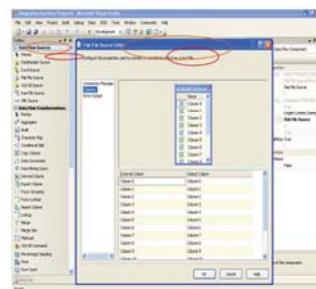
Derived Column Transformation

- Drag a 'Flat File Source' from the Data Flow Task Navigator into Data Flow Task Window.
- Double click or Right click on 'Flat File Source' to open 'Flat File Source Editor'
- Set the 'Connection Manager' by giving appropriate information, e.g. Connection Manager name, File Name.

Copyright © Capgemini 2007. All Rights Reserved

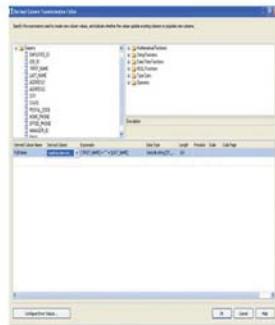
Derived Column Transformation

- Select Columns and check required 'Available External Column'.
- Click 'OK'
- Drag 'Derived Column Transformation' from Data Flow Transformation into Data Flow Task Window and connect Flat File Source to Derived Column Transformation.



Derived Column Transformation

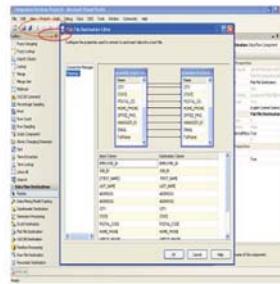
- Double Click or Right Click on the Derived Column Transformation to open the 'Derived Column Transformation Editor'.
- Give Derived Column Name, in Derived Column select 'add as new column', in Expression add the expression which is to be derived, for example we want to concatenate First Name and Last Name of the Employee Table and store it under Separate column 'FULL NAME'.
- Click 'OK'



Copyright © Capgemini 2007. All Rights Reserved

Derived Column Transformation

- Drag 'Flat File Destination' from the Data Flow Destinations into Data Flow Task Window and connect Derived Column output to the Flat File Destination.
- Set the Connection Manager as explained previously.
- Create the required mapping.
- Click 'OK'



Copyright © Capgemini 2010. All Rights Reserved.

SQL Server Integration Services (SSIS) Training Kit(Part 3)

Lesson 16: Export Column
Transformation

Export Column Transformation

- The Export Column transformation reads data in a data flow and inserts the data into a file.
- The data to be written must have a DT_TEXT, DT_NTEXT, or DT_IMAGE data type.
- You can configure the Export Column transformation in the following ways:
 - Specify the data columns and the columns that contain the path of files to which to write the data.
 - Specify whether the data-insertion operation appends or truncates existing files.
 - Specify whether a byte-order mark (BOM) is written to the file.
- Note:A BOM is written only when the data is not appended to an existing file and the data has the DT_NTEXT data type.

Copyright © Capgemini 2010. All Rights Reserved.

Export Column Transformation

- EXAMPLE: The Data below shown has POLICYNUMBER column(type should be dt_text,dt_ntext or dt_image) which has to be exported to the path mentioned in column called PLANNAME.

POLICYNUMBER	CARRIERCODE	POLICYSTATUS	PLANNAME
POL1	CAR3	3	D:\TUSHAR\45895 on pc-p41284\assignments_Files\new1.txt
POL1	CAR1	4	D:\TUSHAR\45895 on pc-p41284\assignments_Files\new.txt
POL2	CAR2	5	D:\TUSHAR\45895 on pc-p41284\assignments_Files\new2.txt

1st record will be exported to new1.txt

2nd record will be exported to new.txt

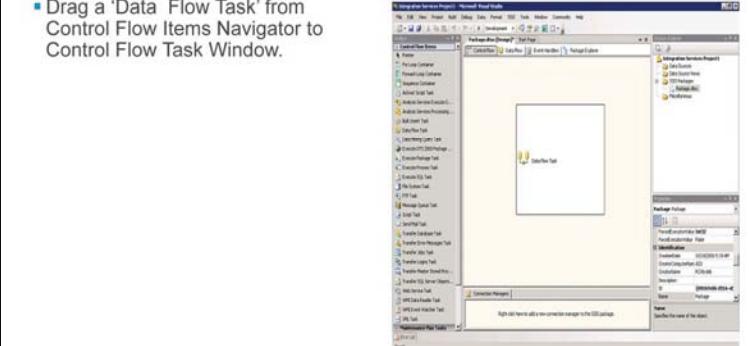
3rd record will be exported to new2.txt



Copyright © Capgemini 2010. All Rights Reserved.

Export Column Transformation

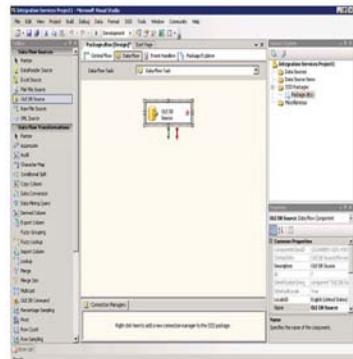
Drag a 'Data Flow Task' from Control Flow Items Navigator to Control Flow Task Window.



Capgemini
INFORMATION TECHNOLOGY CONSULTANTS

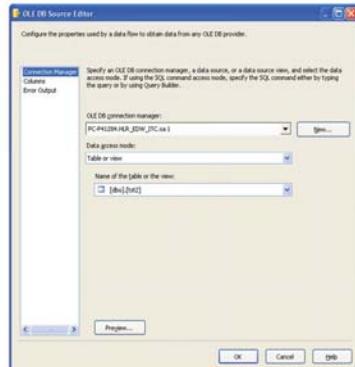
Export Column Transformation

- Drag a 'OLE DB Source' from Data Flow Task Navigator to Data Flow Task Window.



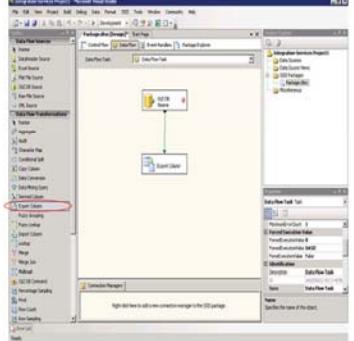
Export Column Transformation

- 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.



Export Column Transformation

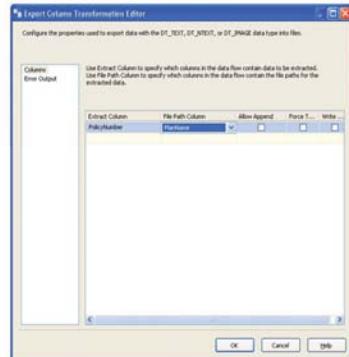
- Drag 'Export Column' transformation from Data Flow Transformation Navigation Toolbox to the data flow task window.
- Link 'Export Column Transformation' with 'OLE DB Source'.



Capgemini
connecting people through technology

Export Column Transformation

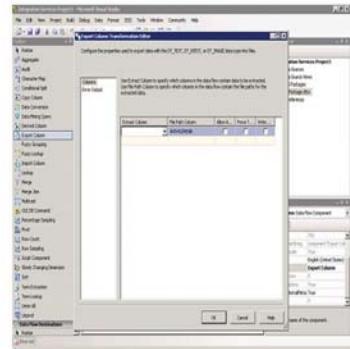
- Double Click on 'Export Column transformation' or Right click and click 'Edit'.
- Properly set the fields 'extract column' and 'file path column' and others.
- As per previous example set 'policynumber' as extract column and file path column as 'planname'



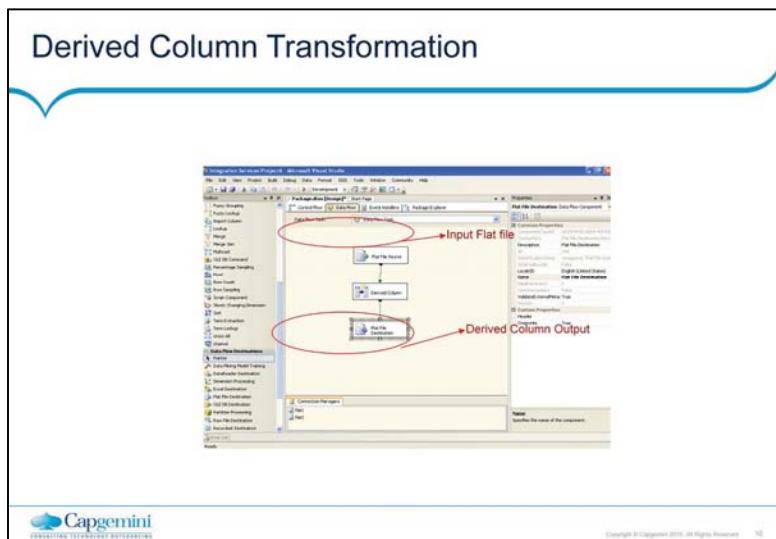
Copyright © Capgemini 2010. All Rights Reserved.

Export Column Transformation

- Allow Append property if checked will append the data to the same file on every load
- Force Truncate property will truncate the data for every load.
- Click Ok
- Execute the dataflow.



Copyright © Capgemini 2007. All Rights Reserved.



Fuzzy Grouping Transformation

Lesson 17:

Fuzzy Grouping Transformation

- The transformation output includes all input columns, one or more columns with standardized data, and a column that contains the similarity score. The score is a decimal value between 0 and 1. The canonical row has a score of 1. Other rows in the fuzzy group have scores that indicate how well the row matches the canonical row. The closer the score is to 1, the more closely the row matches the canonical row. If the fuzzy group includes rows that are exact duplicates of the canonical row, these rows also have a score of 1. The transformation does not remove duplicate rows; it groups them by creating a key that relates the canonical row to similar rows.

Fuzzy Grouping Transformation

The transformation produces one output row for each input row, with the following additional columns:

- `_key_in` : a column that uniquely identifies each row.
- `_key_out` : a column that identifies a group of duplicate rows. The `_key_out` column has the value of the `_key_in` column in the canonical data row. Rows with the same value in `_key_out` are part of the same group. The `_key_out` value for a group corresponds to the value of `_key_in` in the canonical data row.
- `_score` : a value between 0 and 1 that indicates the similarity of the input row to the canonical row.

These are the default column names and you can configure the Fuzzy Grouping transformation to use other names. The output also provides a similarity score for each column that participates in a fuzzy grouping



Copyright © Capgemini 2010. All Rights Reserved.

Fuzzy Grouping Transformation

"The Fuzzy Grouping transformation performs data cleaning tasks by identifying rows of data that are likely to be duplicates and selecting a canonical row of data to use in standardizing the data."

Types of match

1. Exact Match: An exact match guarantees that only rows that have identical values in that column will be grouped. Exact matching can be applied to columns of any Integration Services data type except DT_TEXT, DT_NTEXT, and DT_IMAGE.
2. Fuzzy match: A fuzzy match groups rows that have approximately the same values. The method for approximate matching of data is based on a user-specified similarity score. Only columns with the DT_WSTR and DT_STR data types can be used in fuzzy matching.



Copyright © Capgemini 2010. All Rights Reserved. 12

Fuzzy Grouping Transformation

EXAMPLE

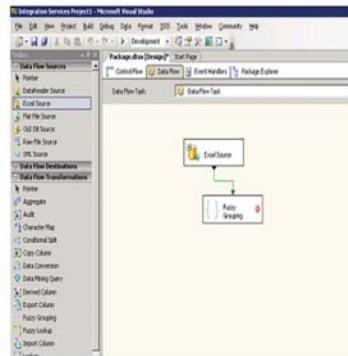
Suppose that the entries into the system are done manually across many countries and the CITY 'AUSTRALIA' is saved with many similes example 'AUSTRILIA', 'USSTRAILIA' or 'AUSTERILYA' etc. these are need to grouped to correct one. The transformation generate the value accordance with the similarity between the wrong one and right one, Based on acceptance criteria like .70 and above is treated as OR grouped to 'AUSTRALIA'



Copyright © Capgemini 2010. All Rights Reserved. 19

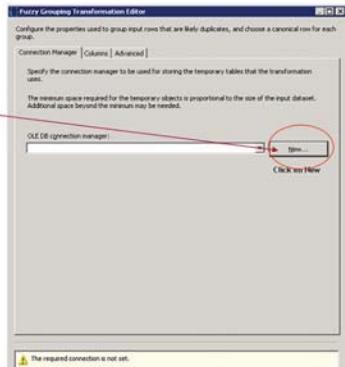
Fuzzy Grouping Transformation

- From the Toolbox, drag the Fuzzy Grouping transformation to the design surface.
 - Connect the Fuzzy Grouping transformation to the data flow by dragging the connector—the green or red arrow—from the data source or a previous transformation to the Fuzzy Grouping transformation.
 - Double-click the Fuzzy Grouping transformation.



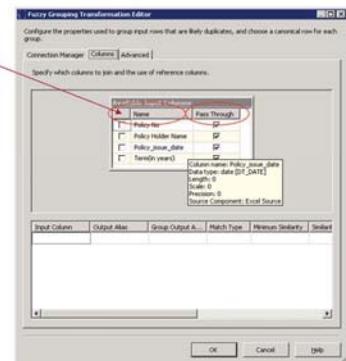
Fuzzy Grouping Transformation

- In the Fuzzy Grouping Transformation Editor dialog box, on the Connection Manager tab, select an OLE DB connection manager that connects to a SQL Server 2008 database. This will be used by SSIS for storing temporary tables that the transformation uses.
- Select a OLE DB connection



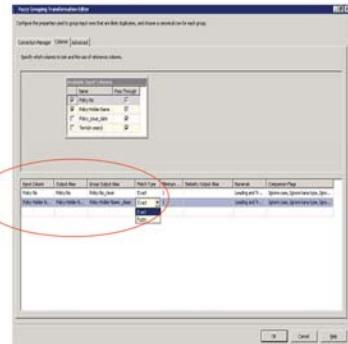
Fuzzy Grouping Transformation

- Click the Columns tab and, in the Available Input Columns list, select the check box of the input columns to use to identify similar rows in the dataset.
- Select the check box in the Pass Through column to identify the input columns to pass through to the transformation output. Pass-through columns are not included in the process of identification of duplicate rows.

Copyright © Capgemini 2010. All Rights Reserved

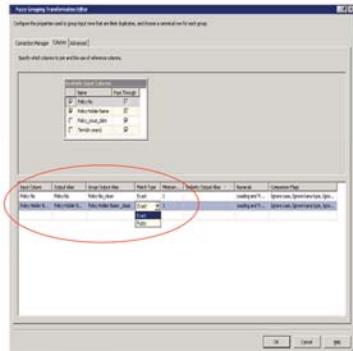
Fuzzy Grouping Transformation

- Optionally, update the names of output columns in the Output Alias column.
- Optionally, update the names of cleaned columns in the Group Output Alias column.
- Optionally, update the type of match to use in the Match Type column.
- Note: At least one column must use fuzzy matching.



Fuzzy Grouping Transformation

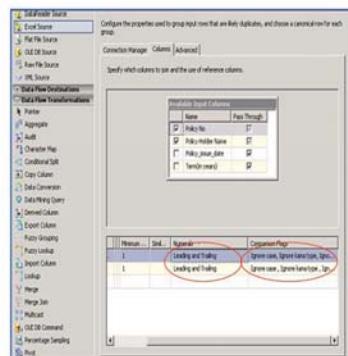
- Specify the minimum similarity level columns in the Minimum Similarity column. The value must be between 0 and 1. The closer the value is to 1, the more similar the values in the input columns must be to form a group. A minimum similarity of 1 indicates an exact match.
- Optionally, update the names of similarity columns in the Similarity Output Alias column



Copyright © Capgemini 2010. All Rights Reserved.

Fuzzy Grouping Transformation

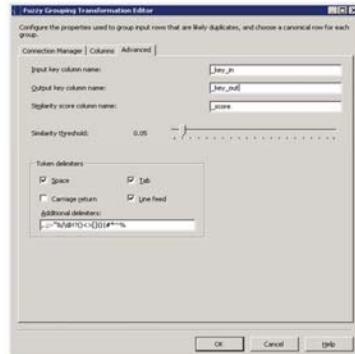
- To specify the handling of numbers in data values, update the values in the Numerals column.
- To specify how the transformation compares the string data in a column, modify the default selection of comparison options in the Comparison Flags column.
- Click 'Advanced tab'.



Copyright © Capgemini 2010. All Rights Reserved.

Fuzzy Grouping Transformation

- Click the Advanced tab to modify the names of the columns that the transformation adds to the output for the unique row identifier (`_key_in`), the duplicate row identifier (`_key_out`), and the similarity value (`_score`).
- Optionally, adjust the similarity threshold by moving the slider bar. The similarity threshold indicates how strictly the transformation identifies duplicates. The similarity thresholds can be set at the component and the column levels. The column-level similarity threshold is only available to columns that perform a fuzzy match.



Copyright © Capgemini 2010. All Rights Reserved.

Fuzzy Grouping Transformation

- The similarity range is 0 to 1. The closer to 1 the threshold is, the more similar the rows and columns must be to qualify as duplicates. You specify the similarity threshold among rows and columns by setting the Min Similarity property at the component and column levels. To satisfy the similarity that is specified at the component level, all rows must have a similarity across all columns that is greater than or equal to the similarity threshold that is specified at the component level.
- Optionally, clear the token delimiter check boxes to ignore delimiters in the data, and Click OK.



Fuzzy Lookup Transformation

Lesson 18:

Fuzzy Lookup Transformation

- The Fuzzy Lookup transformation performs data cleaning tasks such as standardizing data, correcting data, and providing missing values.
- This transformation differs from the Lookup transformation in its use of fuzzy matching. The Lookup transformation uses an equi-join to locate matching records in the reference table. It returns either an exact match or nothing from the reference table. In contrast, the Fuzzy Lookup transformation uses fuzzy matching to return one or more close matches from the reference table.
- This transformation has one input and one output.
- Only input columns with the DT_WSTR and DT_STR data types can be used in fuzzy matching. Exact matching can use any DTS data type except DT_TEXT, DT_NTEXT, and DT_IMAGE.

Copyright © Capgemini 2010. All Rights Reserved.

Fuzzy Lookup Transformation

- Columns that participate in the join between the input and the reference table must have compatible data types. For example, it is valid to join a column with the DT_WSTR data type to a column with the SQL Server nvarchar data type, but invalid to join a column with the DT_WSTR data type to a column with the int data type.
- **EXAMPLE**
- Suppose that the entries into the system are done manually across many countries and the CITY 'AUSTRALIA' is saved with many similes example 'AUSTRILIA', 'USSTRAILIA' or 'AUSTERILYA' etc. when lookup on these wrong one they are need to be treated as correct one i.e. 'AUSTRALIA'. The transformation generate the value accordance with the similarity between the wrong one and right one, Based on acceptance criteria like .70 and above is treated 'AUSTRALIA'.

Copyright © Capgemini 2010. All Rights Reserved.

Fuzzy Lookup Transformation

Steps to configure

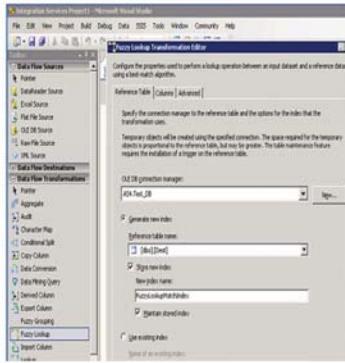
- In Business Intelligence Development Studio, open the Integration Services project that contains the package you want.
- In Solution Explorer, double-click the package to open it.
- Click the Data Flow tab, and then, from the Toolbox, drag the Fuzzy lookup transformation to the design surface.
- Connect the Fuzzy lookup transformation to the data source or a previous transformation to the Fuzzy lookup transformation.
- Double-click the Fuzzy lookup transformation to edit the pr.



Copyright © Capgemini 2010. All Rights Reserved.

Fuzzy Lookup Transformation

- On Clicking Reference tab will be open
- Create a new connection by using the Configure OLE DB Connection Manager dialog box.



The screenshot displays the 'Configure Fuzzy Lookup Transformation Editor' dialog box within the Microsoft Visual Studio interface. The dialog is titled 'Configure Fuzzy Lookup Transformation Editor' and contains the following configuration options:

- OLE DB connection manager:** Set to 'AM.Fact_08'.
- Generate new index:** Checked.
- Reference table name:** Set to '(all) [dev].[Fact]'
- Sign new index:** Checked.
- New index name:** Set to 'FuzzyLookupIndex'
- Use existing index:** Checked.

The background of the Visual Studio window shows a list of available transformations under the 'Data Flow Transformations' category, including 'Fuzzy Lookup' which is currently selected.

Fuzzy Lookup Transformation

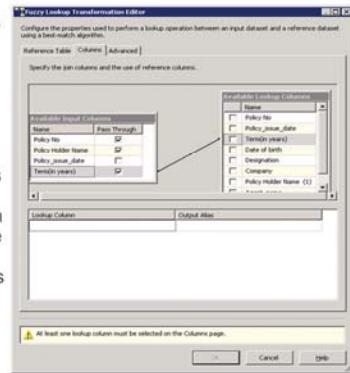
- Refer the Picture in previous slide
 - Generate new index :Specify that the transformation should create a new index to use for the lookup.
 - Reference table name :Select the existing table to use as the reference (lookup) table.
 - Store new index :Select this option if you want to save the new lookup index.
 - New index name :If you have chosen to save the new lookup index, type a descriptive name for it.
 - Maintain stored index :If you have chosen to save the new lookup index, specify whether you also want SQL Server to maintain the index.
 - Use existing index :Specify that the transformation should use an existing index for the lookup.
 - Name of an existing index :Select a previously created lookup index from the list.



Copyright © Capgemini 2010. All Rights Reserved. 19

Fuzzy Lookup Transformation

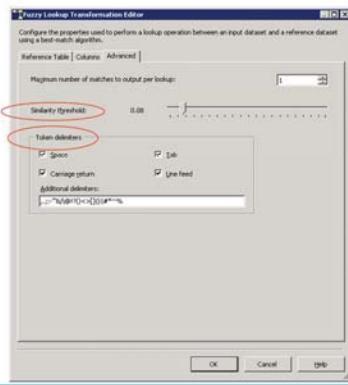
- In COLUMNS tab join the ports on basis of which lookup will be performed. The both input and lookup port should have same Integration Services Data Types.
- Pass Through : Specify whether to include the input columns in the output of the transformation.
- Lookup Column : Select lookup columns from the list of available reference table columns. Your selections are reflected in the check box selections in the Available Lookup Columns table. Selecting a column in the Available Lookup Columns table creates an output column that contains the reference table column value for each matching row returned.



Copyright © Capgemini 2010. All Rights Reserved

Fuzzy Lookup Transformation

- Output Alias :Type an alias for the output for each lookup column. The default is the name of the lookup column with a numeric index value appended; however, you can choose any unique, descriptive name.
- Maximum number of matches to output per lookup
- Specify the maximum number of matches the transformation can return for each input row. The default is 1.



Fuzzy Lookup Transformation

■ Similarity threshold

- Set the similarity threshold at the component level by using the slider. The closer the value is to 1, the closer the resemblance of the lookup value to the source value must be to qualify as a match. Increasing the threshold can improve the speed of matching since fewer candidate records need to be considered.

■ Token delimiters

- Specify the delimiters that the transformation uses to tokenize column values. Press Ok



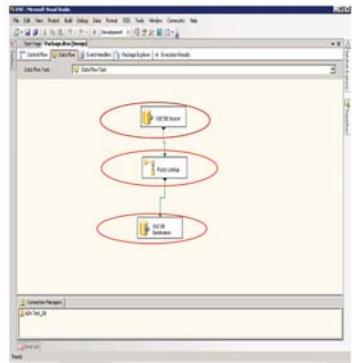
Copyright © Capgemini 2010. All Rights Reserved

Fuzzy Lookup Transformation

- The closer the value of Similarity Threshold is to 1, the closer the resemblance of the lookup value to the source value, & It must be to qualify as a match. Increasing the threshold can improve the speed of matching since fewer candidate records left for the consideration.

Fuzzy Lookup Transformation

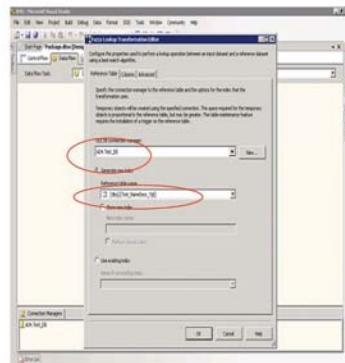
- In the Example there is Source Table "Test_NameDesc". It has got 2 columns "fname" and "desc"
- "Test_NameDesc_Tgt" is the Lookup Table having 2 columns "fname" and "desc"
- "Test_NameDesc_123" is the Target Table having 2 columns "fname" and "desc" From the Source Tables, Values from the Column "fname" needs to be Looked up with Column "fname" from Lookup Table and the "fname and desc" needs to be populated in the Target Table
- There are 3 transformation
 1. Source Transformation
 2. Fuzzy Lookup Transformation
 3. Destination Transformation



Copyright © Capgemini 2010. All Rights Reserved.

Fuzzy Lookup Transformation

- Select the Database Connection
- Select the Lookup Table



Microsoft Visual Studio

Fuzzy Lookup Transformation Editor

Configure the properties used to perform a lookup operation between an input dataset and a reference dataset.

Source: Ola Test DB

Reference: AdventureworksLT [AdventureworksLT]

Connection: Ola Test DB

Reference database: AdventureworksLT

General

Use exact match

Match on all columns

Output

Output Columns

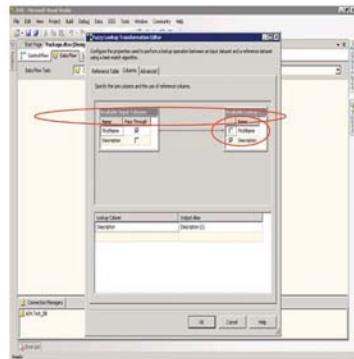
OK Cancel Help

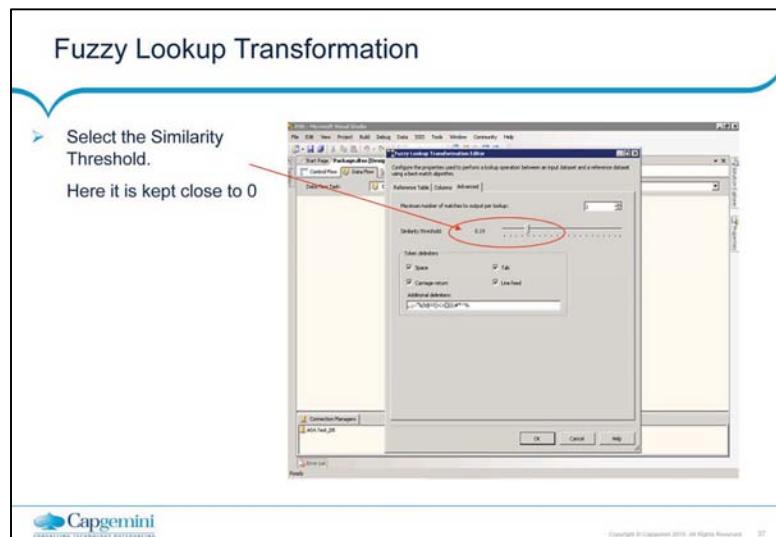
Capgemini

Copyright © Capgemini 2010. All Rights Reserved.

Fuzzy Lookup Transformation

- Map the Lookup Columns from Source Table with the Lookup Table
- Select the Description.





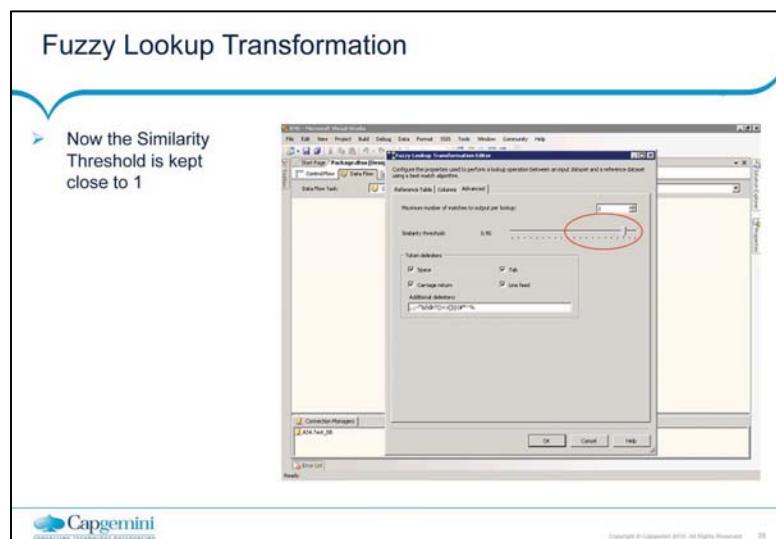
Fuzzy Lookup Transformation

All the Columns with the Similarity found is been loaded in Target Table with the "fname" and Desc"

	fname	desc
1	amit	b123
2	kaushal	NULL
3	amita	b123
4	abhinav	c123
5	aditya	d213
6	kalash	NULL

The screenshot shows the Microsoft SQL Server Management Studio interface. A query has been run against the 'master' database, specifically targeting the 'syscolumns' system table. The results grid displays several columns of data, with the last two columns, 'name' and 'desc', being highlighted by a red circle. Below the results grid, the status bar indicates 'Query executed successfully'.

Capgemini
INNOVATING THROUGH INTEGRATION



Fuzzy Lookup Transformation

Only 1 Columns with the Similarity found is been loaded in Target Table with the "fname and Desc". Rest Columns are populated with NULL.

	fname	desc
1	amit	NULL
2	kaushal	NULL
3	amita	b123
4	abhinav	NULL
5	aditya	NULL
6	kalash	NULL

Microsoft SQL Server Management Studio

File Edit View Query Tools Task Window Help

Object Explorer

Server Explorer

Toolbox

Task List

Results Grid

Messages

Object Explorer

Server Explorer

Toolbox

Task List

Results Grid

Messages

Name Message

amit Match

NULL No match

Query executed successfully.

Copyright © Capgemini 2010. All Rights Reserved.

Capgemini

Import Column Transformation

Lesson 19:

Import Column Transformation

- Import Column Transformation reads the data from files and adds the data to columns in a data flow. Using this Transformation, a package can add text and images stored in separate files to a data flow.
- For example, a data flow loading data into table that stores product information can include Import Column Transformation to import customer reviews of each product from files and add the reviews to data flow.



© Copyright © Capgemini 2010. All Rights Reserved. 42

Import Column Transformation

Example::

- Load the records for the field 'PLANNAME' from the following path mentioned in the output data set by using Import Column Transformation.

POLICYNUMBER	CARRIERCODE	POLICYSTATUS	PAYMENTMETH...	CARRI...	PLANNAME
POL1	CAR3	3	4	56	D:\dipankar\new1.txt
POL1	CAR1	4	889	22	D:\dipankar\new1.txt
POL2	CAR2	5	66	NULL	D:\dipankar\new2.txt
NULL	NULL	NULL	NULL	NULL	NULL



Copyright © Capgemini 2010. All Rights Reserved. 43

Import Column Transformation

Output::

- Records from the field 'PLANNNAME' is successfully loaded in the output dataset
- after Import Column Transformation.

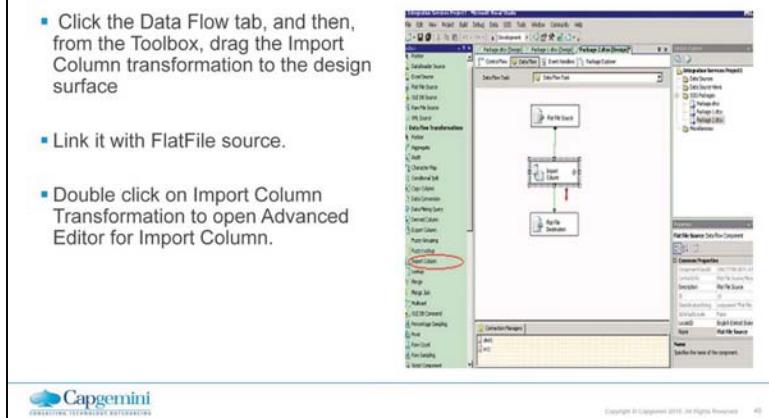
POLICYNUMBER	CARRIERCODE	POLICYSTATUS	PLANNNAME	IMPORTED CO...
POL1	CAR3	3	D:\sparkar\ne...	POL4
POL1	CAR1	4	D:\sparkar\ne...	POL5
POL2	CAR2	5	D:\sparkar\ne...	POL6



Copyright © Capgemini 2010. All Rights Reserved. 47

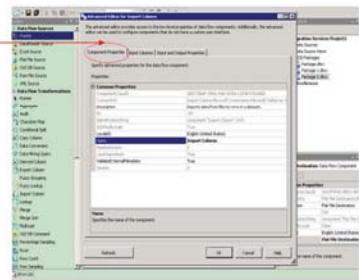
Import Column Transformation

- Click the Data Flow tab, and then, from the Toolbox, drag the Import Column transformation to the design surface
- Link it with FlatFile source.
- Double click on Import Column Transformation to open Advanced Editor for Import Column.



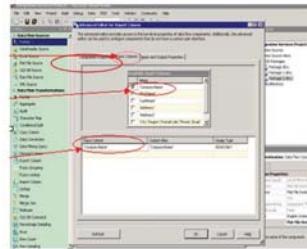
Import Column Transformation

- Component Properties:
- This tab is used for specifying advanced properties of Data Flow Component such as Name of component, Description, Locale ID.



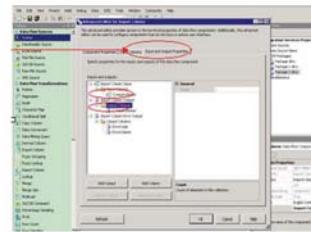
Import Column Transformation

- Input Columns:
- One of the Columns of source contains the path of a file (from which data is to be Imported).
- This column is to be selected under Input Columns tab.
- When the selected column is displayed, one may alter its usage type (Read Only / Read-Write).



Import Column Transformation

- Input & Output Properties:
- Firstly, create a new column (under Import Column Output section).
- Select the datatype of this column as Text/Image.
- The ID of this column is to be mentioned against "FileDataColumnID" property for the selected Input Column (in the previous tab) under Import Columns Input.



Lookup Transformation

Lesson 20:

Lookup Transformation

"The Lookup transformation performs lookups by joining data in input columns with columns in a reference dataset".

NOTE :

- The reference dataset can be an existing table or view or a new table, or the result of an SQL statement. The Lookup transformation uses an OLE DB connection manager to connect to the database that contains the data that is the source of the reference dataset.
- The Lookup transformation performs an equi-join between values in the transformation input and values in the reference dataset. If there is no matching entry in the reference dataset, no join occurs and no values are returned from the reference dataset. This is an error, and the transformation fails, unless it is configured to ignore errors or redirect error rows to the error output.
- The lookups performed by the Lookup transformation are case sensitive.



Copyright © Capgemini 2010. All Rights Reserved.

Lookup Transformation

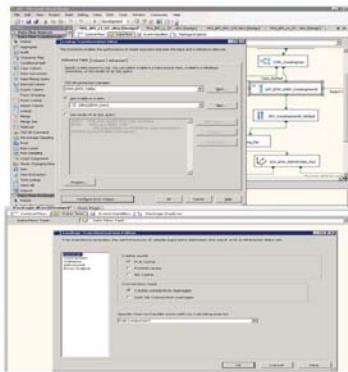
- Typically, values from the reference dataset are added to the transformation output. The values from the reference table can replace column values or can be added to new columns.
- Caching SQL Statement:
"The Lookup transformation can be configured to use a caching SQL statement to select a subset of the reference dataset, thus, limiting its size".
- EXAMPLE
 - Suppose, Our source consist of CITY_CODE and in target we want CITY_NAME instead. We have a master table consisting of code and name of cities, This transformation take city code as input, And lookup on master table, matches the code and return the corresponding Name of the city.



Copyright © Capgemini 2010. All Rights Reserved.

Lookup Transformation

- Click the Data Flow tab, and then, from the Toolbox, drag the Lookup transformation to the workspace area.
- Connect the Lookup transformation to the data flow by dragging a connector—the green or red arrow—from a source or a previous transformation to the Lookup transformation.
- Double-click the Lookup transformation component to open the Editor.
- In the General tab, select the Cache Mode and the Connection Type.



Copyright © Capgemini 2010. All Rights Reserved.

Lookup Transformation

- Select Cache Connection Manager to select the required Connection Manager Name and the file name.

Cache Connection Manager Editor

The transformation enables the performance of single-row joins between the input and a reference data set.

General | Columns

Cache connection manager: Cache Connection Manager

Description:

Use file cache

File name: C:\temp\cache.dtsvc

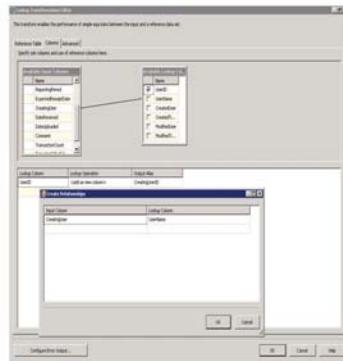
Cache refresh interval: 00:00:00

This connection level of the package does not apply to the cache. If this cache is used across multiple environments, use an account with the DTS role to connect directly to the location or folder in which you store the file.



Lookup Transformation

- Select Columns tab to create relationships between input fields for Lookup transformation and Reference table fields.
- Right-click in the space between "Available Input columns" & "Available lookup columns" to open the dialog box for creating the relationships.
- Once relationships are created, close the dialog box (Click OK) and then include the list of columns to be fetched incase a matching entry is found from the reference table.



Lookup Transformation

- Advanced tab has to be selected if one need to mention Caching SQL Statement by enabling Memory restriction and / or Caching
- Click OK to complete lookup transformation



Copyright © Capgemini 2010. All Rights Reserved.

Lookup Transformation

- Source Input

First_Name	Last_Name	City_Code
saurabh	ganguly	452001
sachin	tendulkar	400045
rahu	dravid	300016

- Lookup Table

CITY_MASTER	City_Name
452001	Kalkattha
400045	Bombay
300016	Pune

- Lookup Output

First_Name	Last_Name	City_Code	Lkp_City_Name
saurabh	ganguly	452001	Kalkattha
sachin	tendulkar	400045	Bombay
rahu	dravid	300016	Pune

Merge Transformation

Lesson 21:

Merge Transformation

"The Merge transformation combines two sorted datasets into a single dataset. The row from each datasets are inserted into the output based on values in their key columns."

By including Merge transformation into the dataflow following tasks can be performed :

- Merge data from two data sources, such as tables and files.
- Create complex datasets by nesting Merge transformations.
- Remerge rows after correcting errors in the data.
- The Union All transformation is used instead of Merge transformation in the following situation :
 - The transformation inputs are not sorted.
 - The combined output does not need to be sorted.
 - The transformation has more than two inputs.



Copyright © Capgemini 2010. All Rights Reserved. 19

Merge Transformation

- EXAMPLE: The data coming from two different databases is as shown below.

Input Data Source 1

POLICYNUMBER	CARRIERCODE	POLICYST...	PAYMENTM...	CARRIERPA...	PLANNAME	LINEOFBU...
POL2	CAR2	5	NULL	NULL	NULL	1
POL1	CAR2	567	25	NULL	NULL	1
POL1	CAR1	1	NULL	NULL	NULL	NULL
POL1	CAR1	1	NULL	NULL	NULL	NULL
POL1	CAR1	1	NULL	NULL	NULL	NULL

Input Data Source 2

POLICYNU...	CARRIERC...	POLICYST...	PAYMENT...	CARRIERP...	PLANNAME	LINEOFBU...
POL1	CAR1	8	NULL	NULL	NULL	6
POL1	CAR1	1	NULL	NULL	NULL	NULL
POL2	CAR2	5	NULL	NULL	NULL	1
POL7	CAR2	567	25	NULL	NULL	1
POL1	CAR1	1	NULL	NULL	NULL	NULL

- The data is sorted based on PolicyNumber & CarrierCode Columns using Sort transformation. This Sorted data is given as a input to merge transformation.



Copyright © Capgemini 2010. All Rights Reserved.

Merge Transformation

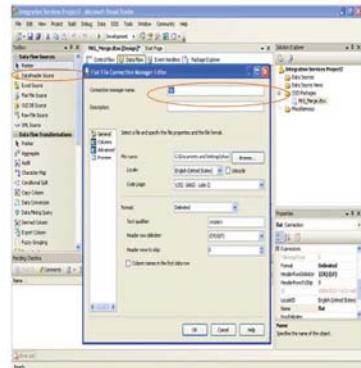
- The output of the Merge Transformation is as shown. The Merging of records is done on the basis of 'POLICYNUMBER' & 'CARRIERCODE' which are used as key columns.

POLICYNUMBER	CARRIERCODE	POLICYSTATUS	PAYOUTMETHOD	CARRIERPARTNER	LINESOFBUSINESS
POL1	CAR1	8	NULL	NULL	NULL
POL1	CAR1	1	NULL	NULL	NULL
POL1	CAR1	1	NULL	NULL	NULL
POL1	CAR1	1	NULL	NULL	NULL
POL1	CAR1	1	NULL	NULL	NULL
POL1	CAR1	1	NULL	NULL	NULL
POL1	CAR2	567	25	NULL	NULL
POL2	CAR2	5	NULL	NULL	1
POL2	CAR2	5	NULL	NULL	NULL
POL7	CAR2	567	25	NULL	1

Copyright © Capgemini 2010. All Rights Reserved.

Merge Transformation

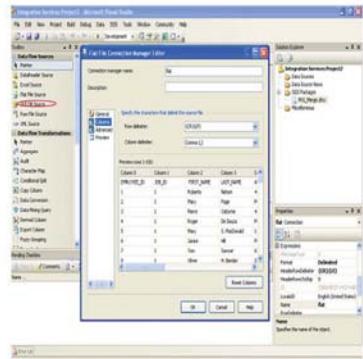
- Drag the 'Flat File Source' from Data Flow Task Navigator from Data Flow Task Window.
- Double click or Right click on 'Flat File Source' to open 'Flat File Source Editor'
- Set the 'Connection Manager' by giving appropriate information, e.g. Connection Manager Name, File Name.



Copyright © Capgemini 2010. All Rights Reserved.

Merge Transformation

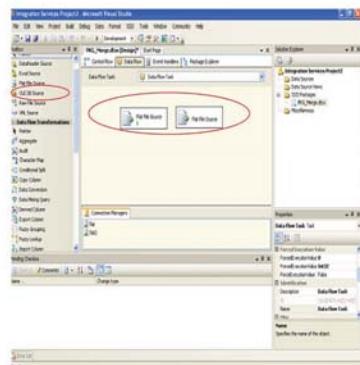
- Select Columns and check required 'Column'.
- Click 'OK'



Copyright © Capgemini 2009. All Rights Reserved.

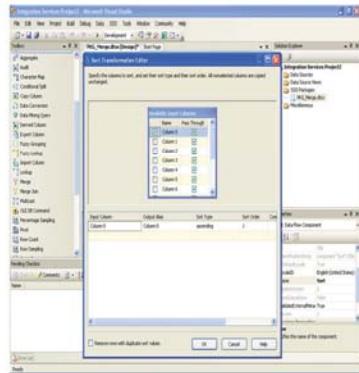
Merge Transformation

- Drag another 'Flat File Source' from Data Flow Task Navigator into Data Flow Task Window.
- Create a Flat File Connection as explained in the previous slide.



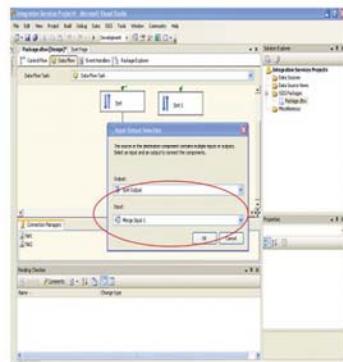
Merge Transformation

- Drag a 'Sort Transformation' from the Data Flow Transformation into the Data Flow Task Window.
- Connect the output of one Flat File Source to the Sort Transformation.
- Double click or Right click the Sort Transformation to enter in Sort Transformation Editor.
- Put a check on the right check box of the field according to which sorting is to be done.
- Click OK.



Merge Transformation

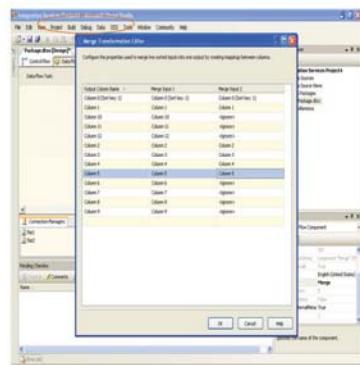
- Drag a 'Merge Transformation' from the Data Flow Transformation into the Data Flow Task Window.
- Connect the Output of one Sort Transformation to the Merge Transformation, an Input Output Selection Window will be opened. Select an Input and an Output to connect the components.
- Similarly connect the output of the other Sort Transformation to the Merge Transformation.



Copyright © Capgemini 2010. All Rights Reserved.

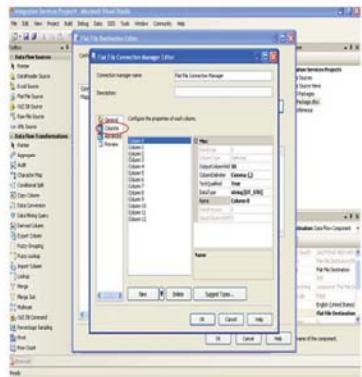
Merge Transformation

- Double Click or Right Click on the Merge Transformation to enter into the 'Merge Transformation Editor'.
- In Merge Transformation Editor one can configure the properties to merge two sorted inputs into one output by creating mappings between the columns.
- Click OK.



Merge Transformation

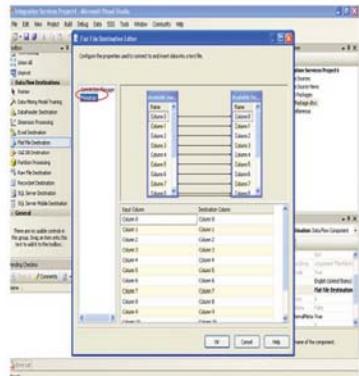
- Drag the 'Flat File Destination' from the Data Flow Destination into Data Flow task Window.
- Double Click or Right Click the Flat File Destination to enter into the 'Flat File Destination Editor'.
- Click on 'Connection Manager' and create connection as described before. Click on Columns to specify the number of columns.
- Click OK.



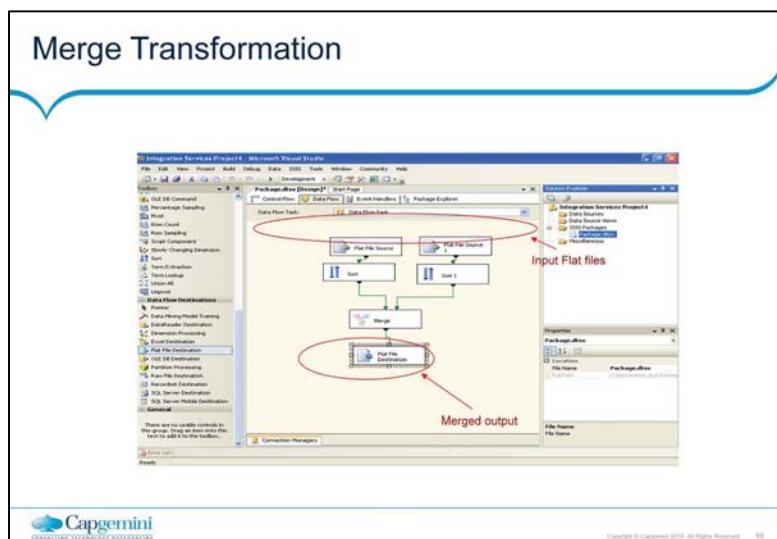
Copyright © Capgemini 2007. All Rights Reserved.

Merge Transformation

- Click on Mapping and map the corresponding columns.
- Click OK.



Copyright © Capgemini 2007. All Rights Reserved.



Merge Join Transformation

Lesson 22:

Merge Join Transformation

- EXAMPLE: Following are the two different data sources named as Source1 & Source2 respectively. Here we are using Inner Join with the help of Merge Join Transformation.

POLICYNUMBER	CARRIERCODE	POLICYSTATUS	PAYMENTMETHOD
POL1	CAR3	3	4
POL1	CAR1	4	889
POL2	CAR2	5	66

POLICYNUMBER	CARRIERCODE	POLICYSTATUS	PAYMENTMETHOD
POL1	CAR1	8	545
POL3	CAR4	44	555
POL2	CAR2	5	NULL

- The Table shown below represents output when Inner Join is adopted for POLICYNUMBER & CARRIERCODE fields.

POLICYNUMBER	CARRIERCODE	POLICYSTATUS	POLICYSTATUS (1)	PAYMENTMETHOD...	PAYMENTMETHOD (1)
POL1	CAR1	4	8	889	545
POL2	CAR2	5	5	66	

Merge Join Transformation

- This transformation has two inputs and one output. It does not support an error output.

You can configure the Merge Join transformation in the following ways:

- Specify the join is a FULL, LEFT, or INNER join.
- Specify the columns the join uses.
- Specify whether the transformation handles null values as equal to other nulls



Copyright © Capgemini 2010. All Rights Reserved. 17

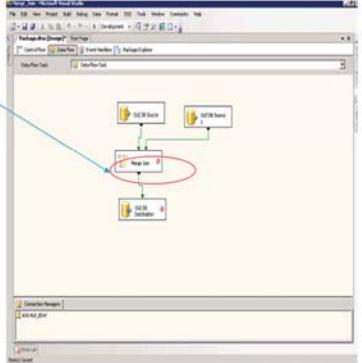
Merge Join Transformation

- In Business Intelligence Development Studio, open the Integration Services project that contains the package you want.
- In Solution Explorer, double-click the package to open it.
- Click the Data Flow tab, and then, from the Toolbox, drag the Merge Join transformation to the design surface.
- Connect the Merge Join transformation to the data flow by dragging the connector—the green or red arrow—from a data source or a previous transformation to the Merge Join transformation.
- Double-click the Merge Join transformation.
- In the Merge Join Transformation Editor dialog box, select the type of join to use in the Join type list.

Copyright © Capgemini 2010. All Rights Reserved. 79

Merge Join Transformation

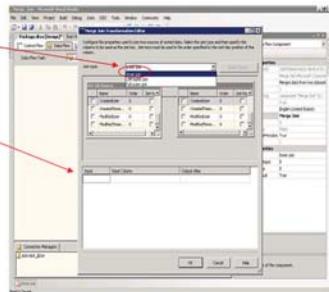
- Configure the Is Sorted Property of Source Transformation to TRUE.
- Double Click on the Merge Join Transformation to Open.



Capgemini

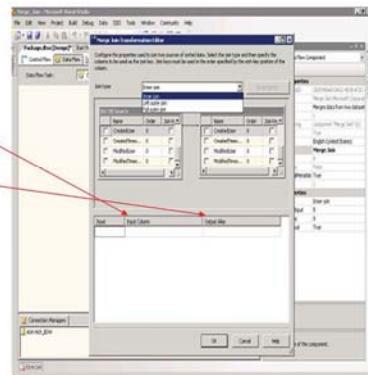
Merge Join Transformation

- Join type
- Specify whether you want to use an inner join, left outer join, or full join.
- Swap Inputs
- Switch the order between inputs by using the Swap Inputs button. This selection may be useful with the Left outer join option.
- Input
- For each column that you want in the merged output, first select from the list of available inputs.
- Inputs are displayed above in two separate tables. Select columns to include in the output. Drag columns to create a join between the tables.



Merge Join Transformation

- Input Column
- Select a column to include in the merged output from the list of available columns on the selected input.
- 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 2010. All Rights Reserved.

Merge Join Transformation

- Drag columns in the left input to columns in the right input to specify the join columns. If the columns have the same name, you can select the Join Key check box and the Merge Join transformation automatically creates the join.
- In the left and right inputs, select the check boxes of additional columns to include in the output. Join columns are included by default.
- Optionally, update the names of output columns in the Output Alias column.
- Click OK.
- To save the updated package, click Save Selected Items on the File menu.

Copyright © Capgemini 2010. All Rights Reserved

Merge Join Transformation

- The Merge Join transformation provides an output that is generated by joining two sorted datasets using a FULL, LEFT, or INNER join.
- The Merge Join transformation requires that both inputs be sorted and that the joined columns have matching metadata. For example, you cannot join a column that has a numeric data type with a column that has a character data type. If the data has a string data type, the length of the column in the second input must be less than or equal to the length of the column in the first input with which it is merged.
- The data can be sorted by setting sort options on the outputs of the source or other upstream data flow components, or by inserting a Sort transformation into the data flow before the Merge Join transformation. If the sort options indicate that the data is sorted, but the data is not actually sorted, the merge join operation may have unpredictable results.

Copyright © Capgemini 2010. All Rights Reserved. 19

Multicast Transformation

Lesson 23:

Multicast Transformation

"The Multicast transformation creates copies of dataset. This transformation distributes its input to one or more output"

- Using the Multicast transformation, a package can create logical copies of data.
- This capability is useful when the package needs to apply multiple sets of transformations to the same data.
- This transformation has one input and multiple outputs. It does not support an error output.

EXAMPLE

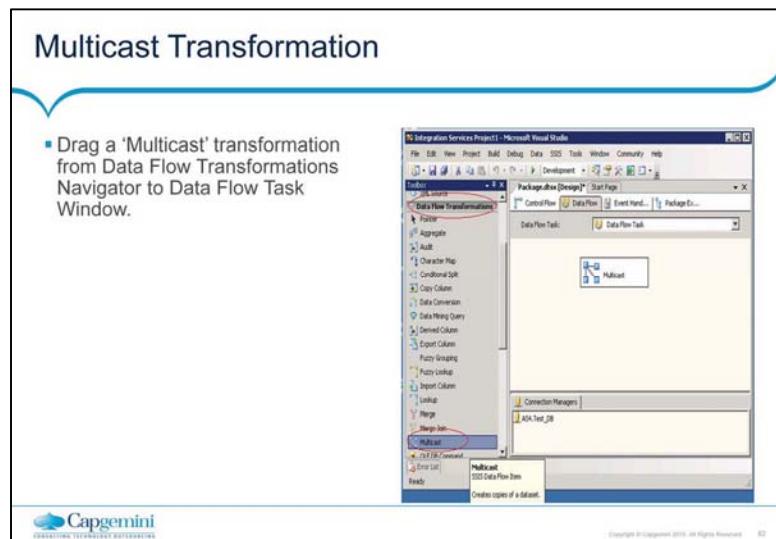
- One copy of the data is aggregated and only the summary information is loaded into its destination, While another copy of the data is extended with lookup values and derived columns before it is loaded into its destination.



Copyright © Capgemini 2010. All Rights Reserved.

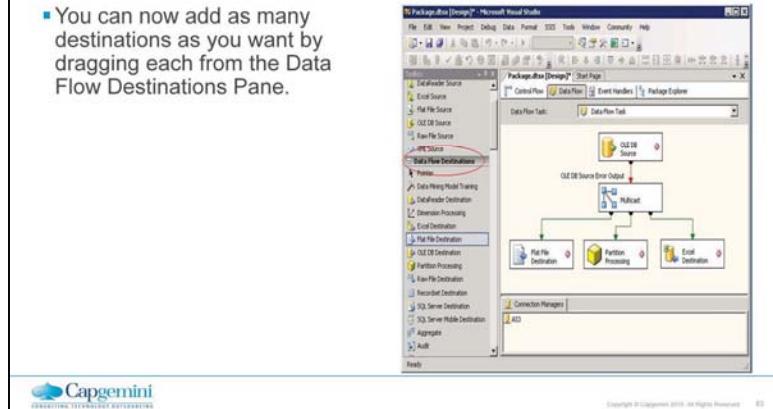
Multicast Transformation

- To configure Multicast Transformation, you can set properties through SSIS Designer or programmatically.
- You configure the Multicast transformation by adding outputs.
- The Multicast transformation distributes its input to one or more outputs.
- This transformation is similar to the Conditional Split transformation. Both transformations direct an input to multiple outputs.
- The difference between the two is that the Multicast transformation directs every row to every output, and the Conditional Split directs a row to a single output. For more information, see Conditional Split Transformation.



Multicast Transformation

- You can now add as many destinations as you want by dragging each from the Data Flow Destinations Pane.



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.

Copyright © Capgemini 2016. All Rights Reserved.

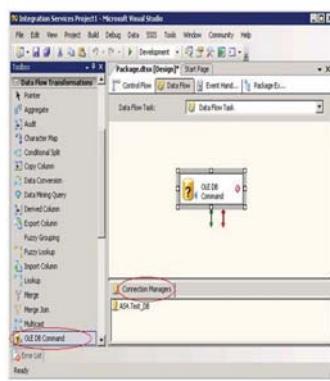
OLEDB Command Transformation

- 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 2016. All Rights Reserved.

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

▪ 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

The advanced editor provides access to the fine-grained properties of data flow components. Additionally, the advanced editor can be used to configure components that do not have a custom user interface.

Connection Managers Component Properties Column Mappings Input and Output Properties

Select the connection managers to be used for the data flow component.

Connection managers:

Name	Connection Manager	Description
OLEDBConn	AQA_Test_DB	The OLE DB native connect...
AQA_Test_DB		

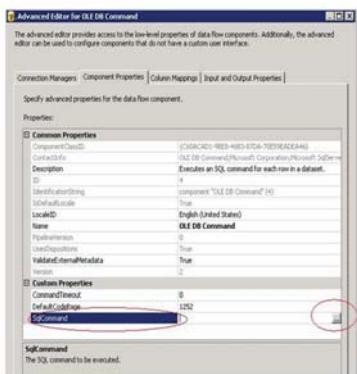
OLEDB Command Transformation

Copyright © Capgemini 2006. All Rights Reserved.

Capgemini

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.

The screenshot shows the 'Advanced Editor for OLE DB Command' window. The left pane displays the 'Inputs and Outputs' tab, which contains a tree view with 'OLE DB Command Input' expanded, showing 'External Columns' and 'Input Columns'. The right pane shows the 'Common Properties' section with the following details:

Description:	EncryptConnection: Command Encryption
ConnectionString:	OLEDBCommand
CommandText:	True
MaxBatchSize:	1000
IdentifierString:	Input "OLE DB Command Input"
IsCommand:	False
Name:	OLE DB Command Input
SourceName:	1003
TransactFromInput:	No

At the bottom of the editor, there are buttons for 'Add Output', 'Add Column', 'Remove Output', and 'Remove Column'.

Capgemini
www.capgemini.com

Percentage Sampling Transformation



Copyright © Capgemini 2010. All Rights Reserved.

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 2016. All Rights Reserved

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"



Copyright © Capgemini 2006. All Rights Reserved

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	Cty
"10019"	"Bobbi"	"Arndt"	"Market Place"	"1000 S Nicole..."	"Sametown"
"10025"	"Butch"	"Bobbi"	"Town & Count..."	"108 Hilltop Ct"	"Smithville"
"10029"	"Cheri"	"Bulsky"	"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 2008. All Rights Reserved.

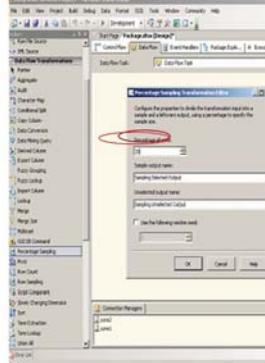
Percentage Sampling Transformation

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

Capgemini

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

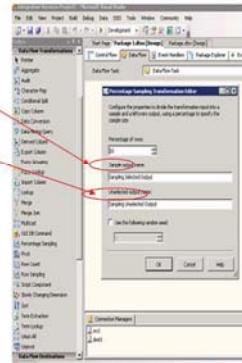


The screenshot shows the 'Configure Percentage Sampling Transformation' dialog box. The 'Percentage' dropdown is set to 20. The 'Sampling method' dropdown is set to 'Sampling method based on row count'. The 'Sampling method based on row count' checkbox is checked. The 'Use the filtering rowset seed' checkbox is unchecked.

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.

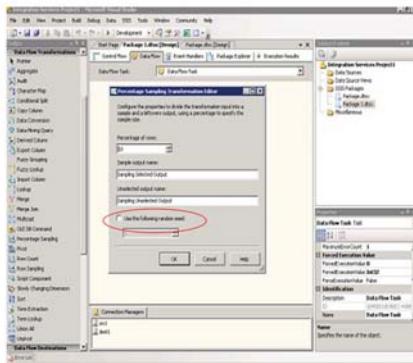


Copyright © Capgemini 2008. All Rights Reserved.

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.



Copyright © Capgemini 2008. All Rights Reserved

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 2016. All Rights Reserved.

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 2016. All Rights Reserved. 10

Pivot Transformation

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

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

Copyright © Capgemini 2010. All Rights Reserved.

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

Copyright © Capgemini 2010. All Rights Reserved. 20

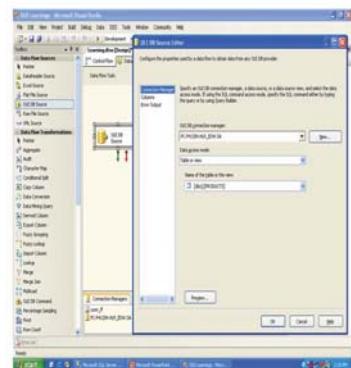
Pivot Transformation

- 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 2016. All Rights Reserved. 21

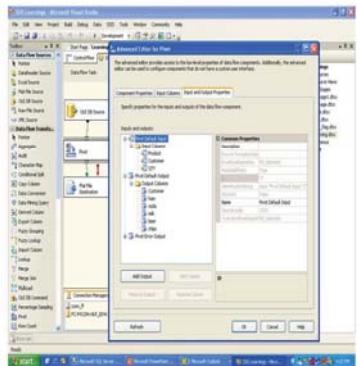
Pivot Transformation

- 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 2016. All Rights Reserved. 30

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 2010. All Rights Reserved. 30

Row Count Transformation

Lesson 25:

Row Count Transformation

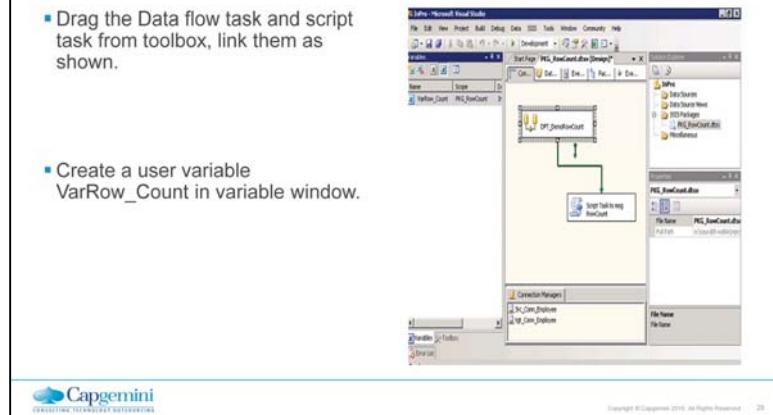
- “ The Row Count transformation counts rows as they pass through a data flow and stores the final count in a variable.”
- A SQL Server 2008 Integration Services (SSIS) package can use row counts to update the variables used in scripts, expressions, and property expressions. For example, the variable 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 2008. All Rights Reserved. 21

Row Count Transformation

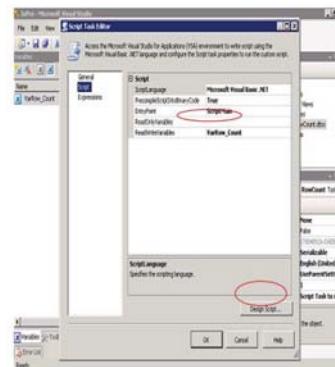
- Drag the Data flow task and script task from toolbox, link them as shown.

- Create a user variable VarRow_Count in variable window.



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.



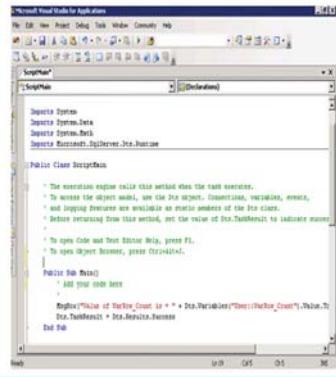
Row Count Transformation

- 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)
```

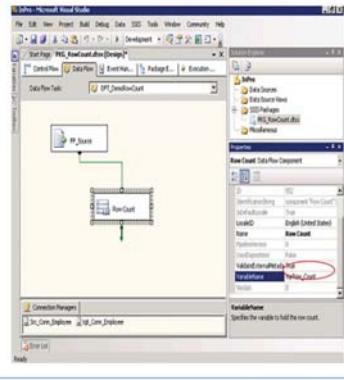
- Save the script. Close the editor and click ok on script task editor.

- 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.



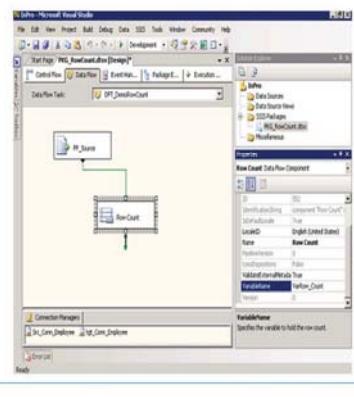
Row Count Transformation

▪ 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.



Copyright © Capgemini 2008. All Rights Reserved.

Row Count Transformation-2012 Improvements

- New User interface is introduce 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 2010. All Rights Reserved. 30

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 2010. All Rights Reserved. 50

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	MONTHSEMPIOED	YEARSEMPIOED	MODIFIEDDATE
1	POL1	HANN	1	Cicketer	500000	1	1	24	2
2	POL2	HANN	2	Cicketer	200000	1	1	10	1
3	POL3	HANN	3	Businessmen	1000000	1	1	60	5
4	POL4	HANN	4	Businessmen	100000	1	11	5	1
5	POL5	HANN	5	Businessmen	500000	1	1	20	2
6	POL6	HANN	6	Service	200000	1	1	12	1
7	POL7	HANN	7	Service	100000	1	1	24	2
8	POL8	HANN	8	CA	600000	1	1	12	1
9	POL9	HANN	9	Bulder	1E+07	1	1	48	4
10	POL10	HANN	10	Pilot	6000000	11	1	24	2



Copyright © Capgemini 2006. All Rights Reserved.

Row Sampling Transformation

Output:

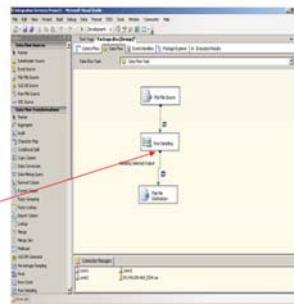
- The following is the output data set comprising of 5 rows after Row Sampling Transformation is done. This is the output in case of Sampling Selected Output. The records not displayed here are part of Sampling Unselected Output.

POLICYNUMBER	CARRIERCODE	PARTYID	OCCUPATION	ANNUALEARNINGS	ICD9	PHRASEID	MONTHSEMPLOYED	YEARSEMPLOYED	MODIFIEDDATE
1	POL1	HANN	1	Cocketer	500000	1	1	24	2
2	POL7	HANN	7	Service	100000	1	1	24	2
3	POL10	HANN	10	Pilot	600000	11	1	24	2
4	POL6	HANN	6	Service	200000	1	1	12	1
5	POL5	HANN	5	Businessmen	500000	1	1	20	2

Copyright © Capgemini 2008. All Rights Reserved. 30

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.



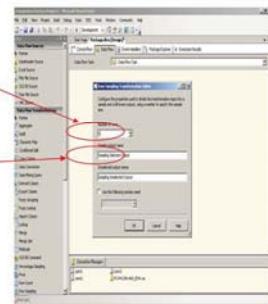
Row Sampling Transformation

- 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

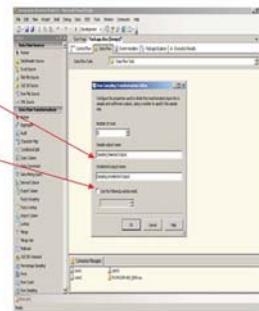
Provide a unique name for the output that will include the sampled rows. The name provided will be displayed within the SSIS Designer



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.

Copyright © Capgemini 2008. All Rights Reserved.

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 2010. All Rights Reserved. 47

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, True 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 2010. All Rights Reserved. 40

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.

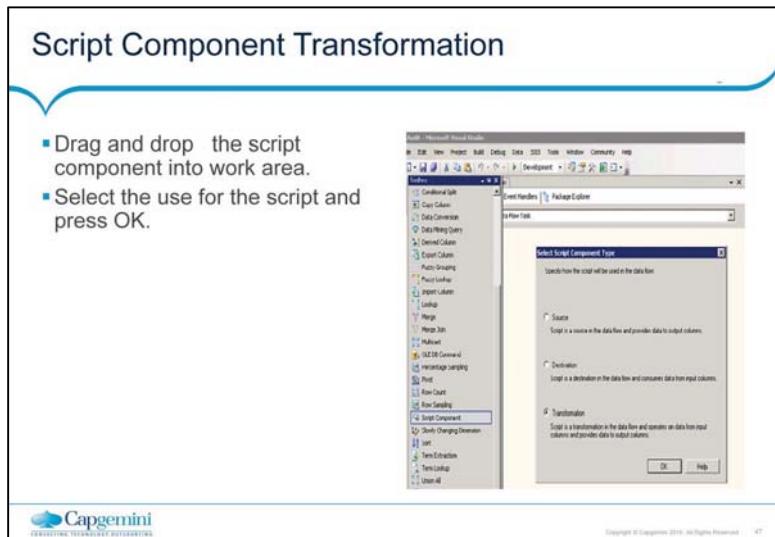
Copyright © Capgemini 2016. All Rights Reserved. 40

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:

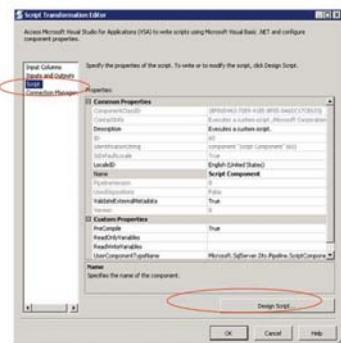
- When used as a source, the Script component has no input and supports multiple outputs.
- When used as a transformation, the Script component supports one input and multiple outputs.
- When used as a destination, the Script component supports one input and has no outputs.

Copyright © Capgemini 2016. All Rights Reserved. 40



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 2016. All Rights Reserved.

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 2016. All Rights Reserved.

Slowly Changing Dimension Transformation

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

Copyright © Capgemini 2016. All Rights Reserved

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 2010. All Rights Reserved.

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

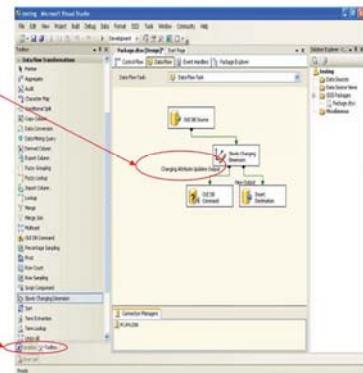
Records After Insertion

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 2010. All Rights Reserved. 50

Slowly Changing Dimension Transformation

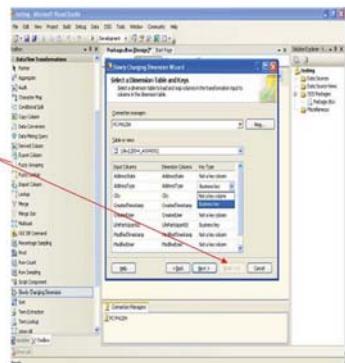
- Choose the dimension table or view you want to update. After you select the connection manager, you can select the table or view from the data source.
- Choose the connection manager to access the data source that contains the dimension table that you want to update.



Copyright © Capgemini 2008. All Rights Reserved.

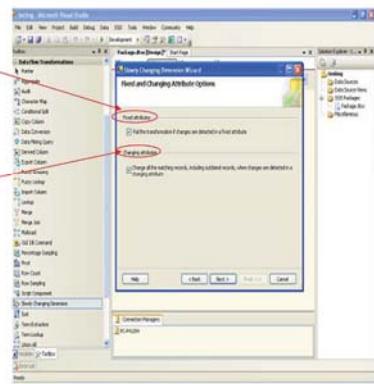
Slowly Changing Dimension Transformation

- 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.

Copyright © Capgemini 2008. All Rights Reserved 50

Slowly Changing Dimension Transformation

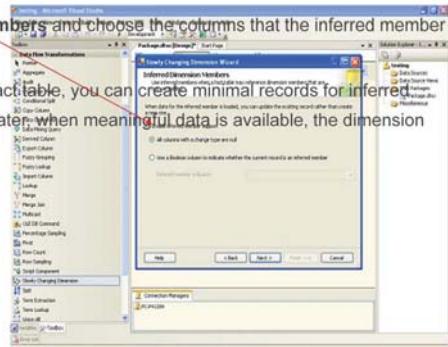
- Set fixed and changing attribute options. For the Fixed attribute change type, you can specify whether the Slowly Changing Dimension transformation fails when changes are detected in these columns. For the Changing attribute change type, you can specify whether all matching records, including outdated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

Copyright © Capgemini 2008. All Rights Reserved.

Slowly Changing Dimension Transformation

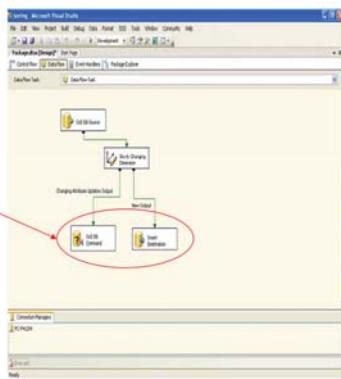
- Specify support for **inferred members** and choose the columns that the inferred member record contains.

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.

Copyright © Capgemini 2010. All Rights Reserved.

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 2010. All Rights Reserved. 01

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 2016. All Rights Reserved. 63

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 2010. 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 2010. All Rights Reserved.

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	Date	Op	Department	Salary
"10019"	"Bobbi"	"Avery"	"Market Place"	"1000 S Nicollet....."	"Saint Paul"	"1991-06-12 21"	"I"	"Sales"	"100,000"
"10020"	"Bruce"	"Bryant"	"Market Place"	"1000 S Nicollet....."	"Saint Paul"	"1991-06-12 21"	"I"	"Sales"	"100,000"
"10024"	"Bruce"	"Byron"	"La Salle Clinic"	"1009 E Wisconsin....."	"Milwaukee"	"1991-03-18 03"	"I"	"Nursing"	"223,000"
"10025"	"Butch"	"Bubba"	"Town & Country"	"1009 Hilltop Ct....."	"Smithville"	"1991-06-30 00"	"A"	"Sales"	"224,000"
"10026"	"Cecil"	"Bouchers"	"Satum of Asia"	"1110 W Northland....."	"Milwaukee"	"1991-03-18 03"	"I"	"Sales"	"225,000"
"10027"	"Cecil"	"Brauer"	"AAU Member ..."	"1110 W Northland....."	"Milwaukee"	"1991-03-18 03"	"I"	"Sales"	"226,000"
"10028"	"Carol"	"Brauer"	"Office Support"	"1112 E Glendale....."	"Smithville"	"1991-11-07 15"	"A"	"Sales"	"227,000"
"10029"	"Carol"	"Buulyah"	"AAU Member ..."	"1112 E Glendale....."	"Smithville"	"1990-01-01 15"	"A"	"Sales"	"228,000"
"10030"	"Carol"	"Buulyah"	"AAU Member ..."	"1112 E Glendale....."	"Smithville"	"1990-01-01 15"	"A"	"Sales"	"229,000"
"10031"	"Chuck"	"Carpendale"	"Kite Electric"	"1115 S Dreher St....."	"Smithville"	"1994-03-01 00"	"A"	"Sales"	"230,000"
"10032"	"Chuck"	"Cars"	"Alpha 1"	"1115 Valley F....."	"Smithville"	"1994-01-01 00"	"A"	"Sales"	"231,000"
"10033"	"David"	"Casper"	"Bigfoot C....."	"1200 N Parkin....."	"Smithville"	"1990-01-01 15"	"A"	"Sales"	"232,000"
"10035"	"David"	"Catterson"	"Fox Computer"	"1200 N Parkin....."	"Smithville"	"1990-01-01 15"	"A"	"Sales"	"234,000"
"10036"	"Conrad"	"Cay"	"Valley Lanes C....."	"1211 N Highland....."	"Smithville"	"1990-01-01 15"	"A"	"Sales"	"235,000"
"10037"	"Coy	"Cay"	"Hercules Headline"	"1221 N Highland....."	"Smithville"	"1990-01-01 15"	"A"	"Sales"	"236,000"
"10038"	"Dale"	"Cooperman"	"Kodak Toyota"	"1221 N Lince St"	"Smithville"	"1990-01-01 15"	"A"	"Sales"	"237,000"
"10039"	"Dan"	"Dog"	"AAU"	"1222 N Super....."	"Smithville"	"1990-01-01 15"	"A"	"Sales"	"238,000"
"10040"	"Darin"	"Dorothy"	"Dodd"	"1300 E 3rd Ave....."	"Dover Blvd."	"1990-01-01 15"	"A"	"Sales"	"239,000"
"10041"	"Darin"	"Dorothy"	"Dobberman"	"1300 E 3rd Ave....."	"Dover Blvd."	"1990-01-01 15"	"A"	"Sales"	"240,000"
"10042"	"Darin"	"Dora"	"Furious Health"	"1300 E Calumet....."	"Smithville"	"1990-03-20 09"	"A"	"Sales"	"241,000"
"10043"	"David"	"Docherty"	"AAU"	"13002 S Ridger"	"Smithville"	"1990-03-26 02"	"A"	"Sales"	"242,000"


Copyright © Capgemini 2010. All Rights Reserved

Sort Transformation

Output::

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

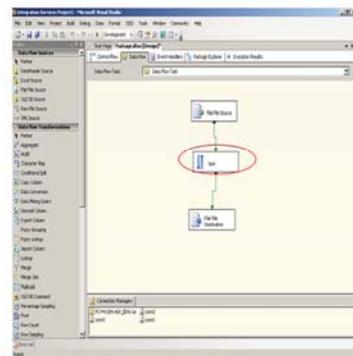
"Account No"	"First Name"	"Last Name"	"Company"	"Addre..."	"City"	"State"	"Zip"	"Department"	"Salary"
					"Jonestown"		"125.00"	"223.00"	
					"Smithville"		"130.00"	"224.00"	
					"Smithville"		"135.00"	"225.00"	
					"Jonestown"		"140.00"	"226.00"	
					"Smithville"		"145.00"	"227.00"	
					"Jonestown"		"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"	
					"Montgomery"		"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"	



Copyright © Capgemini 2016. All Rights Reserved.

Sort Transformation

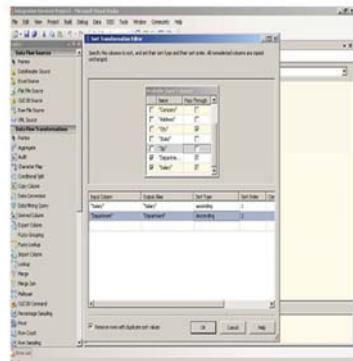
- Drag flat file Source, Sort Transformation and flat file Destination as shown
- Configure the FF_Source with Src_Conn_Employee connection.
- Link output of FF_Source to sort transformation.
- Double click to open transformation editor as shown



Copyright © Capgemini 2010. All Rights Reserved.

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 2008. All Rights Reserved. 60

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 phases.
- 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 2016. All Rights Reserved.

Term Extraction Transformation

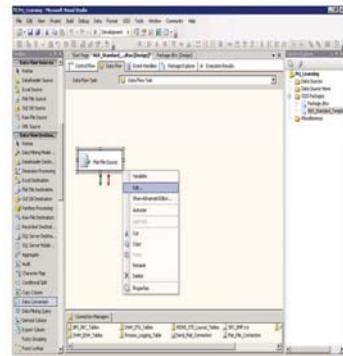
EXAMPLE

- 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 2014. All Rights Reserved. 71

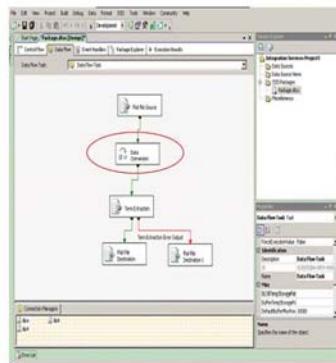
Term Extraction Transformation

- 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.)

Copyright © Capgemini 2006. All Rights Reserved. 73

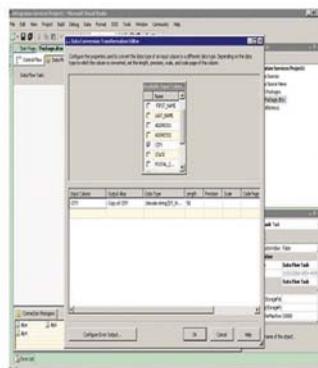
Term Extraction Transformation

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

Copyright © Capgemini 2010. All Rights Reserved. 73

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.

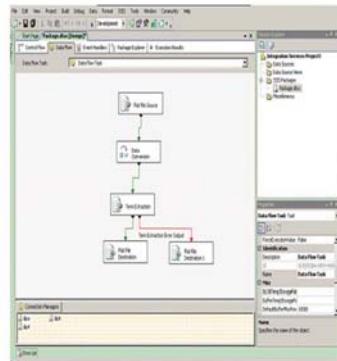


Copyright © Capgemini 2008. All Rights Reserved.

Page 03-74

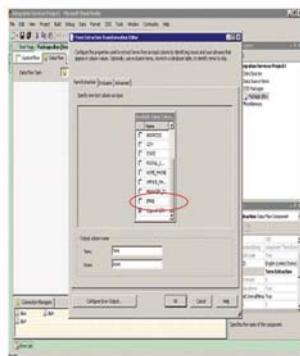
Term Extraction Transformation

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

Copyright © Capgemini 2008. All Rights Reserved. 70

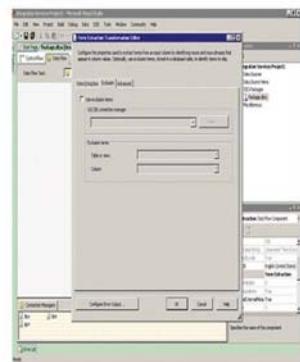
Term Extraction Transformation

- 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.

Copyright © Capgemini 2006. All Rights Reserved

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.



Copyright © Capgemini 2008. All Rights Reserved.

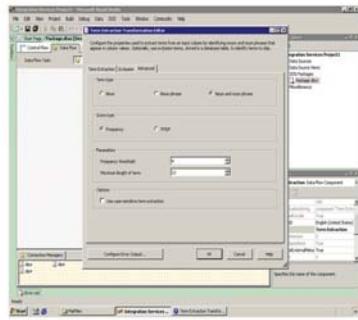
Term Extraction Transformation

- 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 = $(\text{frequency of } T) * \log(\frac{\#\text{rows in Input}}{\#\text{rows having } T})$

Copyright © Capgemini 2008. All Rights Reserved. 70

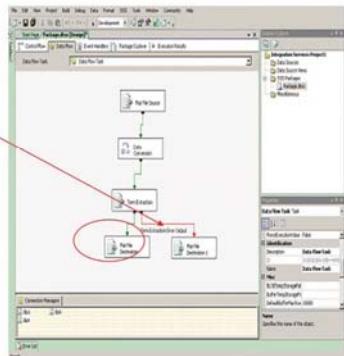
Term Extraction Transformation

- 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.

Copyright © Capgemini 2008. All Rights Reserved.

Term Extraction Transformation

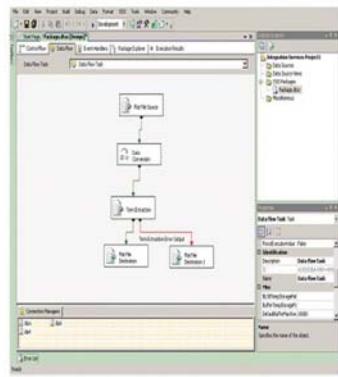
- Drag & drop the 'Flat file Destination' and connect it with the 'Term Extraction transformation'& make necessary connection in the connection manager.
- You can connect another 'Flat file Destination' for 'Term extraction Error output' which is optional.



Copyright © Capgemini 2006. All Rights Reserved.

Term Extraction Transformation

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

Copyright © Capgemini 2010. All Rights Reserved.

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 2010. All Rights Reserved.

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 2014. All Rights Reserved.

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 2016. All Rights Reserved.

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 2016. All Rights Reserved. 80

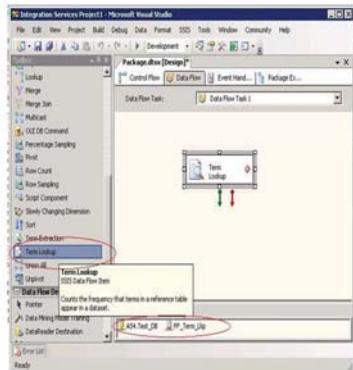
Term Lookup Transformation

- 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 2016. All Rights Reserved.

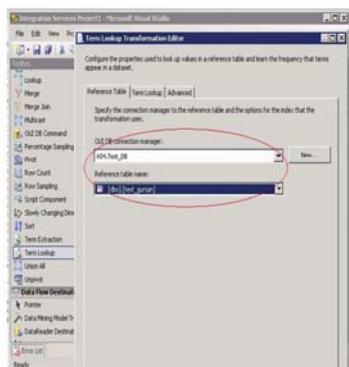
Term Lookup Transformation

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



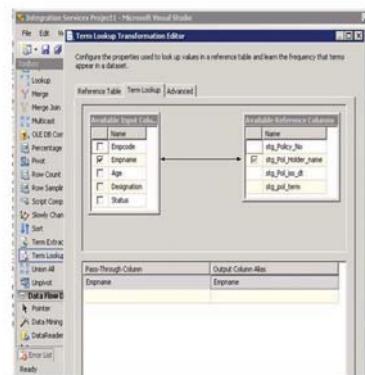
Term Lookup Transformation

- 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

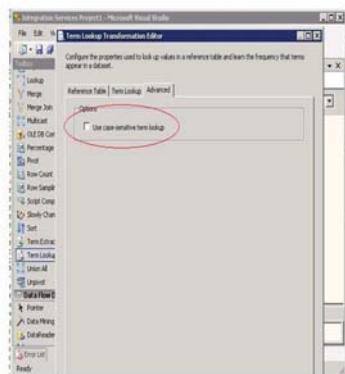
- In 'Term Lookup' tab, match available input columns with Reference columns.



Copyright © Capgemini 2006. All Rights Reserved.

Term Lookup Transformation

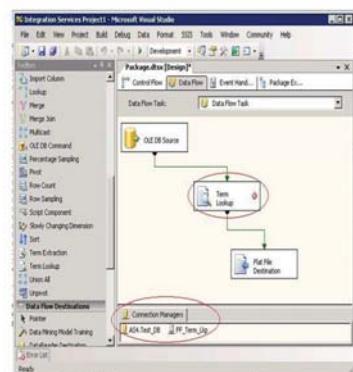
- 'Advance Setting' tab allows you to enable case sensitive lookup.



Copyright © Capgemini 2008. All Rights Reserved.

Term Lookup Transformation

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

Copyright © Capgemini 2008. All Rights Reserved.

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

50

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 2016. All Rights Reserved. 50

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"	"Braver"	"Benes Corp."	"110 W North Wat...
"10028"	"Carol"	"Braun"	"AAL Member ..."	"1115 E Glendale A...
"10029"	"Chen"	"Buksay"	"Office Support"	"1122 Milwaukee S...
"10030"	"Chuck"	"Busi"	"EAA"	"1134 S Franklin St...
"10031"	"Chuck"	"Carpenter"	"Yurz Electric"	"115 S Drew St", "2...

"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"...



Copyright © Capgemini 2016. All Rights Reserved

Union All Transformation

Output::

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

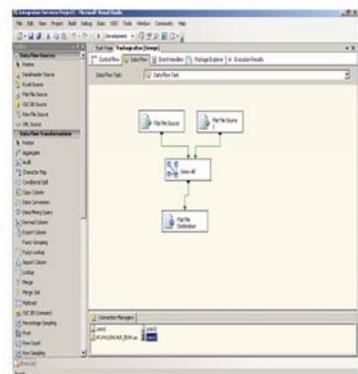
Account No	First Name	Last Name	Company	Address
"10027"	"Carol"	"Brauer"	"Bemiss Corp."	"110 W North ...
"10028"	"Carol"	"Braun"	"AAL Member ...	"1115 E Glend...
"10029"	"Cheri"	"Buksyk"	"Office Support"	"1122 Milwauk...
"10030"	"Chuck"	"Buss"	"EAA"	"1134 S Franklin...
"10031"	"Chuck"	"Carpenter"	"Kurz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



Copyright © Capgemini 2010. All Rights Reserved

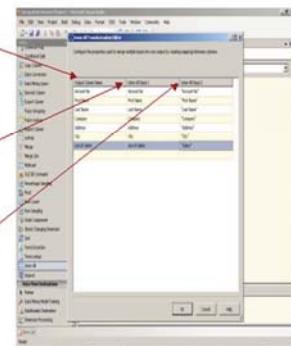
Union All Transformation

- 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.

Copyright © Capgemini 2008. All Rights Reserved.

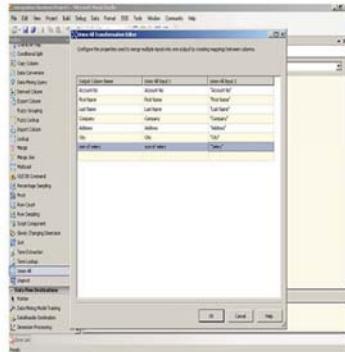
Union All Transformation

- 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

Copyright © Capgemini 2016. All Rights Reserved.

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.

Copyright © Capgemini 2008. All Rights Reserved.

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

Copyright © Capgemini 2016. All Rights Reserved. 102

Unpivot Transformation

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

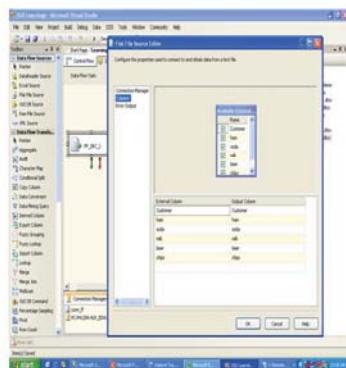
Pivot Key Col Name	False Cust	True Product	False Qty
Kate	Ham	2	
Kate	Soda	6	
Kate	Milk	1	
Kate	Beer	12	
Fred	Milk	3	
Fred	Beer	24	
Fred	Chips	2	

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

Copyright © Capgemini 2010. All Rights Reserved. 103

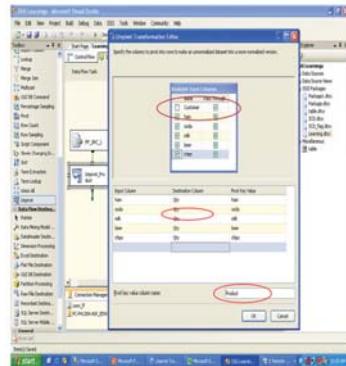
Unpivot Transformation

- 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'.

Copyright © Capgemini 2008. All Rights Reserved. 102

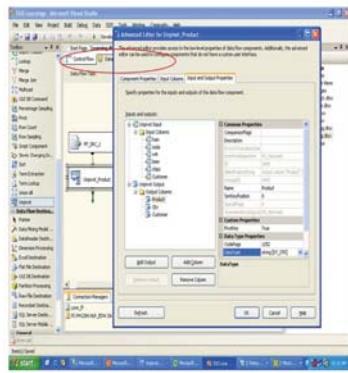
Unpivot Transformation

- In the Unpivot Transformation Editor, Select the input columns, in this example it will be ham, soda, milk,beer,chip.
- 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.

Copyright © Capgemini 2008. All Rights Reserved.

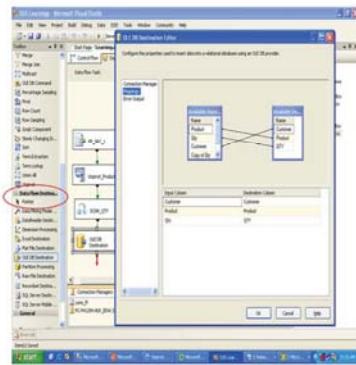
Unpivot Transformation

- 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].

Copyright © Capgemini 2010. All Rights Reserved

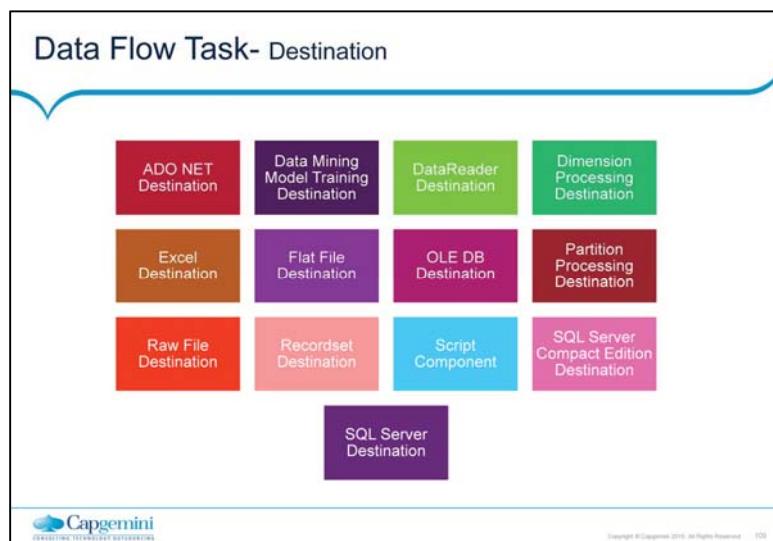
Unpivot Transformation

- 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 2010. All Rights Reserved. P10

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 2010. All Rights Reserved. 101

***SQL Server Integration
Services (SSIS)
Training Kit (PART 3)***

Lesson 10: Operating system-level
tasks

File System Task

- "The File System task performs operations on files and directories in the file system"
- For example, by using the File System task, a package can create, move, or delete directories and files. You can also use the File System task to set attributes on files and directories. For example, the File System task can make files hidden or read-only.
- All File System task operations use a source, which can be a file or a directory. The source can be specified by using a File connection manager that points to the directory or file or by providing the name of a variable that contains the source path.
- The operations that copy and move file and directories and rename files use a destination and a source. The destination is specified by using a File connection manager or a variable. File system task operations can be configured to permit overwriting of destination files and directories.



Copyright © Capgemini 2010. All Rights Reserved.

File System Task

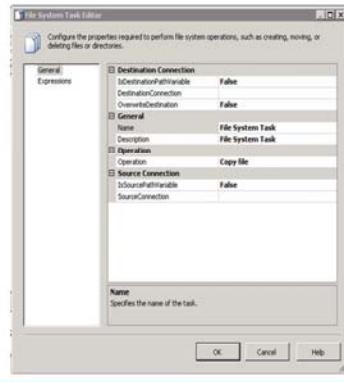
- The File System task includes a predefined set of operations. The following table describes these operations.

Operation	Description
Copy directory	Copies a folder from one location to another.
Copy file	Copies a file from one location to another.
Create directory	Creates a folder in a specified location.
Delete directory	Deletes a folder in a specified location.
Delete directory content	Deletes all files and folders in a folder.
Delete file	Deletes a file in a specified location.
Move directory	Moves a folder from one location to another.
Move file	Moves a file from one location to another.
Rename file	Renames a file in a specified location.
Set attributes	Sets attributes on files and folders. Attributes include Archive, Hidden, Normal, Read only, and System. Normal is the lack of attributes, and it cannot be combined with other attributes. All other attributes can be used in combination.

Copyright © Capgemini 2010. All Rights Reserved.

File System Task

- Is Destination Path Variable
 - Indicate whether the destination path is stored in a variable. This property has the options listed in the following table.
 - True - The destination path is stored in a variable.
 - False - The destination path is specified in a File connection manager.
- Overwrite Destination
 - Specify whether the operation can overwrite files in the destination directory.
- Name
 - Provide a unique name for the File System task. This name is used as the label in the task icon. Task names must be unique within a package.
- Description
 - Type a description of the File System task.



 Capgemini
connecting people through experience

File System Task

■ Operation

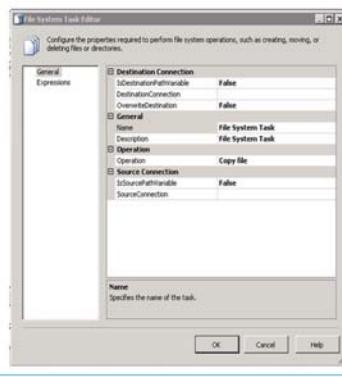
- Select the file-system operation to perform. This property has the options listed in the following table.

■ Is Source Path Variable

- Indicate whether the destination path is stored in a variable. This property has the options listed in the following.
 - True - The destination path is stored in a variable.
 - False - The destination path is specified in a File connection manager.

■ Description

- Type a description of the File System task.

Copyright © Capgemini 2010. All Rights Reserved

FTP Task

- The FTP task downloads and uploads data files and manages directories on servers.
- FTP task can be for the following purposes:

Creates a folder on the FTP server.

Sends a file from the local computer to the FTP server.

Saves a file from the FTP server to the local computer.

Deletes a folder on the local computer.

Deletes a folder on the FTP server.

Deletes a file on the local computer.

Deletes a file on the FTP server.



Copyright © Capgemini 2012. All Rights Reserved.

The FTP Task works very similarly to any FTP utility. It allows us to send and receive files from an FTP location along with other FTP commands.

Send Mail Task

- The Send Mail Task sends email via Simple Mail Transfer Protocol (SMTP).
- The Send Mail Task is most commonly used as a notification tool.
- This allows to receive information about the package that can be passed into the mail task through variables.
- The Send Mail Task can be placed at the end of a Control Flow to send email on the successful
- The Send Mail Task can also be used to send files, because it has the ability to send attachments.

Send Mail Task

- Can configure the Send Mail task in the following ways:

Provide the message text for the e-mail message.

Provide a subject line for the e-mail message.

Set the priority level of the message.

Specify the recipients on the To, Cc, and Bcc lines.



Copyright © Capgemini 2010. All Rights Reserved.

SQL Server Tasks

Lesson 11:

Bulk Insert Task

- The Bulk Insert task provides an efficient way to copy large amounts of data into a SQL Server table or view.
- It can transfer data only from a text file into a SQL Server table or view.
- The destination must be a table or view in a SQL Server database.
- If the text file is located on the same computer as the SQL Server database into which data is inserted, the copy operation occurs at an even faster rate because the data is not moved over the network.



Copyright © Capgemini 2010. All Rights Reserved.

The Bulk Insert task provides an efficient way to copy large amounts of data into a SQL Server table or view. For example, suppose your company stores its million-row product list on a mainframe system, but the company's e-commerce system uses SQL Server to populate Web pages. You must update the SQL Server product table nightly with the master product list from the mainframe. To update the table, you save the product list in a tab-delimited format and use the Bulk Insert task to copy the data directly into the SQL Server table.

The Bulk Insert task can transfer data only from a text file into a SQL Server table or view. To use the Bulk Insert task to transfer data from other database management systems (DBMSs), you must export the data from the source to a text file and then import the data from the text file into a SQL Server table or view.

Execute SQL Task

The Execute SQL task runs SQL statements or stored procedures from a package. The task can contain either a single SQL statement or multiple SQL statements that run sequentially.

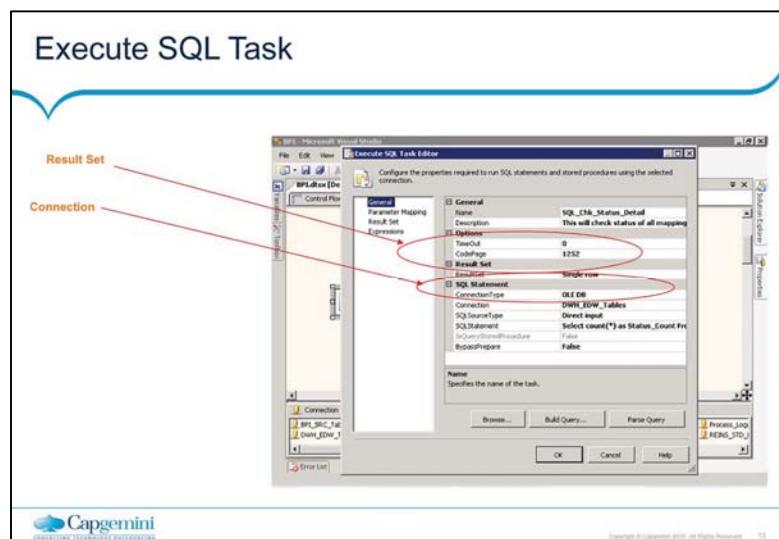
- Execute SQL task can be used for the following purposes:
- Truncate a table or view in preparation for inserting data.
- Create, alter, and drop database objects such as tables and views.
- Re-create fact and dimension tables before loading them.
- Run stored procedures.
- Save the row set returned from a query into a variable.

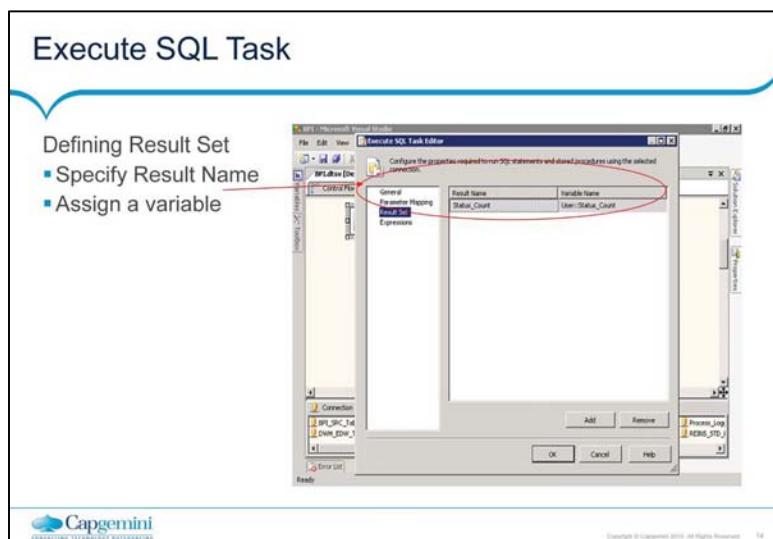


Copyright © Capgemini 2010. All Rights Reserved.

Execute SQL Task

- Specify the type of connection manager to use to connect to a database.
- Specify the type of result set that the SQL statement returns.
- Specify a time-out for the SQL statements.
- Specify the source of the SQL statement.
- Indicate whether the task skips the prepare phase for the SQL statement. If using the ADO connection type, indicate whether the SQL statement is a stored procedure. For other connection types this property is read-only and its value is always false.





Transfer Database Task

- The Transfer Database task transfers a SQL Server database between two instances of SQL Server.
- This task supports two modes of database transfer - namely DatabaseOnline and DatabaseOffline.
- The Transfer Database task can also be configured to permit overwriting a destination database that has the same name, replacing the destination database.



Copyright © Capgemini 2010. All Rights Reserved. 12

The Transfer Database task transfers a SQL Server database between two instances of SQL Server. In contrast to the other tasks that only transfer SQL Server objects by copying them, the Transfer Database task can either copy or move a database. The task can copy a database between instances of SQL Server 2000, instances of SQL Server, or one of each. This task can also be used to copy a database within the same server. The database can be transferred by using online or offline mode. When you use online mode, the database remains attached and it is transferred by using SQL Management Object (SMO) to copy the database objects. When you use offline mode, the database is detached, the database files are copied or moved, and the database is attached at the destination after the transfer finishes successfully. If the database is copied, it is automatically reattached at the source if the copy is successful. In offline mode, the database is copied more quickly, but the database is unavailable to users during the transfer.

Offline mode requires that you specify the network file shares on the source and destination servers that contain the database files. If the folder is shared and can be accessed by the user, you can reference the network share using the syntax `\computername\Program Files\myfolder`. Otherwise, you must use the syntax `\computername\c$\Program Files\myfolder`. To use the latter syntax, the user must have write access to the source and destination network shares.

Transfer Error Messages Task

- The Transfer Error Messages task transfers one or more SQL Server user-defined error messages between instances of SQL Server.
- User-defined messages are messages with an identifier that is equal to or greater than 50000
- The Transfer Error Messages task can be configured to transfer all error messages, or only the specified error messages
- An error message is defined as a duplicate error message if the identifier and the language are the same.
- The Transfer Error Messages task can be configured to handle existing error messages in the following ways:

Overwrite existing jobs.



Fail the task when duplicate jobs exist.



Skip duplicate jobs.



Copyright © Capgemini 2010. All Rights Reserved.

The Transfer Error Messages task transfers one or more SQL Server user-defined error messages between instances of SQL Server. User-defined messages are messages with an identifier that is equal to or greater than 50000. Messages with an identifier less than 50000 are system error messages, and cannot be transferred by using the Transfer Error Messages task.

The Transfer Error Messages task can be configured to transfer all error messages, or only the specified error messages. User-defined error messages may be available in a number of different languages and the task can be configured to transfer only messages in selected languages.

The **sysmessages** table in the **master** database contains all the error messages—both system and user-defined—that SQL Server uses.

The user-defined messages to be transferred may already exist on the destination. An error message is defined as a duplicate error message if the identifier and the language are the same. The Transfer Error Messages task can be configured to handle existing error messages in the following ways:

Overwrite existing error messages.

Fail the task when duplicate messages exist.

Skip duplicate error messages.

Transfer Jobs Task

- The Transfer Jobs task transfers one or more SQL Server Agent jobs between instances of SQL Server.
- The Transfer Jobs task can be configured to transfer all jobs, or only specified jobs. You can also indicate whether the transferred jobs are enabled at the destination.
- The jobs to be transferred may already exist on the destination. The Transfer Jobs task can be configured to handle existing jobs in the following ways:

Overwrite existing jobs.



Fail the task when duplicate jobs exist.



Skip duplicate jobs.

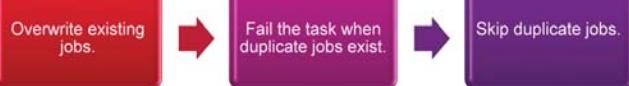


Copyright © Capgemini 2010. All Rights Reserved. 17

We can use SQL Server Integration Services to transfer the jobs from SQL 2005 to another SQL 2005 or SQL 2008.

Transfer Login Task

- The Transfer Logins task transfers one or more logins between instances of SQL Server.
- The Transfer Logins task supports a source and destination that is SQL Server 2000 or SQL Server..
- The Transfer Logins task can be configured to transfer all logins, only specified logins, or all logins that have access to specified databases only.
- The logins to be transferred may already exist on the destination. The Transfer Logins task can be configured to handle existing logins in the following ways:

Copyright © Capgemini 2010. All Rights Reserved. 12

We can use SQL Server Integration Services to **Transfer Login** from SQL 2005 to another SQL 2005 or SQL 2008 using **Transfer Login Task**. The Transfer Logins task supports a source and destination that is SQL Server 2000 or SQL Server. There are no restrictions on which of the two versions to use as a source or destination.

Transfer Master Stored Procedures Task & Transfer SQL Server Objects Task

▪ Transfer Master Stored Procedures Task

- The Transfer Master Stored Procedures task transfers one or more user-defined stored procedures between master databases on instances
- The Transfer Master Stored Procedures task can be configured to transfer all stored procedures or only specified stored procedures. This task does not copy system stored procedures.

▪ Transfer SQL Server Objects Task

- The Transfer SQL Server Objects task transfers one or more types of objects in a SQL Server database between instances of SQL Server.



Copyright © Capgemini 2010. All Rights Reserved. 10

The Transfer Master Stored Procedures task transfers one or more user-defined stored procedures between **master** databases on instances of SQL Server. To transfer a stored procedure from the **master** database, the owner of the procedure must be **dbo**.

The Transfer Master Stored Procedures task can be configured to transfer all stored procedures or only specified stored procedures. This task does not copy system stored procedures.

As other transfer task in **Transfer Master Stored Procedures Task**

The master stored procedures be transferred may already exist on the destination. The Transfer Master Stored Procedures task can be configured to handle existing stored procedures in the following ways:

Overwrite existing stored procedures.

Fail the task when duplicate stored procedures exist.

Skip duplicate stored procedures.

Transfer SQL Server Objects Task

The Transfer SQL Server Objects task transfers one or more types of objects in a SQL Server database between instances of SQL Server. For example, the task can copy tables and stored procedures. Depending on the version of SQL Server that is used as a source, different types of objects are available to copy. For example, only a SQL Server database includes schemas and user-defined aggregates.

Like other Transfer task **Transfer SQL Server Objects Task**

can be configured to transfer all objects, all objects of a type, or only specified objects of a type. For example, you can choose to copy only selected tables in the **AdventureWorks** database.

And if the Transfer SQL Server Objects task transfers tables, you can specify the types of table-related objects to copy with the tables. For example, you can specify that primary keys are copied with tables.

List of objects those can be transfer are:

Tables

Views

Stored Procedures

User-Defined Functions

Defaults

User-Defined Data Types

Partition Functions

Partition Schemes

Schemas

Assemblies

User-Defined Aggregates

User-Defined Types

XML Schema Collection

Scripting Tasks

Lesson 12:

Scripting Tasks

- Script Task
- The Script task provides code to perform functions that are not available in the built-in tasks and transformations that SQL Server Integration Services provides.
- This task can accomplish anything that can be done with any .NET programming. The Script task can also combine functions in one script instead of using multiple tasks and transformations.



Copyright © Capgemini 2012. All Rights Reserved.

When we create a new SQL Server Integration Services (SSIS) package we find ourselves wanting some functionality that the built-in tasks cannot accomplish. This situation is where the Script Task comes into play.

Data Profiling Tasks

Lesson 13:

Data Profiling Task

- "The Data Profiling Task The task lets you analyze data in a SQL Server database and, from the results of that analysis, generate XML reports that can be saved to a file or an SSIS variable "
- By configuring one or more of the task's profile types, you can generate a report that provides details such as a column's minimum and maximum values, or the number and percentage of null values.
- By accessing the results generated by the Data Profiling task, you can design a workflow that automatically determines the appropriate actions to take based on the validity of the source data.



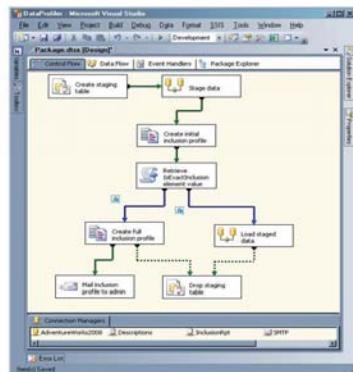
Copyright © Capgemini 2010. All Rights Reserved.



Data Profiling Task

Example

- For our example data profiling task, let's suppose we're extracting product descriptions (such as colors, sizes, and other details) from a text file and want to ensure that the product IDs in that file match the product IDs in the target database before loading the description data. We can stage the data in a SQL Server table and then use the Data Profiling task to generate a Value Inclusion report, which will tell us whether the source IDs are valid. After we determine the data's validity, we can then take the appropriate actions within our SSIS package. Figure shows what such a solution might look like, as it appears in the Control Flow tab of SSIS Designer (in Business Intelligence Development Studio).



Copyright © Capgemini 2010. All Rights Reserved.

14

Data Profiling Task

Solution

- In this solution, we first stage the data because the Data Profiling task requires that the data be in a SQL Server database in order to perform the analysis. Next, we generate the Value Inclusion report (saved to a variable), and then use a Script task to extract the analysis results from that variable. If one or more IDs are invalid, we generate the report as an XML document and email it to an administrator. If all the IDs are valid, we load the data into the target database



Copyright © Capgemini 2010. All Rights Reserved.



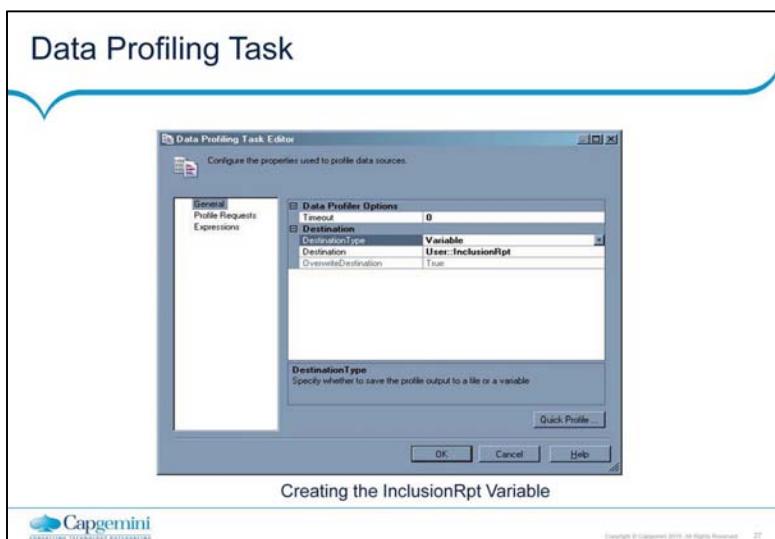
Data Profiling Task

Generating the Inclusion Profile

- The next step is to add the Data Profiling task and connect the precedence constraint from the Data flow task to the Data Profiling task. When you set up a Data Profiling task, you can configure it to save data to an XML file or to an SSIS string variable. In this case, I want to save the report information to a variable so I can later use that variable in the Script task.
- Ultimately, the package will require two string variables—one to hold the report, and a second one to hold the analysis results retrieved through the Script task (this process is explained later in the article). So before configuring the Data Profiling task, we need to create these two string variables, both of which will be configured at the package scope:
 - 1.InclusionRpt, to hold the Value Inclusion report
 - 2.Results, to hold the analysis results retrieved through the Script task.
- When configuring the Data Profiling task, select Variable as the destination type and then specify the InclusionRpt variable as the destination.

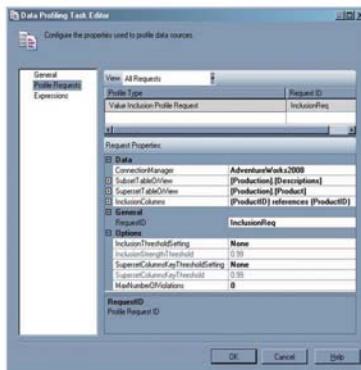


Copyright © Capgemini 2010. All Rights Reserved.



Data Profiling Task

- After specifying our variable, we can now start configuring the Value Inclusion profile.
- Click to the Profile Requests page, and select Value Inclusion Profile Request in the Profile Type drop-down list (in the first cell of the grid). You can then configure the data profile in the bottom section of the Profile Requests page, as shown in the Figure :



Workflow Tasks

Lesson 14:

Execute Package Task

- Execute Package Task can run other packages as part of a workflow.
- A package that runs other packages is generally referred to as the parent package, and the packages that a parent workflow runs are called child packages.
- The child package can be run in the process of the parent package, or it can be run in its own process.
- The Execute Package task can run packages stored in the SQL Server msdb database and packages stored in the file system.
- Can use the Execute Package task for the following purposes:



Copyright © Capgemini 2010. All Rights Reserved.

The Execute Package task extends the enterprise capabilities of Integration Services by letting packages run other packages as part of a workflow.

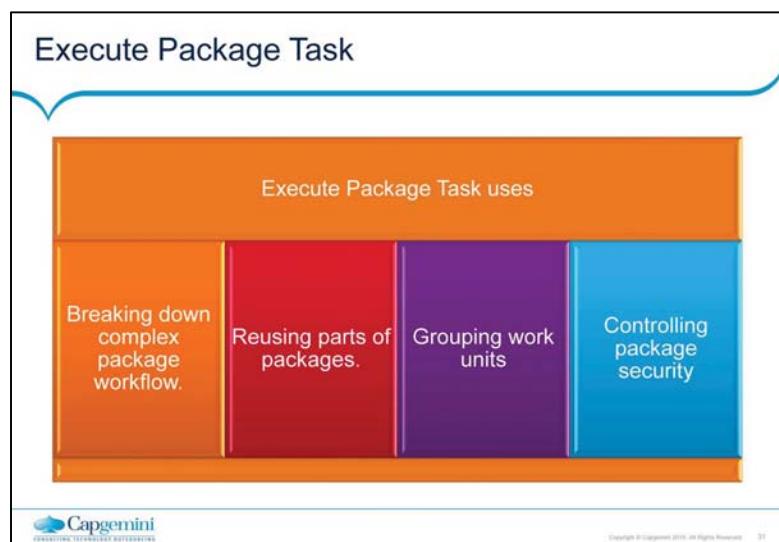
You can use the Execute Package task for the following purposes:

Breaking down complex package workflow. This task lets you break down workflow into multiple packages, which are easier to read, test, and maintain. For example, if you are loading data into a star schema, you can build a separate package to populate each dimension and the fact table.

Reusing parts of packages. Other packages can reuse parts of a package workflow. For example, you can build a data extraction module that can be called from different packages. Each package that calls the extraction module can perform different data scrubbing, filtering, or aggregation operations.

Grouping work units. Units of work can be encapsulated into separate packages and joined as transactional components to the workflow of a parent package. For example, the parent package runs the accessory packages, and based on the success or failure of the accessory packages, the parent package either commits or rolls back the transaction.

Controlling package security. Package authors require access to only a part of a multipackage solution. By separating a package into multiple packages, you can provide a greater level of security, because you can grant an author access to only the relevant packages.



Workflow Tasks.....

- Execute Process Task
- The Execute Process task runs an application or batch file as part of a SQL Server Integration Services package workflow.
- Execute Process task can open any standard application.
- When the Execute Process task runs a command-line application, it provides input to the application through a variable
- Message Queue Task
- The Message Queue task allows you to use Message Queuing (also known as MSMQ) to send and receive messages between SQL Server Integration Services packages, or to send messages to an application queue that is processed by a custom application.



Copyright © Capgemini 2007. All Rights Reserved. 17

The Execute Process task runs an application or batch file as part of a SQL Server Integration Services package workflow. Although we can use the Execute Process task to open any standard application, such as Microsoft Excel or Microsoft Word, you typically use it to run business applications or batch files that work against a data source. For example, you can use the Execute Process task to expand a compressed text file. Then the package can use the text file as a data source for the data flow in the package. As another example, you can use the Execute Process task to run a custom Visual Basic application that generates a daily sales report.

Message Queue Task

You can use the Message Queue task for the following purposes:

Delaying task execution until other packages check in. For example, after nightly maintenance at each of your retail sites, a Message Queue task sends a message to your corporate computer. A package running on the corporate computer contains Message Queue tasks, each waiting for a message from a particular retail site. When a message from a site arrives, a task uploads data from that site. After all the sites have checked in, the package computes summary totals.

Sending data files to the computer that processes them. For example, the output from a restaurant cash register can be sent in a data file message to the corporate payroll system, where data about each waiter's tips is extracted.

Distributing files throughout your enterprise. For example, a package can use a Message Queue task to send a package file to another computer. A package running on the destination computer then uses a Message Queue task to retrieve and save the package locally.

Workflow Tasks.....

The diagram consists of three concentric circles. The outermost circle is purple, the middle is blue, and the innermost is teal. To the right of the circles is a vertical stack of three rectangular boxes. The top box is labeled "Message Queue Uses". The middle box contains the text "Delaying task execution until other packages check in". The bottom box contains the text "Sending data files to the computer that processes them".

Message Queue Uses
Delaying task execution until other packages check in
Sending data files to the computer that processes them

Capgemini
INTEGRATION SERVICES EXPERTS

Copyright © Capgemini 2010. All Rights Reserved.

Processing XML

Lesson 15:

XML Task

- The XML task can retrieve XML documents, apply operations to the documents, merge multiple documents, or validate, compare, and save the updated documents to files and variables.

XML Task uses:			
Dynamically modify XML documents at run time.	Reformat an XML document.	Select sections of an XML document	Merge documents from many source

 Capgemini
INNOVATING TECHNOLOGIES. INNOVATING LIVES.
Copyright © Capgemini 2010. All Rights Reserved.

The Web Service task executes a Web service method.

Web Service task can be used for the following purposes:

Writing to a variable the values that a Web service method returns. For example, you could obtain the highest temperature of the day from a Web service method, and then use that value to update a variable that is used in an expression that sets a column value.

Writing to a file the values that a Web service method returns. For example, a list of potential customers can be written to a file and the file then used as a data source in a package that cleans the data before it is written to a database.

Analysis Services Tasks

Lesson 16:

Analysis Services Tasks

- Analysis Services Tasks within SSIS can be used to create, modify, delete and process Analysis Services objects.
- The different types of Analysis Services tasks are as follows:
- Analysis Services Execute DDL Task : It is similar to the Execute SQL Task, however using the Analysis Services Execute DDL Task you can issue Data Definition Language statements against an Analysis Services system. The DDL statements can be used to create cubes, dimensions, KPI's or any other analytical processing of OLAP objects.
- The Analysis Services Processing Task : It can be used to process analysis services objects such as cubes, dimensions and mining models.
- The Data Mining Query Task : It can be used to run prediction queries based on data mining models built into analysis services. A prediction query creates a prediction for new data by using the mining model chosen.

Maintenance Tasks

Lesson 17 :

Maintenance Tasks

- SQL Server Integration Services includes a set of tasks that perform database maintenance functions. These tasks are commonly used in database maintenance plans, but the tasks can also be included in SSIS packages.



Execute T-SQL Statement Task and Update Statistics Task

- Execute T-SQL statement task
- This task is similar to the Execute SQL Task, however it only supports Transact SQL Statements. It should be used for SQL Server specific SQL statements.
- You cannot use this task to run statements on servers that use other dialects of the SQL language. If you need to run parameterized queries, save the query results to variables, or use property expressions, you should use the Execute SQL task instead of the Execute T-SQL Statement task.
- Update Statistics Task
- This task will update the statistics for one of many tables in one or many databases.

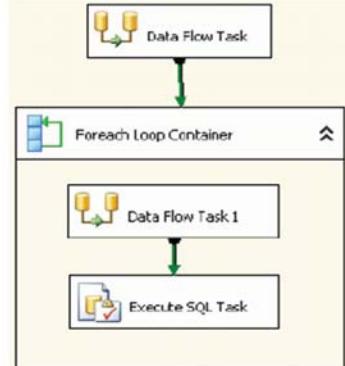
Copyright © Capgemini 2010. All Rights Reserved.

Container

Lesson 18:

ForEach Loop Container

- SQL Server 2008 Integration Services (SSIS) includes the For each Loop container, a control flow element that makes it simple to include a looping construct that enumerates files and objects in the control flow of a package.
- The For each Loop container provides no functionality; it provides only the structure in which you build the repeatable control flow, specify an enumerator type, and configure the enumerator. To provide container functionality, you must include at least one task in the For each Loop container.
- The Foreach Loop container can include a control flow with multiple tasks and other containers. Adding tasks and containers to a Foreach Loop container is similar to adding them to a package, except you drag the tasks and containers to the Foreach Loop container instead of to the package. If the Foreach Loop container includes more than one task or container, you can connect them using precedence constraints just as you do in a package.



For Each Loop Container

SQL Server 2008 Integration Services (SSIS) provides the following enumerator types:

- For Each ADO enumerator to enumerate rows in tables. For example, you can get the rows in an ADO record set.
- For Each ADO.NET Schema Row set enumerator to enumerate the schema information about a data source. For example, you can enumerate and get a list of the tables in the Adventure Works SQL Server database.
- For Each Node list enumerator to enumerate the result set of an XML Path Language (XML Path) expression. For example, this expression enumerates and gets a list of all the authors in the classical period:
`/authors/author[@period='classical']`.
- For Each SMO enumerator to enumerate SQL Server Management Objects (SMO) objects. For example, you can enumerate and get a list of the views in a SQL Server database.



Copyright © Capgemini 2010. All Rights Reserved. 43

For Each Loop Container

- For each File enumerator to enumerate files in a folder. The enumerator can traverse subfolders. For example, you can read all the files that have the *.log file name extension in the Windows folder and its subfolders.
- For Each From Variable enumerator to enumerate the enumerable object that a specified variable contains. The enumerable object can be an array, an ADO.NET Data Table, an Integration Services enumerator, and so on. For example, you can enumerate the values of an array that contains the name of servers.
- For Each Item enumerator to enumerate items that are collections. For example, you can enumerate the names of execu tables and working directories that an Execute Process task uses.

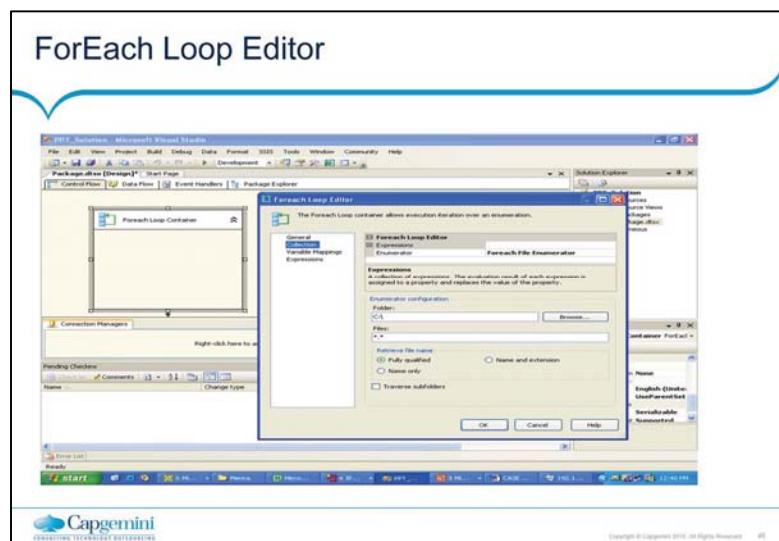


Copyright © Capgemini 2010. All Rights Reserved. 44

ForEach Loop Container

Enumerator	Configuration requirements
ForEach ADO	Specify the ADO object source variable and the enumerator Mode.
ForEach ADO.NET Schema RowSet	Specify the connection to a database and the schema to enumerate.
ForEach File	Specify a folder and the files to enumerate, the format of the file name of the retrieved files, and whether to traverse subfolders.
ForEach From Variable	Specify the variable that contains the objects to enumerate.
ForEach Item	Define the items in the Foreach Item collection, including columns and column data types.
ForEach Node list	Specify the source of the XML document and configure the XPath operation.
ForEach SMO	Specify the connection to a database and the SMO objects to enumerate.

Copyright © Capgemini 2010. All Rights Reserved



ForEach Loop Editor

- Use the General page of the For each Loop Editor dialog box to name and describe a For each Loop container that uses a specified enumerator to repeat a workflow for each member in a collection.
- Use the Collection page of the For each Loop Editor dialog box to specify the enumerator type and configure the enumerator.
- Use the Variables Mappings page of the For each Loop Editor dialog box to map variables to the collection value. The value of the variable is updated with collection values on each iteration of the loop.
- Lets look at the two most commonly used enumerator types:
- For Each File Enumerator
- For Each ADO Enumerator
-



Copyright © Capgemini 2010. All Rights Reserved. 47

ForEach File Enumerator

- In case of 'File enumerator' – Specify the folder path, the for each loop will run for the number of files in this folder.
- If the 'files' option: *.* will execute the for each loop for all files.
*.txt will execute the for each loop for all text files.
- Fully qualified
 - Select to retrieve the fully qualified path of file names.
- Name only
 - Select to retrieve only the file names.
- Name and extension
 - Select to retrieve the file names and their file name extensions.
 - Traverse Subfolders
 - Select to include the subfolders in the enumeration.

Copyright © Capgemini 2010. All Rights Reserved. 49

ForEach ADO Enumerator

- ADO object source variable

- Select a user-defined variable in the list, or click <New variable...> to create a new variable. The variable must have the Object data type, otherwise an error occurs.

- Rows in first table

- Select to enumerate only rows in the first table.

- Rows in all tables (ADO.NET dataset only)

- Select to enumerate rows in all tables. This option is available only if the objects to enumerate are all members of the same ADO.NET dataset.

- All tables (ADO.NET dataset only)

- Select to enumerate tables only.



Copyright © Capgemini 2010. All Rights Reserved.

For loop and sequence container

▪ For loop

- The For Loop container defines a repeating control flow in a package. In each repeat of the loop, the For Loop container evaluates an expression and repeats its workflow until the expression evaluates to False.
- The For Loop container uses the following elements to define the loop:
 - An optional initialization expression that assigns values to the loop counters.
 - An evaluation expression that contains the expression used to test whether the loop should stop or continue.
 - An optional iteration expression that increments or decrements the loop counter.
 - Sequence container
- Sequential Task groups the tasks into multiple control flows and executes the process sequentially. We can use this task widely based on our requirement like, disabling a sequence when it should not process, use it when managing multiple tasks at a same time in one location.

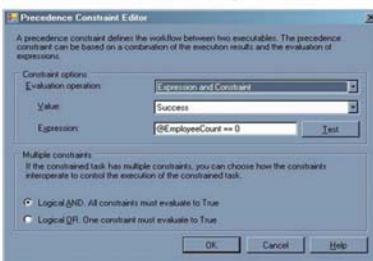


Precedence Constraint

Lesson 19:

Configuring the Precedence Constraint

- Besides the regular Success and Failure constraints, can define precedence constraint Workflow by Expressions
- To add an expression, double-click the precedence constraint to open the Precedence Constraint Editor dialog box, as shown in below figure.



 Capgemini
integrating people and processes

Copyright © Capgemini 2007. All Rights Reserved. 17

Configuring the Precedence Constraint..

- To add an expression to a precedence constraint, the first step is to select Evaluation operation. And Evaluation operation has following options:

Constraint:

- The precedence constraint is evaluated solely on the option selected in the Value property. The precedence constraint will evaluate to true only if the precedence executable runs successfully.

Expression:

- The precedence constraint is evaluated based on the expression defined in the Expression text box. If the expression evaluates to true, the workflow continues and the constrained executable runs.

Expression and Constraint:

- The precedence constraint is evaluated based on both the Value property and the expression. Both must evaluate to true for the constrained executable to run.

Expression or Constraint:

- The precedence constraint is evaluated based on either the Value property or the expression. At least one of these properties must evaluate to true for the constrained executable to run.



Copyright © Capgemini 2010. All Rights Reserved

To add an expression to a precedence constraint, the first step is to select Evaluation operation. And Evaluation operation has following options:

Constraint: The precedence constraint is evaluated solely on the option selected in the Value property. For example, if Constraint is selected as the Evaluation operation option and Success is selected as the Value option (the default settings for both properties), the precedence constraint will evaluate to true only if the precedence executable runs successfully. When the precedence constraint evaluates to true, the workflow continues and the constrained executable runs. When the Constraint option is selected, the Expression property is greyed out.

Expression: The precedence constraint is evaluated based on the expression defined in the Expression text box. If the expression evaluates to true, the workflow continues and the constrained executable runs. If the expression evaluates to false, the constrained executable does not run. When the Expression option is selected, the Value property is greyed out.

Expression and Constraint: The precedence constraint is evaluated based on both the Value property and the expression. Both must evaluate to true for the constrained executable to run.

Expression or Constraint: The precedence constraint is evaluated based on either the Value property or the expression. At least one of these properties must evaluate to true for the constrained executable to run.

Define precedence constraint Workflow by Logical AND or Logical OR

Precedence Constraint Editor dialog box has following two options at the bottom :

Logical AND: All precedence constraints that point to the constrained executable must evaluate to true in order for that executable to run. This is the default option. If it is selected, the arrow is solid.

Logical OR: Only one precedence constraint that points to the constrained executable must evaluate to true in order for that executable to run. If this option is selected, the arrow is dotted.

Configuring the Precedence Constraint..

- Define precedence constraint Workflow by Logical AND or Logical OR:
- Precedence Constraint Editor dialog box has following two options at the bottom:

Logical AND

- All precedence constraints that point to the constrained executable must evaluate to true in order for that executable to run. This is the default option. If it is selected, the arrow is solid.

Logical OR:

- Only one precedence constraint that points to the constrained executable must evaluate to true in order for that executable to run. If this option is selected, the arrow is dotted.



Copyright © Capgemini 2010. All Rights Reserved. 14

To add an expression to a precedence constraint, the first step is to select Evaluation operation. And Evaluation operation has following options:

Constraint: The precedence constraint is evaluated solely on the option selected in the Value property. For example, if Constraint is selected as the Evaluation operation option and Success is selected as the Value option (the default settings for both properties), the precedence constraint will evaluate to true only if the precedence executable runs successfully. When the precedence constraint evaluates to true, the workflow continues and the constrained executable runs. When the Constraint option is selected, the Expression property is greyed out.

Expression: The precedence constraint is evaluated based on the expression defined in the Expression text box. If the expression evaluates to true, the workflow continues and the constrained executable runs. If the expression evaluates to false, the constrained executable does not run. When the Expression option is selected, the Value property is greyed out.

Expression and Constraint: The precedence constraint is evaluated based on both the Value property and the expression. Both must evaluate to true for the constrained executable to run.

Expression or Constraint: The precedence constraint is evaluated based on either the Value property or the expression. At least one of these properties must evaluate to true for the constrained executable to run.

Define precedence constraint Workflow by Logical AND or Logical OR

Precedence Constraint Editor dialog box has following two options at the bottom :

Logical AND: All precedence constraints that point to the constrained executable must evaluate to true in order for that executable to run. This is the default option. If it is selected, the arrow is solid.

Logical OR: Only one precedence constraint that points to the constrained executable must evaluate to true in order for that executable to run. If this option is selected, the arrow is dotted.

Variables and Expressions

Lesson 20:

Variables

- Variables store values that a SQL Server Integration Services package and its containers, tasks, and event handlers can use at run time.
- Can use variables in Integration Services packages for the following purposes:

- Updating properties of package elements at run time
- Including an in-memory lookup table
- Loading variables with data values and then using them to specify a search condition in a WHERE clause.
- Loading a variable with an integer and then using the value to control looping within a package control flow.
- Populating parameter values for Transact-SQL statements at run time.
- Building expressions that include variable values.



Copyright © Capgemini 2010. All Rights Reserved. 10

➤ Variables store values that a SQL Server Integration Services package and its containers, tasks, and event handlers can use at run time. The scripts in the Script task and the Script component can also use variables.

- Can use variables in Integration Services packages for the following purposes:
- Updating properties of package elements at run time. For example, can dynamically set the number of concurrent executables that a Foreach Loop container allows.
 - Including an in-memory lookup table. For example, a package can run an Execute SQL task that loads a variable with data values.
 - Loading variables with data values and then using them to specify a search condition in a WHERE clause. For example, the script in a Script task can update the value of a variable that is used by a Transact-SQL statement in an Execute SQL task.
 - Loading a variable with an integer and then using the value to control looping within a package control flow. For example, can use a variable in the evaluation expression of a For Loop container to control iteration.
 - Populating parameter values for Transact-SQL statements at run time. For example, a package can run an Execute SQL task and then use variables to dynamically set the parameters in a Transact-SQL statement.
 - Building expressions that include variable values. For example, the Derived Column transformation can populate a column with the result obtained by multiplying a variable value by a column value..

➤ **Type of variable**

➤ SSIS supports following two types of variables:

- **System variables:** System variables are defined by Integration Services and cannot create additional system variables. System variables contain useful information about a package, container, task, or event handler. For example, at run time the **MachineName** system variable contains the name of the computer on which the package is running and **StartTime** the time the package started to run. System variables are read-only.
- **User-defined variables :** User-defined variables are defined by package developers, and You can create as many user-defined variables as a package requires.

➤ The names of user-defined and system variables are case sensitive.

Variables...

- Type of variable
- SSIS supports following two types of variables:

System variables

- System variables are defined by Integration Services and cannot create additional system variables. System variables contain useful information about a package, container, task, or event handler.

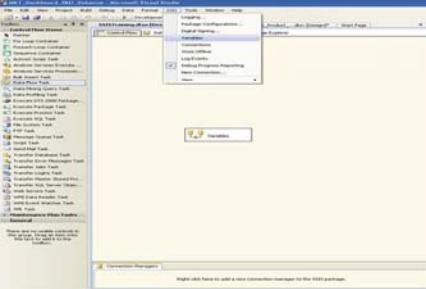
User-defined variables

- User-defined variables are defined by package developers, and You can create as many user-defined variables as a package requires.

- The names of user-defined and system variables are case sensitive.

Variables

- To add a variable
- On the SSIS menu, click Variables or right click on control flow and select variables:

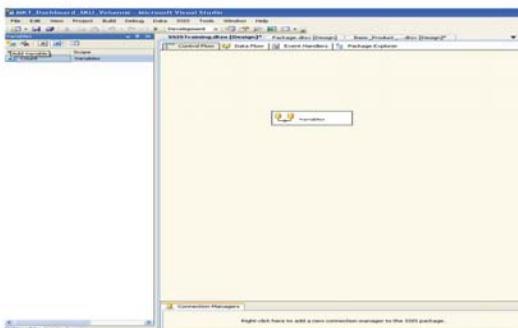


Capgemini
INTEGRATION SERVICES EXPERTS

Copyright © Capgemini 2010. All Rights Reserved.

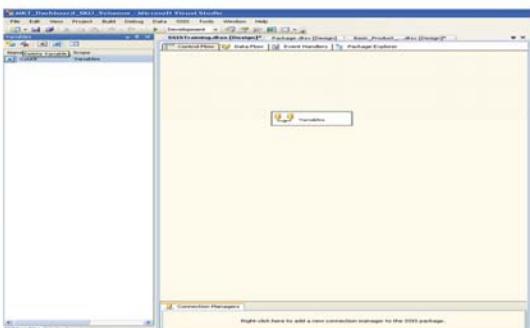
Variables

In the Variables window, click the Add Variable icon. The new variable is added to the list:



The screenshot shows the 'Variables' window in the SSIS IDE. A tooltip 'Add Variable' is displayed over the 'Add Variable' button in the toolbar. The 'Variables' list pane shows a single item named 'Variables'. The 'Properties' pane on the right shows the properties for the selected variable.

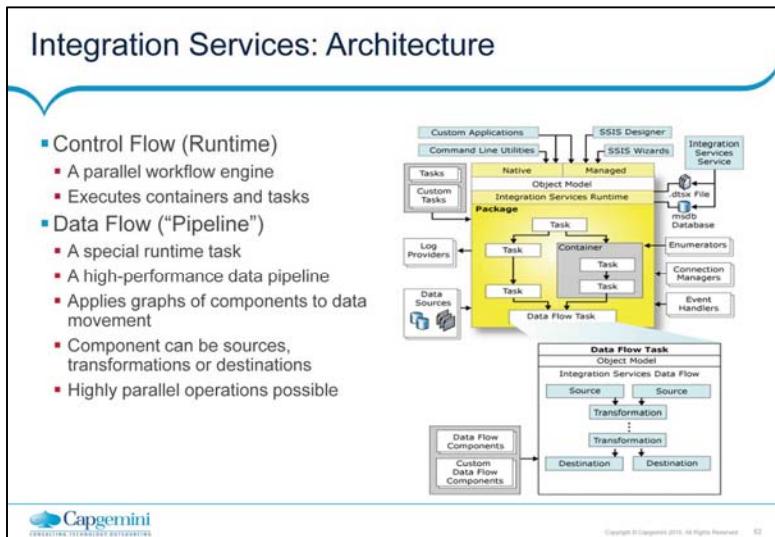
Variables

- To delete a variable

Copyright © Capgemini 2010. All Rights Reserved.

SSIS Architecture

Lesson 21:



Distinguishing between data flow pipeline and package runtime

- Runtime Engine :

- The SSIS Runtime Engine manages the workflow of the packages during runtime, which means its role is to execute the tasks in a defined sequence. As you know, you can define the sequence using precedence constraints. This engine is also responsible for providing support for event logging, breakpoints in the BIDS designer, package configuration, transactions and connections. The SSIS Runtime engine has been designed to support concurrent/parallel execution of tasks in the package.

- Dataflow Pipeline Engine:

- The Dataflow Pipeline Engine is responsible for executing the data flow tasks of the package. It creates a dataflow pipeline by allocating in-memory structure for storing data in-transit. This means, the engine pulls data from source, stores it in memory, executes the required transformation in the data stored in memory and finally loads the data to the destination. Like the SSIS runtime engine, the Dataflow pipeline has been designed to do its work in parallel by creating multiple threads and enabling them to run multiple execution trees/units in parallel. Deploying packages to the SSISDB

Executing packages on the client side or hosted in the SSIS service

- SSIS client components are installed when you install Workstation components - that gives you BIDS (with ability to design and debug SSIS projects and packages) and SSMS (with ability to connect to SSIS service and monitor package execution).
- To run packages in production you need SSIS server components, which are installed when you install Integration Services. That requires server license.
- choose Workstation components, Books Online and development tools.
- That'll install the following components:
 - .BOL
 - .Native client
 - .BIDS
 - .SQLXML4
 - .MSXML6
 - .OWCL11 (Office Windows Components)
 - .SQL Server Backward-Compatibility Files
- Run setup. Choose the Integration Services and Workstation components options.

Copyright © Capgemini 2010. All Rights Reserved. SP

Error Handling, Logging and Transactions

Lesson 22:

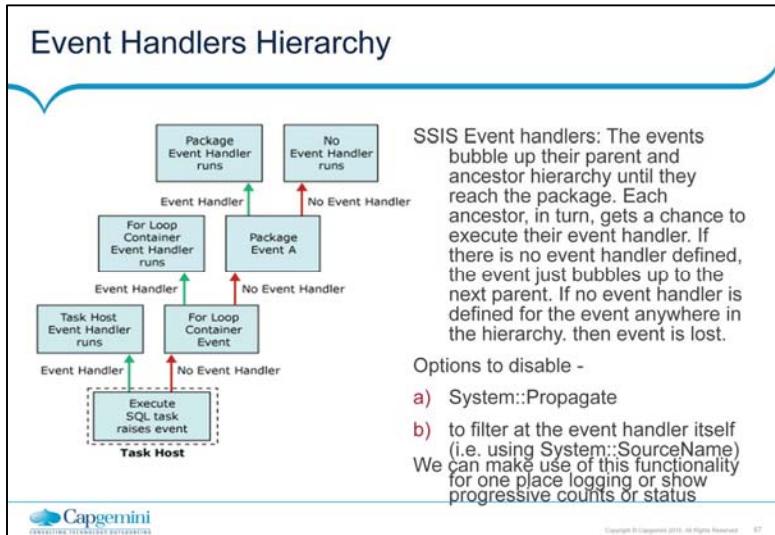
Integration Services Event Handlers

At run time, execu tables (packages and For each Loop, For Loop, Sequence, and task host containers) raise events. For example, an On Error event is raised when an error occurs. You can create custom event handlers for these events to extend package functionality and make packages easier to manage at run time. Event handlers can perform tasks such as the following:

- Clean up temporary data storage when a package or task finishes running.
- Retrieve system information to assess resource availability before a package runs.
- Refresh data in a table when a lookup in a reference table fails.
- Send an e-mail message when an error or a warning occurs or when a task fails.



Copyright © Capgemini 2010. All Rights Reserved.



Event Handler Configuration

Event handlers are members of an event handler collection, and all containers include this collection. If you create the package using SSIS Designer, you can see the members of the event handler collections in the Event Handlers folders on the Package Explorer tab of SSIS Designer.

- You can configure the event handler container in the following ways:
- Specify a name and description for the event handler.
Indicate whether the event handler runs, whether the package fails if the event handler fails, and the number of errors that can occur before the event handler fails.
- Specify an execution result to return instead of the actual execution result that the event handler returns at run time.
- Specify the transaction option for the event handler.
- Specify the logging mode that the event handler uses.



Copyright © Capgemini 2010. All Rights Reserved.

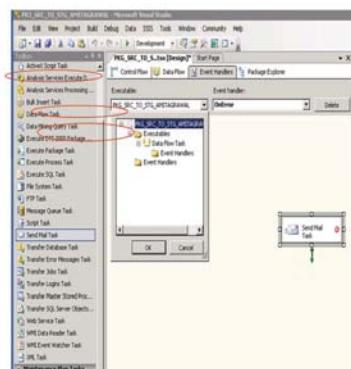


Event Handler Configuration

1) Execu tables: It define various execu tables in a project.

2) Event Handlers can be applied at various levels.

- a) Package level.
- b) Data Flow level.

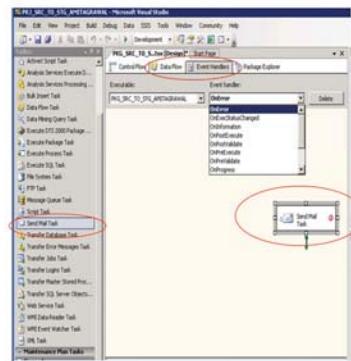


Copyright © Capgemini 2010. All Rights Reserved.

Event Handler Configuration

Event Handlers:

- There is drop down list we can select various events from it.
- Toolbox: We can select and assign various task by drag – drop to a event handler.
- For eg. In this snap shot we can see there is send mail task onError.



Copyright © Capgemini 2010. All Rights Reserved

Run Time Events

Event handler

On Error

Event

The event handler for the On Error event. This event is raised by an executable when an error occurs.

On Exec Status Changed The event handler for the On Exec Status Changed event. This event is raised by an executable when its execution status changes

On Information

The event handler for the On Information event. This event is raised during the validation and execution of an executable to report information. This event conveys information only, no errors or warnings.

On Post Execute

The event handler for the **On Post Execute** event. This event is raised by an executable immediately after it has finished running.

Run Time Events

On Post Validate

The event handler for the On Post Validate event. This event is raised by an executable when its validation is finished.

On Pre Execute

The event handler for the On Pre Execute event. This event is raised by an executable immediately before it runs.

On Pre Validate

The event handler for the On Pre Validate event. This event is raised by an executable when its validation starts.

On Progress

The event handler for the On Progress event. This event is raised by an executable when measurable progress is made by the executable.



Copyright © Capgemini 2010. All Rights Reserved.

Run Time Events

OnQueryCancel

The event handler for the On Query Cancel event. This event is raised by an executable to determine whether it should stop running.

OnTaskFailed

The event handler for the On Task Failed event. This event is raised by a task when it fails.

OnVariableValueChanged

The event handler for the On Variable Value Changed event. This event is raised by an executable when the value of a variable changes. The event is raised by the executable on which the variable is defined. This event is not raised if you set the Raise Change Event property for the variable to False.

OnWarning

The event handler for the On Warning event. This event is raised by an executable when a warning occurs.



Copyright © Capgemini 2010. All Rights Reserved. 79

Security

- Encryption - Using encryption to secure packages or parts of packages.
- Sensitive Data Protection - Identifying and protecting passwords and other sensitive data.
- Roles - SQL Server roles to control access to packages stored in SQL Server. Operator, Limited User, and Admin.
- Digital Signing - Code signing certificate to ensure a package hasn't changed.
- Define protection level appropriately to add security to package.

Execution , debugging and Protection

Lesson 23:

Debugging

- Expanded Error, Warning and Information Reporting
- Watch windows
- Data Viewers in the data flow task
- Breakpoints – Script task and package breakpoints integrated
- Error, warning and Information view window
- Log event viewing in log window
- All events logged

Restartability - Checkpoints

- Enables recovery from failed state
- Saves work at the task level
- Variable values saved
- Saves to XML temporary file
- Checkpoint file removed when package restarts
- Runtime feature, not dataflow restartability



Copyright © Capgemini 2012. All Rights Reserved.

Deployment

Lesson 24:

Difference between Project and Package Deployment Model

Project Deployment Model	Package Deployment Model
A project is the unit of deployment.	A package is the unit of deployment.
Parameters are used to assign values to package properties.	Configurations are used to assign values to package properties.
A project, containing packages and parameters, is built to a project deployment file (.ispac extension).	Packages (.dtsx extension) and configurations (.dtsConfig extension) are saved individually to the file system.
A project, containing packages and parameters, is deployed to the SSISDB catalog on an instance of SQL Server.	Packages and configurations are copied to the file system on another computer. Packages can also be saved to the MSDB database on an instance of SQL Server.

Copyright © Capgemini 2010. All Rights Reserved.

Switching between project and package deployment modes

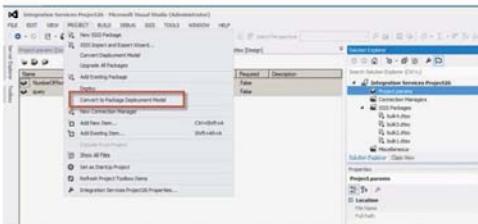
- SQL Server 2005 and 2008 use the legacy Package Deployment Model. This model is based on the package, which is the unit of deployment. However, SSIS packages usually interact with others and it is necessary to pass connection and parameters from one package to another. The new model is the Project Deployment Model which means deployment is per project and not per package.
- In SQL Server 2008, we store all the information in configuration files. Now we store the configuration information in project connections and project parameters. As per below we can switch between project and package deployment modes
 - Convert Project Deployment Model to legacy Package Deployment Model
 - Convert legacy Package Deployment Model to Project Deployment Model



Copyright © Capgemini 2010. All Rights Reserved.

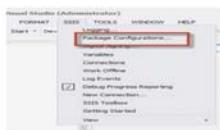
Convert Project Deployment Model to Legacy Package Deployment Model in SSIS

- Open SSDT and create a SSIS project. In the solution parameters, add some parameters to the project.
- In order to convert this Project Deployment Model to the legacy Package Deployment Model, go to Project > Convert to Package Deployment Model:



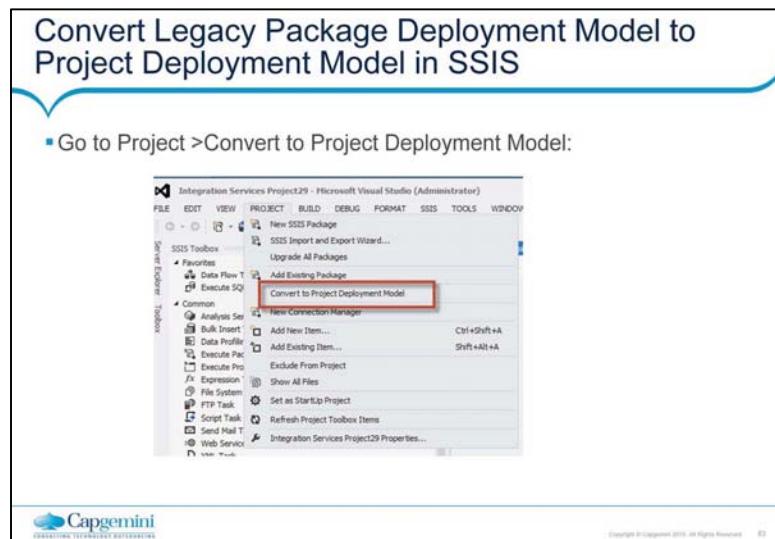
Convert Project Deployment Model to Legacy Package Deployment Model in SSIS

- You will receive an error message related to the compatibility check. It will verify if the package can be converted:
- if you check the messages, you will notice that these new features are not compatible with the Package Deployment Model. This is because Project Parameters are a new feature not supported in the old Package Deployment Model:
- Remove the project parameters in the package and try again. If everything is OK, the menu will show the Package Configurations option. This option is only available when you convert from Project Model to Package Model.



Capgemini
INNOVATE. INSPIRE. DIFFERENTIATE.

Copyright © Capgemini 2012. All Rights Reserved.



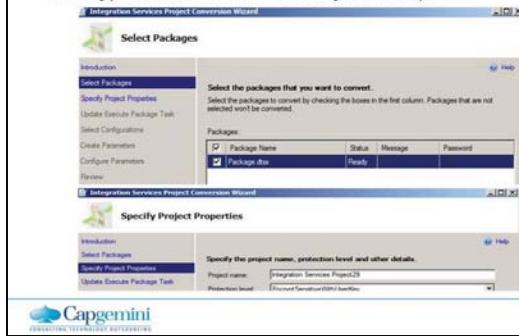
Convert Legacy Package Deployment Model to Project Deployment Model in SSIS

■ The Integration Services Project Conversion Wizard will be displayed:

Copyright © Capgemini 2010. All Rights Reserved. 17

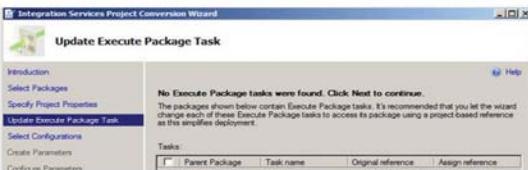
Convert Legacy Package Deployment Model to Project Deployment Model in SSIS

- Select the Package you want to convert
- Specify the project name and the protection level (by default, the EncryptSensitiveWithUserKey is used):



Convert Legacy Package Deployment Model to Project Deployment Model in SSIS

- The next step is to update the execution Tasks:



- You can select the existing configurations (the XML file that we created)



Copyright © Capgemini 2010. All Rights Reserved.

Convert Legacy Package Deployment Model to Project Deployment Model in SSIS

- In Create Parameters, we will create a project parameter to store the information of the XML configuration file of the previous step. You can assign a name and the scope for the parameter:



- Specify and configure the values of the parameter and then can see result. Once converted will notice in the menu that the option to create **Package Configuration** disappeared.



Copyright © Capgemini 2010. All Rights Reserved. 37

Running packages from SQL Server

- Four options available to execute an SSIS package:
- DTEXEC command line utility
- DTEXECUI windows application
- SSISDB
- SQL Server Agent

Execute Package using DTEXEC

SQL Server includes the command line tool DTEXEC.EXE which can be used to execute an SSIS package. DTEXEC can be run from a Command Prompt or from a batch (.BAT) file.

```
graph TD; A[Navigate to the Tutorial-Sample-1] --> B[Type the following command to execute the CreateSalesForecastInput.dtsx package]; B --> C[To see the complete list of command line options for DTEXEC type: DTEXEC /?]
```

D:\> DTEXEC /FILE CreateSalesForecastInput.dtsx
D:\> DTEXEC /?

Overview

SQL Server includes the command line tool DTEXEC.EXE which can be used to execute an SSIS package. DTEXEC can be run from a Command Prompt or from a batch (.BAT) file.

Explanation

To begin open a Command Prompt and navigate to the Tutorial-Sample-1 project folder as shown below:

It is not necessary to run DTEXEC from the folder where the SSIS package is located; it's just easier to change to the directory for demonstration purposes. Type the following command to execute the CreateSalesForecastInput.dtsx package:

DTEXEC /FILE CreateSalesForecastInput.dtsx

To see the complete list of command line options for DTEXEC type:

DTEXEC /?

Execute Package using DTEXECUI

The diagram illustrates the process of executing an SSIS package using the DTEXECUI application. It consists of four purple rounded rectangular boxes connected by downward-pointing arrows:

- Step 1: Navigate to the Tutorial-Sample-1 project folder.
- Step 2: Double-click on the SSIS package.
- Step 3: As a general rule you can simply click the Execute button to run your package.
- Step 4: Fine tune your execution by clicking through the various screens and entering your own settings.

The screenshot shows the 'Execute Package Utility' window. The left sidebar lists options: File, Open, Save, Import, Export, Command File, Connection Manager, Catalog Options, Run, Advanced, Logging, Set Values, Verification, and Command Line. The main area has sections for 'Package source' (File system), 'Server' (dropdown menu), 'Log on to the server' (radio buttons for 'Use Windows Authentication' and 'Use SQL Server Authentication'), 'User name' (text box), 'Password' (text box), and 'Package' (text box containing 'C:\mondioran\sql\Tutorial-Sample-1\CreateSalesForecastInput.dtsx'). At the bottom are 'Reset', 'Execute', and 'Close' buttons.

Capgemini
INNOVATING TECHNOLOGIES. INNOVATING LIVES.
Copyright © Capgemini 2010. All Rights Reserved.

Windows application to execute SSIS packages

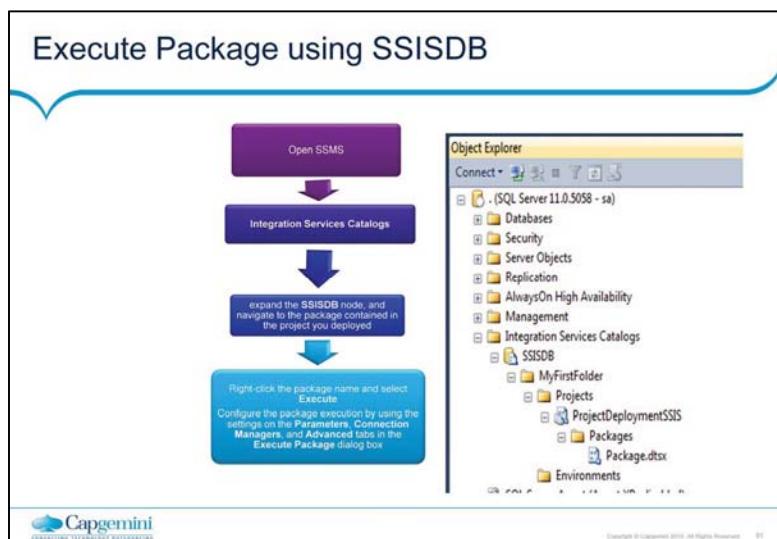
(DTEXECUI) Overview

SQL Server includes the Windows application DTEXECUI.EXE which can be used to execute an SSIS package. DTEXECUI provides a graphical user interface that can be used to specify the various options to be set when executing an SSIS package. You can launch DTEXECUI by double-clicking on an SSIS package file (.dtsx). You can also launch DTEXECUI from a Command Prompt then specify the package to execute.

Explanation

To begin open Windows Explorer (or My Computer) and navigate to the Tutorial-Sample-1 project folder. Double-click on the CreateSalesForecastInput.dtsx SSIS package and you will see the following multi-page dialog displayed:

As you can see there are many settings available when you use this utility. As a general rule you can simply click the Execute button to run your package. You can also fine tune your execution by clicking through the various screens and entering your own settings. After changing the settings click on Command Line which will show you the DTEXEC command line based on the settings you have chosen.



Open SQL Server Management Studio and connect to the instance of SQL Server that contains the Integration Services catalog.

In Object Explorer, expand the **Integration Services Catalogs** node, expand the **SSISDB** node, and navigate to the package contained in the project you deployed.

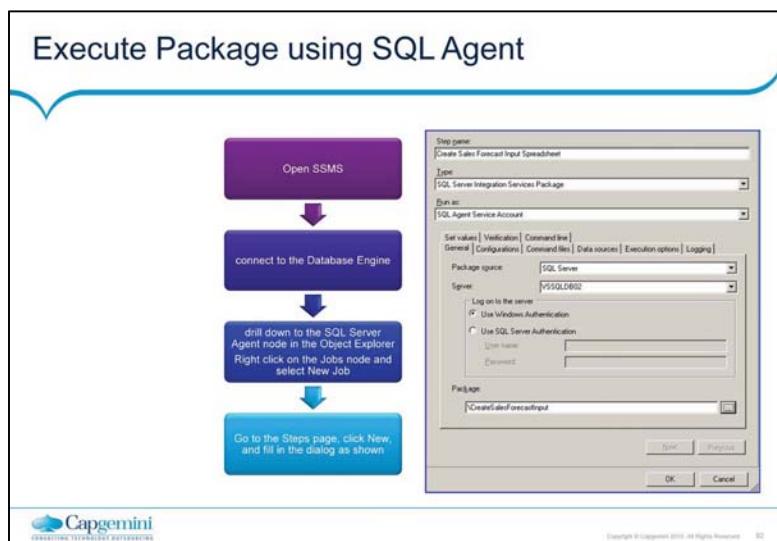
Right-click the package name and select **Execute**.

Configure the package execution by using the settings on the **Parameters**, **Connection Managers**, and **Advanced** tabs in the **Execute Package** dialog box.

Click **OK** to run the package.

-or-

Use stored procedures to run the package. Click **Script** to generate the Transact-SQL statement that creates an instance of the execution and starts an instance of the execution. The statement includes a call to the catalog.create_execution, catalog.set_execution_parameter_value, and catalog.start_execution stored procedures. For more information about these stored procedures



(SQL Server Agent) Overview

SQL Server Agent includes the SQL Server Integration Services Package job step type which allows you to execute an SSIS package in a SQL Server Agent job step. This can be especially handy as it allows you to schedule the execution of an SSIS package so that it runs without any user interaction.

Explanation

To begin open SSMS, connect to the Database Engine, and drill down to the SQL Server Agent node in the Object Explorer. Right click on the Jobs node and select New Job from the popup menu. Go to the Steps page, click New, and fill in the dialog as shown below:

In the example above the SSIS package to be executed is deployed to SQL Server (i.e. the MSDB database). You can also execute packages deployed to the file system or the SSIS package store.

Note that the Run as setting is the SQL Agent Service Account. This is the default setting although from a security standpoint it may not be what you want. You can setup a Proxy that allows you to give a particular credential permission to execute an SSIS package from a SQL Server Agent job step.

The first step to setting up the proxy is to create a credential (alternatively you could use an existing credential). Navigate to Security then Credentials in SSMS Object Explorer and right click to create a new credential as shown below:

Navigate to SQL Server Agent then Proxies in SSMS Object Explorer and right click to create a new proxy as shown below:

You must specify the credential and check SQL Server Integration Services Package. Now when you create or edit a SQL Server Agent job step, you can specify the proxy for the Run as setting as shown below:

Scheduling

Set up the proxy is to create a credentials

↓

Navigate to Security then Credentials

↓

Right click to create a new credential as shown

↓

Navigate to SQL Server Agent then Proxies in SSMS Object Explorer

↓

Right click to create a new proxy as shown

Credential name: SSIS_Prop
Identity: M55QLD802551SE\mc
Password:
Confirm password:

Proxy name: SSIS_Prop
Credential name: SSIS_Execution_Account
Description:

Select to the following subsystems:

- Agent
- Script
- Operating system (CmdExec)
- File and Filegroup
- Replication Merge
- Replication Queue Reader
- Replication Queue Writer
- Replication Transaction-Log Reader
- SQL Server Agent Command
- SQL Server Analysis Services Query
- SQL Server Integration Services Package

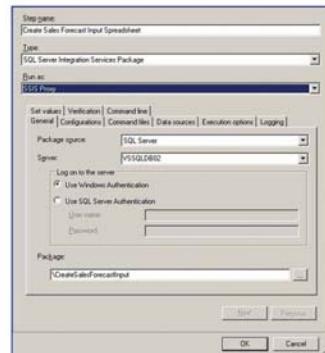
OK Cancel

 Capgemini
INNOVATION THROUGH INTEGRATION

Copyright © Capgemini 2010. All Rights Reserved.

Scheduling

Specify the credential and check SQL Server Integration Services Package. Now when you create or edit a SQL Server Agent job step, you can specify the proxy for the Run as setting as shown



Troubleshoot a SSIS package that doesn't run when you call the package from a SQL Server Agent job step

- The recommended methods for resolving this issue, including creating a proxy account, modifying the package Protection Level property setting, saving sensitive data in a package configuration file, and storing a package in the SQL Server msdb database.
- When you call a package from a SQL Server Agent job step and the package doesn't run, one of the following conditions is true:
 - The user account that runs the package as a job step differs from the original package author.

—or—
 - The user account does not have the required permissions to make connections or to access resources outside the package
 -



Copyright © Capgemini 2010. All Rights Reserved. 10

The recommended methods for resolving this issue, including creating a proxy account, modifying the package ProtectionLevel property setting, saving sensitive data in a package configuration file, and storing a package in the SQL Server msdb database. When you call a package from a SQL Server Agent job step and the package doesn't run, one of the following conditions is true:

The user account that runs the package as a job step differs from the original package author.

—or—

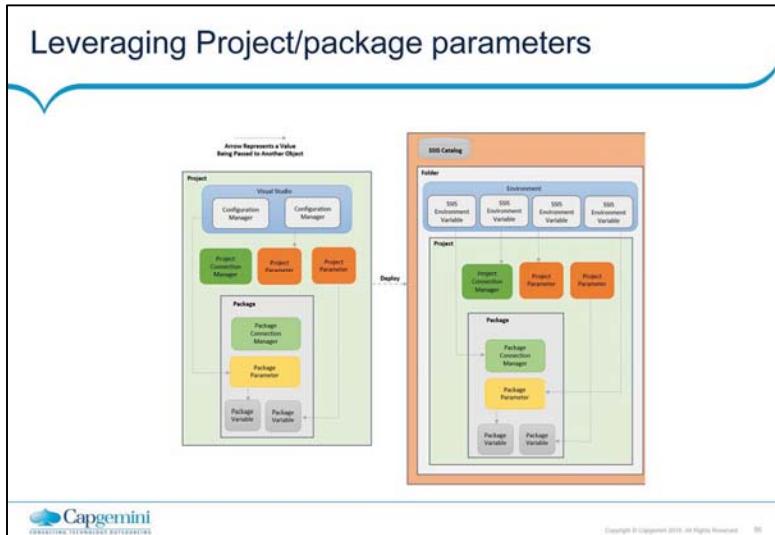
The user account does not have the required permissions to make connections or to access resources outside the package

When the user account that calls the package from the job step differs from the original package author, the package protection level may prevent the package from running.

This is because the user account either cannot decrypt the package or the package-sensitive data, or the user account cannot provide the sensitive data that is missing from the package.

Examples of sensitive data are the password part of a connection string, a variable that is marked sensitive, etc.

There are a couple of recommended methods for resolving issues with encryption and sensitive data.



SSIS Project & Package Parameters

Project Parameters

- Project parameters are new with SSIS 2012. A project parameter can be shared among all of the packages in an SSIS project. You want to use a project parameter when, at run-time, the value is the same for all packages.

Name	Data type	Value	Sensitive	Required	Description
FlatFileOrder_Schema1	String	V:\w\RSN\2\B1\FlatFile\schema1	False	False	
FlatFileOrder_Schema2	String	V:\w\RSN\2\B1\FlatFile\schema2	False	False	

Project Parameter

Package Parameters

Package parameters are also new with SSIS 2012. A package parameter is exactly the same as a project parameter – except that the scope of a package parameter is the individual package it resides in. You want to use a package parameter when, at run-time, the value is different for each package.

Name	Data type	Value	Sensitive	Required	Description
SummaryOrDetailValueExpected	String	Summary	False	False	

Package Parameter



Copyright © Capgemini 2012. All Rights Reserved.



Use of package and project parameters

- Parameters are useful for providing runtime values to properties during a package execution. Hence, in a way they replace the concept of configurations that we had for SSIS packages.
- Package parameters are useful for providing values to specific package executions. As they are package scope, they would only be available to the package in which they were created.
- Project parameters are available to all the packages in a project. They are useful for configuring values which can be shared between packages. For instance if you wanted to have a single parameter containing a server name to be used by multiple packages, then Project Parameters will be useful for you.
- You can use a parameter anywhere in an expression by referring it using the following syntax:

- The evaluation order is similar to using a variable in an expression and then executing a package. Hence, the value gets evaluated and assigned to the property at Validation phase.



Copyright © Capgemini 2010. All Rights Reserved.

Performance Tuning

Lesson 25:

Integration Services: Internals

- Two Engines:

- Runtime engine {highly parallel control flow engine that coordinates the execution of tasks or units of work within SSIS and manages the engine threads that carry out those tasks.}
- Data flow "pipeline" engine {manages the data pipeline}

- Inside the Data Flow

- Logical pipeline
- Physical memory buffers
- Execution Trees {is a group of transformations which start at either a source adapter or an asynchronous transform and end at the first asynchronous transform or a destination adapter. A data buffer has the scope of an execution tree.}
- Changes in SSIS 2008: Execution Trees are referred as "paths", that a path can be further divided into "subpaths."



Copyright © Capgemini 2010. All Rights Reserved. 102

Integration Services: Internals

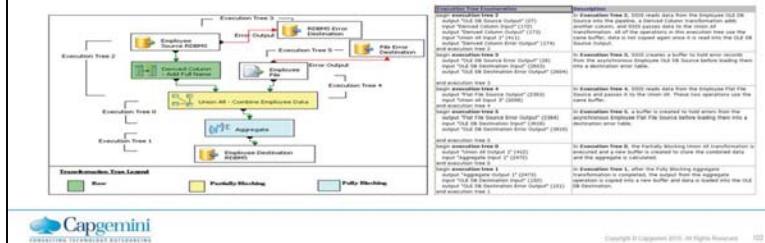
- Row Transformation (synchronous transformation):
 - Row-by-row basis
 - Do not block data flow in the pipeline
 - Data is not copied around, only pointers
 - Examples: Data Conversion, Derived Columns, Copy column, Multicast, Row count, Lookup etc.
- Partially Blocking Transformation (asynchronous transformation):
 - Introduces new buffers in memory layout
 - Transformed data is copied into new buffers
 - Examples: Merge, Merge Join, Union All etc.
- Blocking Transformation (asynchronous transformation):
 - Must see all data before passing on rows
 - Block the data flow – can be heavy on memory
 - May also use "private buffers" to assist with transforming data
 - Examples: Sort, Aggregate etc.



Copyright © Capgemini 2010. All Rights Reserved. 101

Integration Services: Internals (Execution Trees)

- Execution trees help us understand the buffer usage and check other alternatives for execution in terms of improving performance. (See how SSIS creates buffer for Source was observed and the same buffer being used by Derived Column transformation as it is a row/synchronous transformation. Separate sets of buffers are being created for Union All and Aggregate transformations as they are Partially Blocking and Blocking transformations respectively and need additional buffers to store their outcome.)



Capgemini
INTEGRATION SERVICES BEST PRACTICES

Copyright © Capgemini 2010. All Rights Reserved. 10

Tuning Tips

- Remove redundant columns
 - Use SELECT statements as opposed to tables
 - SELECT * is your enemy
 - Also remove redundant columns after every async component!
- Filter rows
 - WHERE clause is your friend
 - Conditional Split in SSIS
 - Concatenate or re-route unneeded columns
- Parallel loading
 - Source system split source data into multiple chunks
 - Flat Files – multiple files
 - Relational – via key fields and indexes- Multiple Destination components all loading same table
- Which component?
 - Bulk Import Task vs. Data Flow
 - Bulk Import might give better performance if there are no transformations or filtering required, and the destination is SQL Server.
 - Lookup vs. MergeJoin (LeftJoin) vs. set based statements in SQL
 - MergeJoin might be required if you're not able to populate the lookup cache.
 - Set based SQL statements might provide a way to persist lookup cache misses and apply a set based operation for higher performance.
 - Script vs. custom component: Script might be good enough for small transforms that're typically not reused



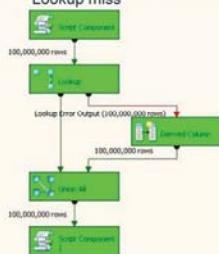
SQL Server 2005 SSIS Tuning the Dataflow Task.htm



Copyright © Capgemini 2010. All Rights Reserved. 103

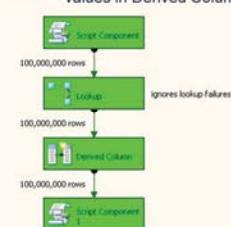
Performance Tuning: Case Study

Use Error Output for handling
Lookup miss



105 seconds

Ignore lookup errors and check for null looked up
values in Derived Column



83 seconds



Copyright © Capgemini 2010. All Rights Reserved. 104

Best Practices

Lesson 26:

Best Practices : Learning from Previous Projects

- Use template packages whenever possible. This enables consistency in
 - Creating Variables
 - Connection Manager
 - Setting common package properties
 - Using same Config Files across packages
- Use OLE DB connections unless there is a compelling reason to do otherwise.
- Only Configure package variables.
- One target table per package.
- Write comments ("annotate") in the designer editor, describing the functionality of the task, transformation etc.
- For large volume data, avoid row based operations, instead use Update set operations.
- Minimize blocking: As far as possible avoid use of asynchronous transformation.
- Maintain separate development, UAT and production environment. Also use source control tool (like VSS) – to check-in package .xml or solution files.
- BufferTempStoragePath - The location for temporary storage of buffer data. In case of space issues on the primary drive [in most of cases the C:/], this property should be set to the alternate path.

Copyright © Capgemini 2010. All Rights Reserved. 100

Best Practices : Learning from Previous Projects

- If one package is being called from other package, the called package can execute either in its own memory or in the memory of the parent package. Setting the Execute Out Of Process property of the Execute Package Task to true. This forces SSIS to create a separate process for each child package to run in.
- Explicitly Type Casting in Derive column Transformation increases the overhead. Usage of Data conversion Transformation to do the necessary casting will resolve the problem.
- Ignoring Execution Status of the Dataflow task, set 'Force Execution Result' property of current task to 'Completion'.
- Limit the number of data flows in a single package. To avoid memory issues, while opening, saving and executing SSIS packages. Break the package logically into multiple packages.
- Observation: Package stored on file system executes faster than ones stored on database. Also debugging on file system is better.
- Maximize Parallelism: caution-Multiple parallel flows at time slows down the process depending on the available server resources-Some properties that impact parallel runs are
- Max Concurrent Executables (Package property)- define how many tasks can run concurrently.
- Engine Threads (Data Flow property) - how threads the engine would create
- Data Flow Performance Improvement, properties that impact - 'DefaultBufferSize' to 100000000. 'DefaultBufferMaxRows' to 500000 (depending on server resources).
- Loading tables with no constraints. Use - FAST Load option, this is similar to bulk insert operation, which generally can be used for staging table loads. The performance will be faster than the normal load

Copyright © Capgemini 2010. All Rights Reserved. 107

Hands on Training Lab

SQLServer Integration Services

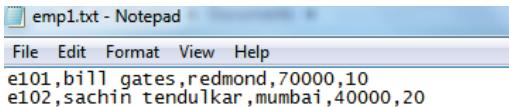
Table of Contents

LAB 1-DATA LOAD FROM FLAT FILE INTO RELATIONAL TABLE	3
LAB 2-DERIVED COLUMN TRANSFORMATION.....	8
LAB 3- AGGREGATE TRANSFORMATION.....	21
LAB 4- CONDITIONAL SPLIT TRANSFORMATION.....	28
LAB 5- LOOK UP TRANSFORMATION.....	34
LAB 6- COPY COLUMN TRANSFORMATION.....	48
LAB 7- MERGE TRANSFORMATION.....	54
LAB 8- MERGE JOIN TRANSFORMATION	60
LAB 9- UNION ALL TRANSFORMATION	63
LAB 10-SLOWLY CHANGING DIMENSION.....	65
LAB 11- EXTRACTING, TRANSFORMING AND LOADING DATA.....	73
LAB 12- EXECUTE SQL TASK & SCRIPT TASK	78
LAB 13-FILE SYSTEM TASK.....	86
LAB 14-FOR EACH LOOP CONTAINER WITH EXECUTE SQL TASK AND SCRIPT TASK..	93
LAB 15- EXECUTE PROCESS TASK.....	100
LAB 16- USING ERROR REDIRECT ROW.....	107
LAB 17- TRANSACTION SUPPORT AND CHECKPOINT.....	110
LAB 18- USING DATA VIEWER.....	117
LAB 19- USING PACKAGE LOGGING.....	119
LAB20-PROJECT DEPLOYMENT	121
LAB21- PACKAGE EXECUTION USING SQL SERVER AGENT.....	130

Lab 1-Data load from Flat File into Relational Table

Objective	How to extract data from flat file and load into a relational table.
Lab Setup	<ul style="list-style-type: none">• SSDT tool• Login details for connecting to the database

1. Create a separate folder "Project" for placing input data feed file
2. In that folder create a txt file emp1.txt and type this data

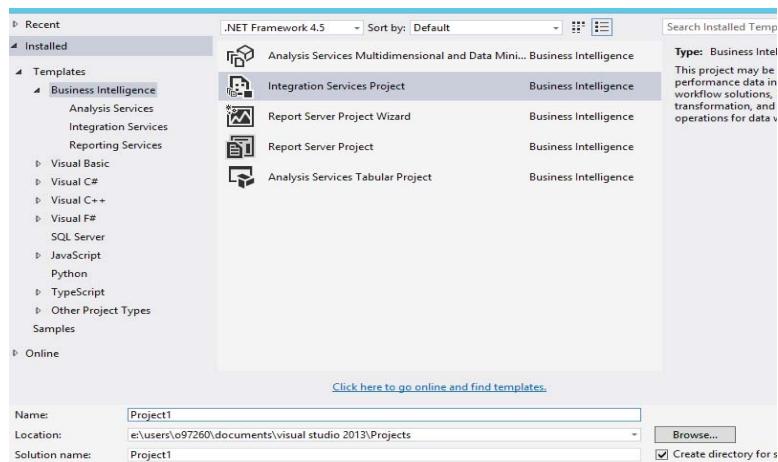


emp1.txt - Notepad
File Edit Format View Help
e101,bill gates,redmond,70000,10
e102,sachin tendulkar,mumbai,40000,20

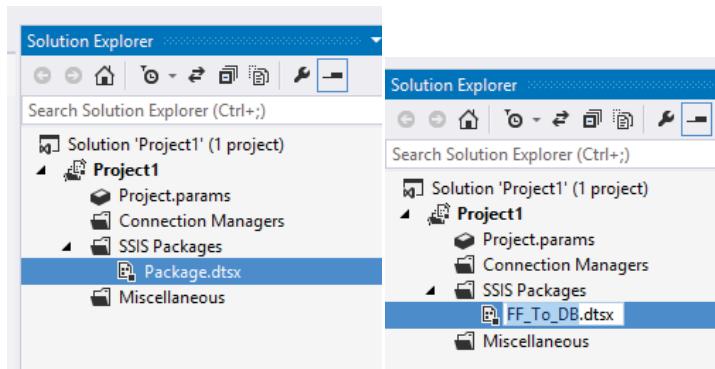
3. Connect to sql server using SSMS and create the following table

```
createtable emp1
(
    empno varchar(10)primarykey,
    empname varchar(50),
    city varchar(50),
    sal money,
    deptno int
)
```

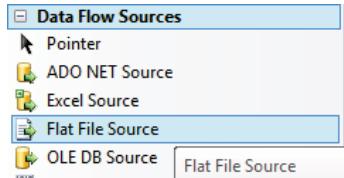
4. Start a new SSIS project. Start BIDS and click file => new project
 - o Select integration services project among Business Intelligence Projects.
 - o Rename it to "Project1".



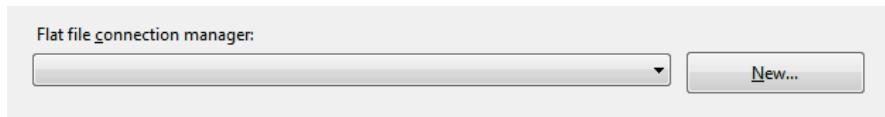
5. From Solution Explorer rename package.dtsx to FF_To_DB.dtsx



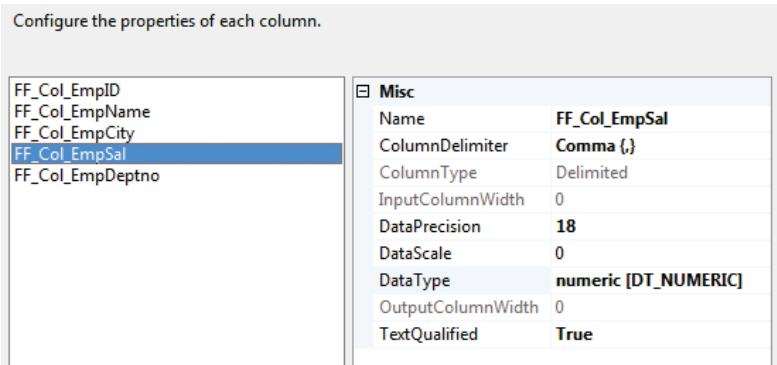
6. In the package designer, goto data flow tab and click the link in the middle to create a new data flow task. (this data flow task is automatically called from control flow).
7. In the data flow task designer drag "Flat File Source task" from toolbox
 - o Rename it "Emp Flat File Data"



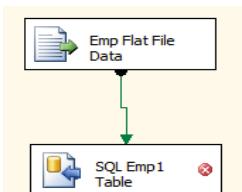
- Double click the above dragged task to popup a dialog box. Click "New" Button to create a FF connection Manager.



- Give Connection Manager Name as: FF_Emp_CM
- For filename, browse and locate emp1.txt.
- Click Columns TAB to ensure columns identified properly
- Click Advanced TAB and rename cols as below:
- For Sal field set data type "currency [DT_CY]"
- For Deptno field set data type "four-byte signed integer [DT_I4]"



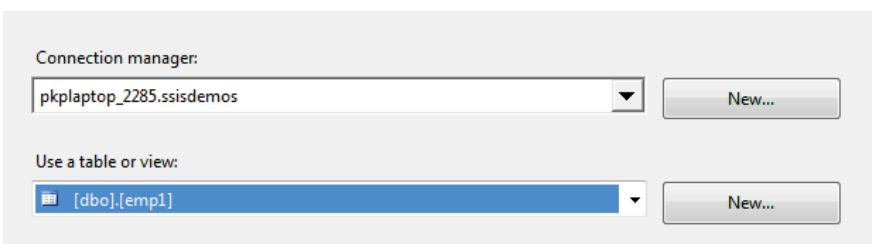
- Click OK
- In Toolbox , from Destinations section drag sql server destination
 - Rename it as SQL Emp1 Table
- Connect the green connector of Flat File Source to SQL Emp1 Table destination.



18. Double Click SQL Emp1 Table destination task.
- o Click New to create New Connection Manager, again click new
 - o Give the connection details as appropriate



19. Test Connection, Click Ok twice.
20. Select the emp1 table created earlier in the dialog box as below:



21. Go to Mappings tab and map columns as below:

Input Column	Destination Column
FF_Col_EmpID	empno
FF_Col_EmpName	empname
FF_Col_EmpCity	city
FF_Col_EmpSal	sal
FF_Col_EmpDeptno	deptno

22. Click Ok to come to data flow designer surface
23. Click Run Button  on toolbar, the package should run successful. Click Stop Button explicitly on toolbar.
24. Goto SSMS and check the sql server table data using [select*from emp1](#)

Lab 2-Derived Column transformation

Objective	To load the data from excel sheet to OLE DB database.
Lab Setup	<ul style="list-style-type: none">• SSDT tool• Existing SSIS Project and connection to database

Source:

Microsoft Excel - Employee details					
	A	B	C	D	E
1	Emp code	Emp Name	Address	Age	Salary
2	101	a	xyz	23	20000
3	102	b	abc	43	56000
4	103	c	pqr	34	40000
5	104	d	asd	56	59000
6	105	e	abc	33	39000
7	106	f	def	27	29000
8	107	g	ghi	47	49000
9	108	h	jkl	55	70000
10	109	i	mno	28	30000
11	110	j	lop	38	36000
12	111	k	stu	49	50000
13	112	l	vwx	29	31000
14	113	m	yz	30	34698
15	114	n	qwe	40	49875
16	115	o	rtyu	50	58000
17	116	p	fnfd	39	39785
18	117	q	sfh	25	26000
19	118	r	jkg	26	27154
20	119	s	rnb	36	36987
21	120	t	afgh	47	46987
22					
23					
24					
25					
26					
27					
28					
29					
30					

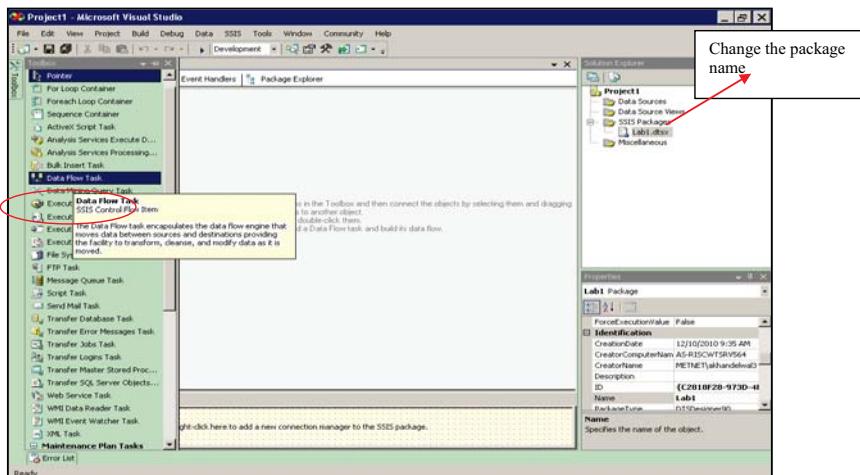


Employee_data.csv

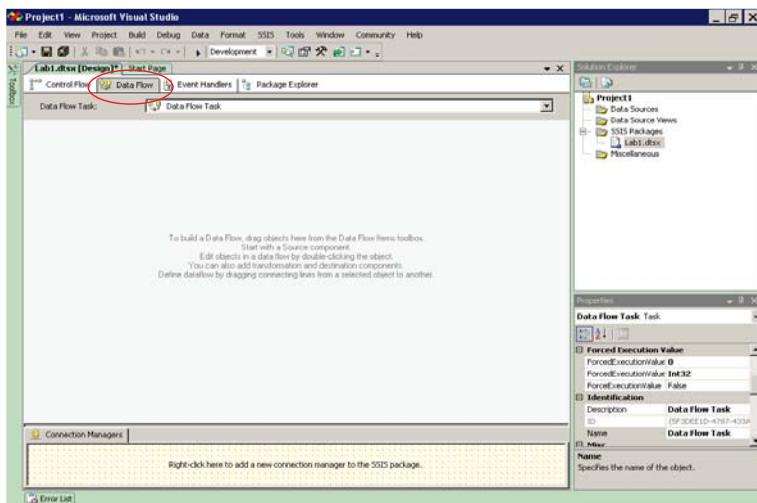
Source File attached here:

1)Create a new package named “Lab1” under the already created project.

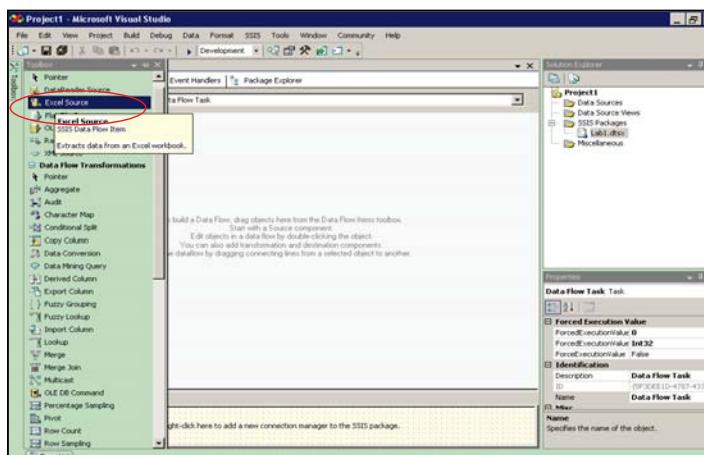
2)Click on the toolbox and drag and drop the Data flow task on control flow task:



3) Double click the data flow task or click on the data flow task and select the data flow from the tab to begin the mapping:

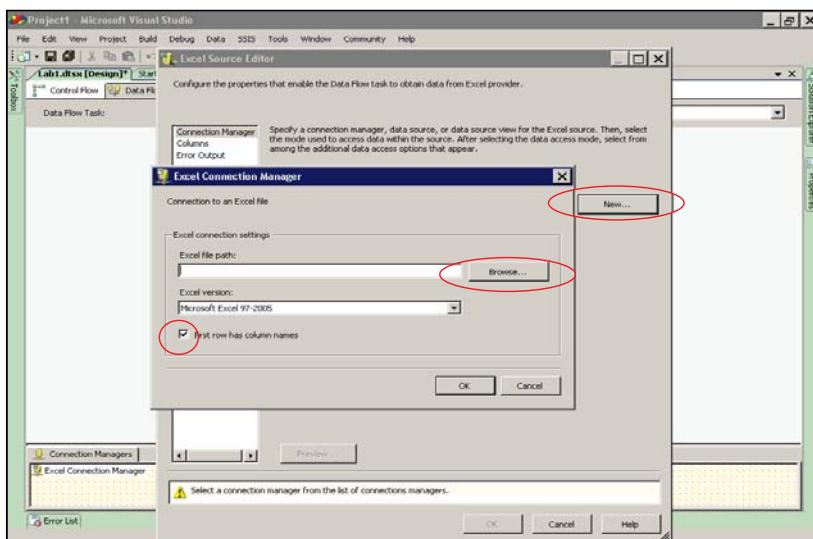


- 4) Select a source from the toolbox. Here we are taking the excel source:



- 5) Double click the source to open the editor:

5.1) Create a new connection manager:

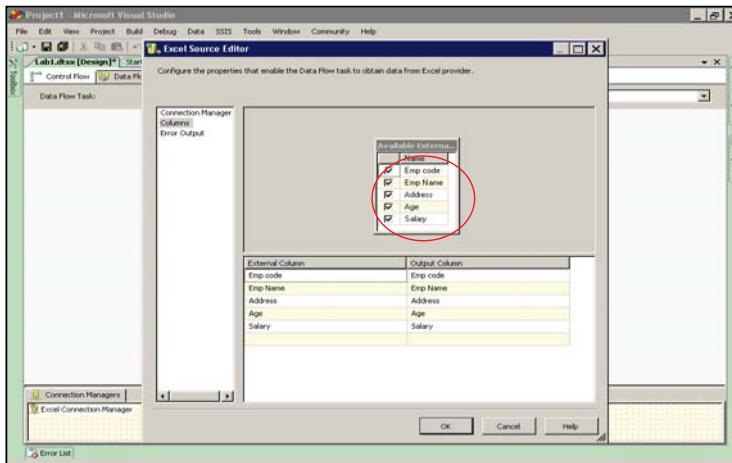


If the excel file contains columns then you must check the box for "First row has column names" check box.

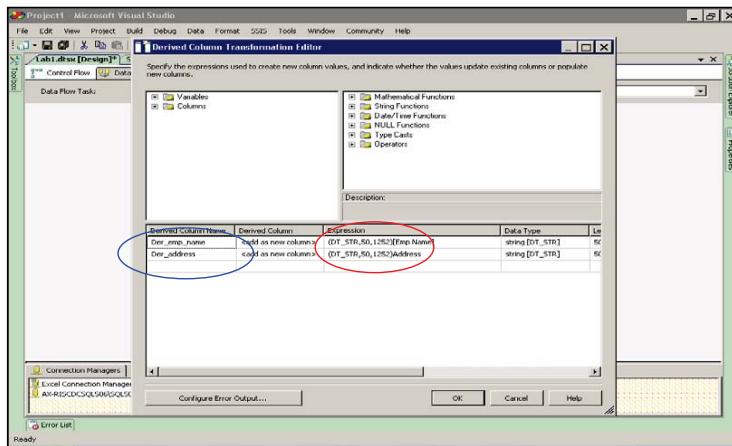
Click 'OK' after selecting the file and select Data access mode as – Table or view and name of the excel sheet as 'sheet1' if the data is in that sheet.

You can also click the "preview" button to view the data in the source table.

5.2) Click on columns to select the required columns in the target table:

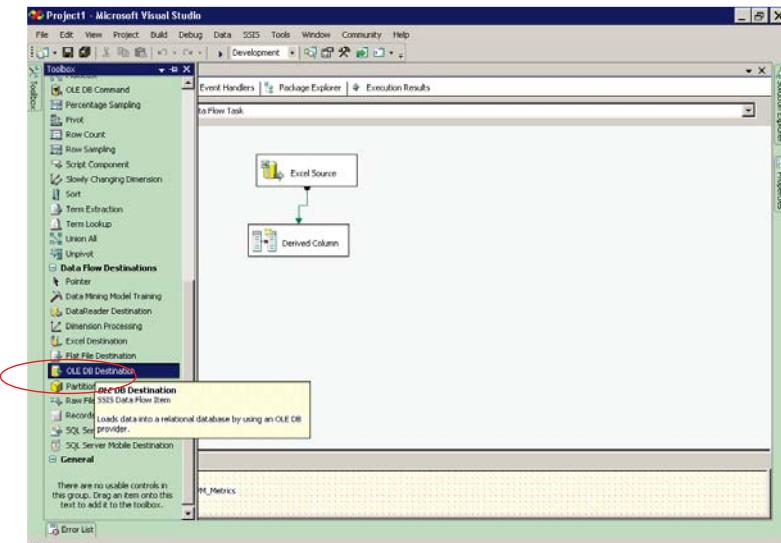


6) Select Derived column transformation from the toolbox and double click it to get the editor:



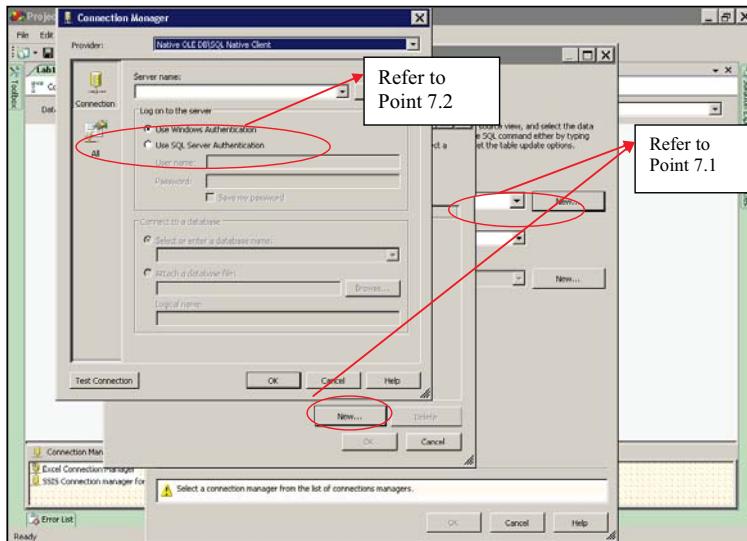
Change the data type of Emp_name and address (Unicode in source) to string and click OK.

7) Create a target table in the database (SQL Server) if not present, to transform the data:
Select OLE BD Destination as target table:



Connect Derived column transformation and target using the GREEN connector:

- 7.1) Double click OLE DB destination to open the editor. Create a New Connection Manager:

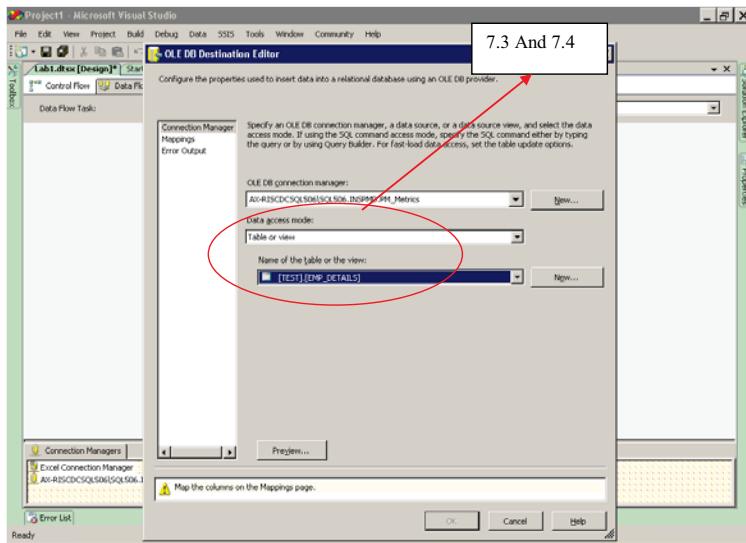


7.2) Select the server name, use SQL Server Authentication (use valid user name and password) and select a database name and test the connection.

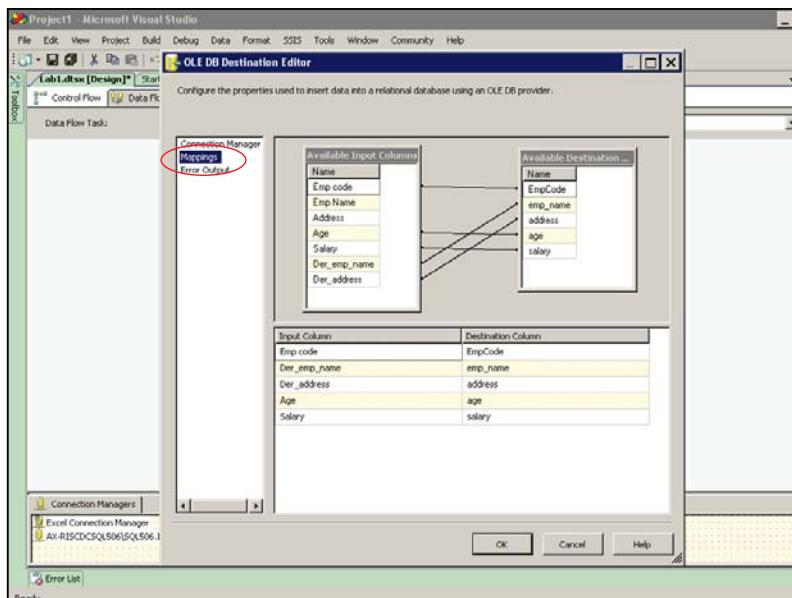
Click OK till you are able to view the editor with a connection manager.

7.3) Select Data Access Mode as- Table or view.

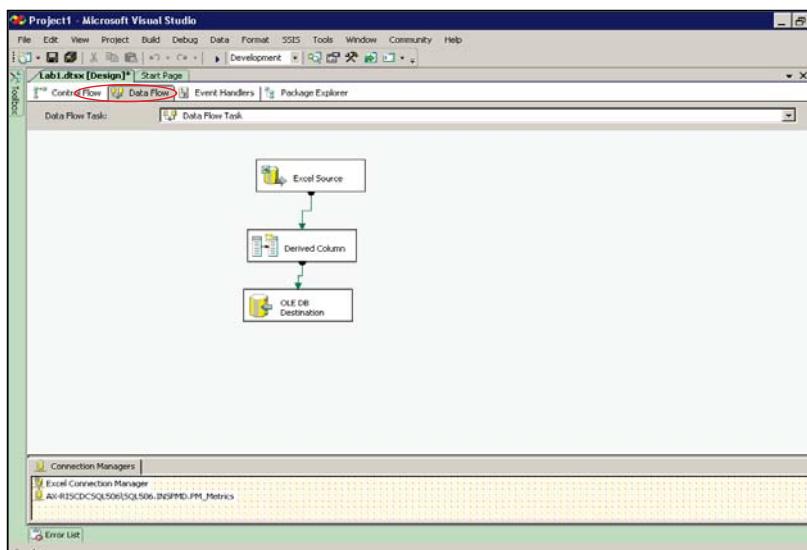
7.4) And then select the name of the target table.



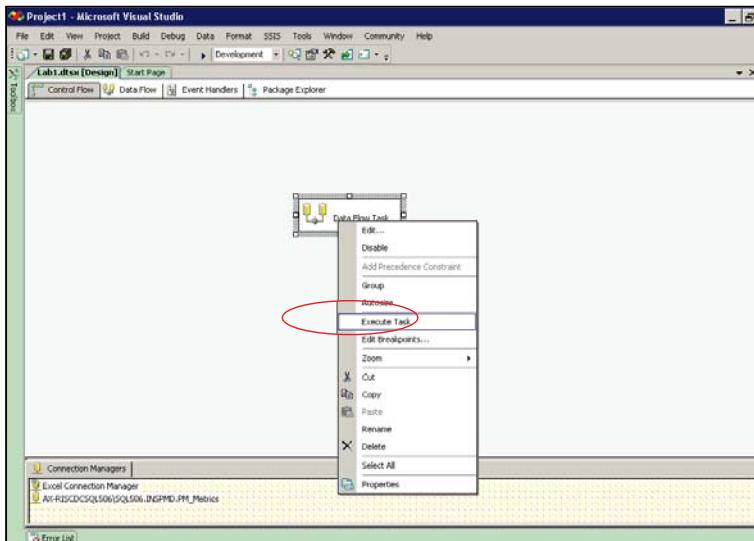
7.5) Click on Mappings to map the data from source to target.



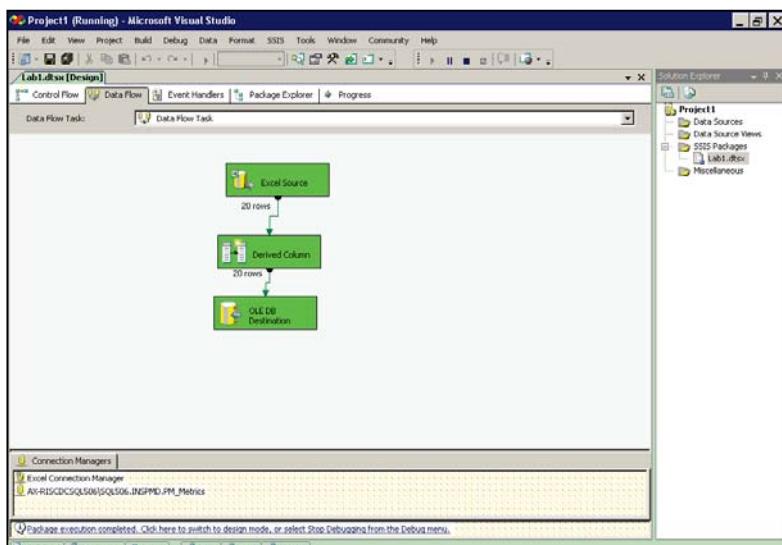
The mapping from source to target would be like:



- 8) Save the project.
Switch to the control flow task and right click on data flow task. Click on Execute task.



9) After executing the data flow and on successful completion the Data Flow will be as follows:



You can check the data in the target table using SQL Server.

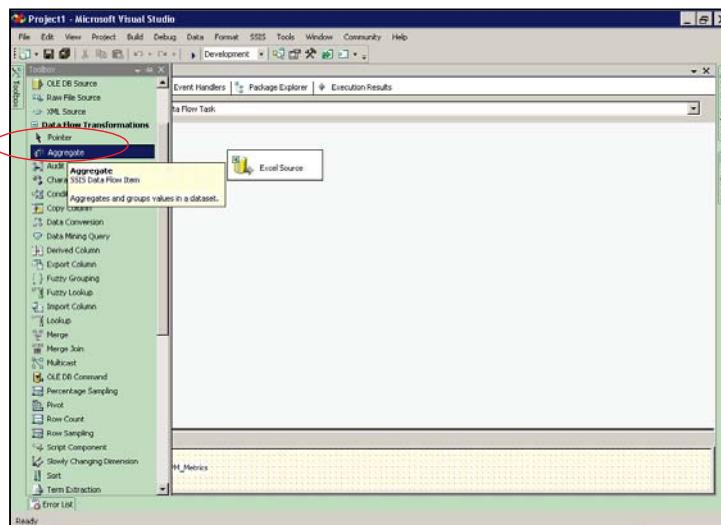
Emp_code	emp_name	Address	Age	salary
101	a	xyz	23	20000
102	b	abc	43	56000
103	c	pqr	34	40000
104	d	asd	56	59000
105	e	abc	33	39000
106	f	def	27	29000
107	g	ghi	47	49000
108	h	jkl	55	70000
109	i	mno	28	30000
110	j	lop	38	36000

111	k	stu	49	50000
112	l	vwx	29	31000
113	m	yz	30	34698
114	n	qwe	40	49875
115	o	rtyu	50	58000
116	p	fnfd	39	39785
117	q	sfh	25	26000
118	r	jkg	26	27154
119	s	mnb	36	36987
120	t	afgh	47	46987

Lab 3- Aggregate Transformation

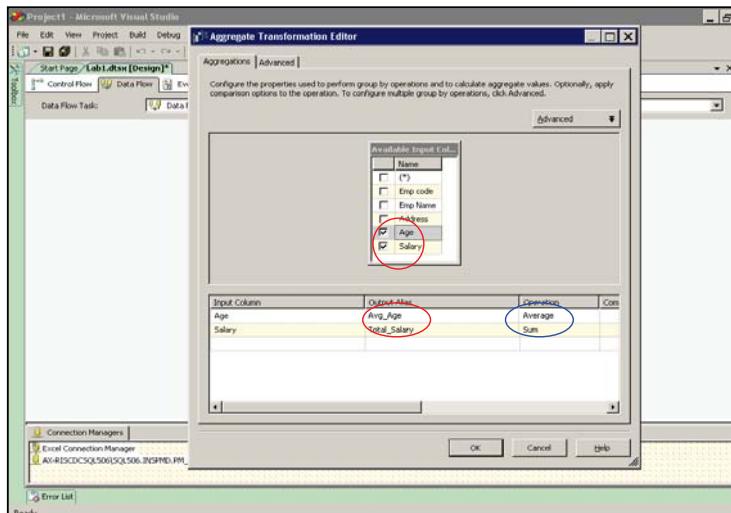
Objective	To find the average age in the organization and the total salary paid
Lab Setup	<ul style="list-style-type: none"> SSDT tool Existing SSIS Project and Package

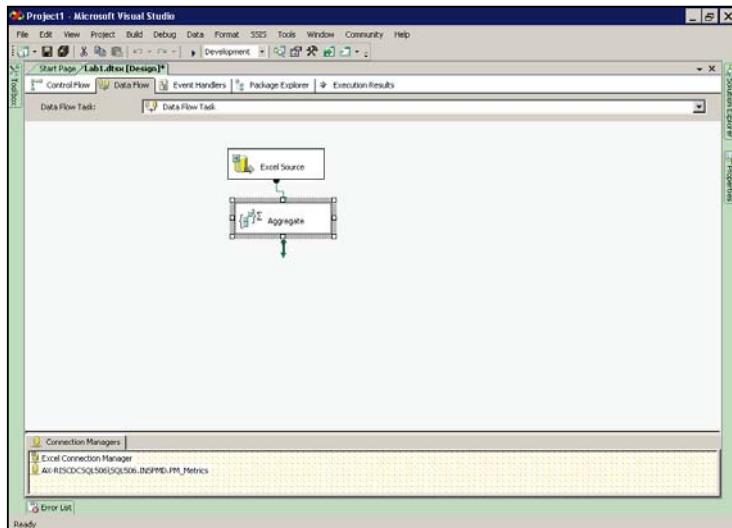
1) In the Data flow Task after selecting the source table, select aggregate transformation for the tool box.



2) Double click the transformation to open the editor.

- ➔ Select age and salary.
- ➔ Select the 'Operation' on AGE as average and on salary as SUM and then click on OK.

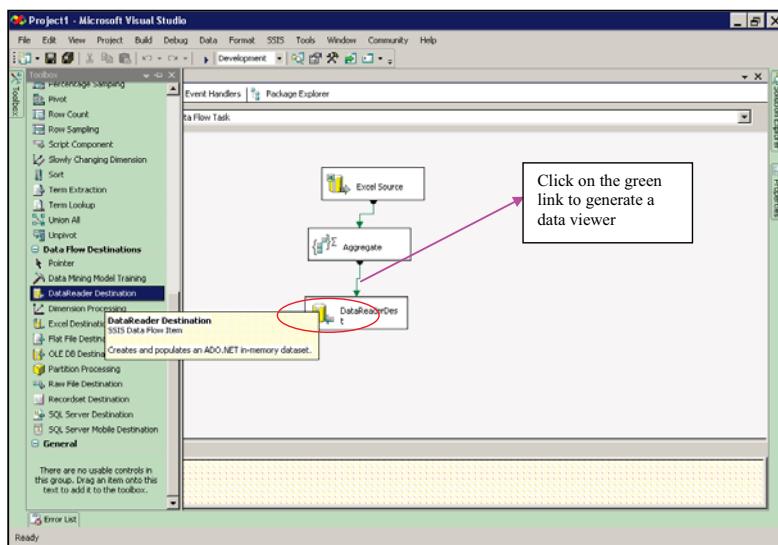




Notes:

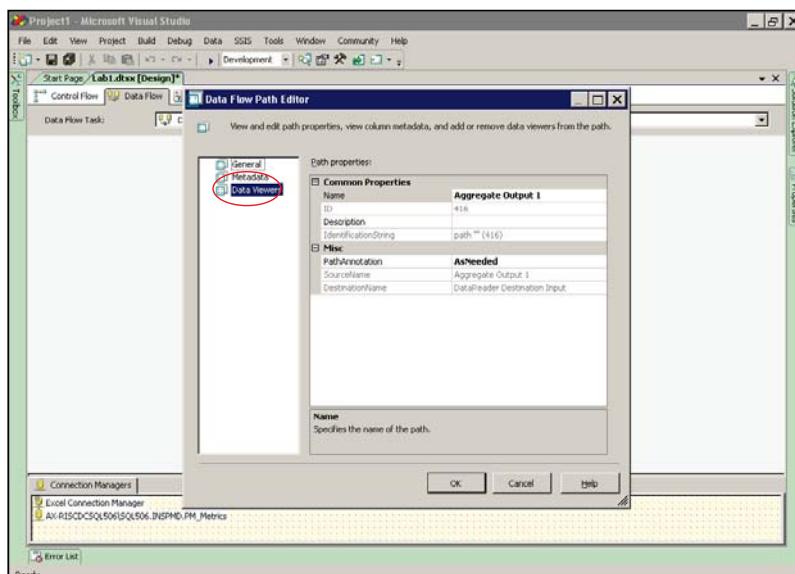
You can use DATA READER Destination as dummy target table. This table is not save anywhere. It is only used to check the final data when we do not want to make any change or load the final data into the target table.

This destination can be used without any configuration changes, in effect creating a 'null' destination. The data goes nowhere but we can trouble shoot any issue in our derived column upstream before

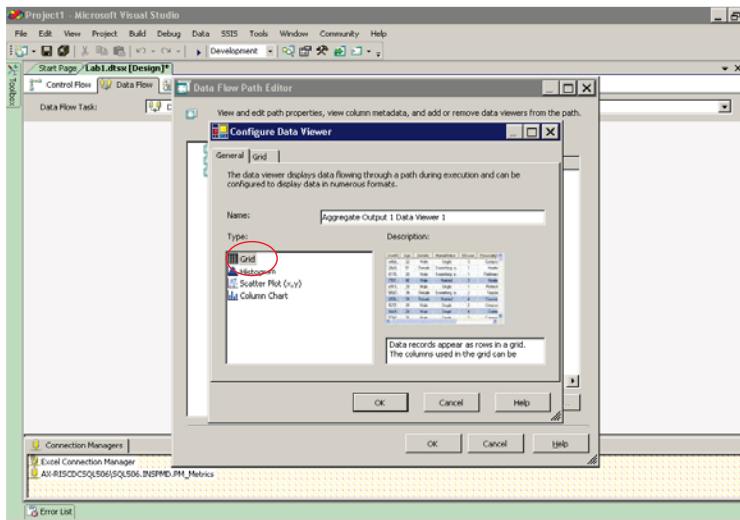


To view the data we have to use DATA VIEWER.

- 3) Select DATA READER Destination from the toolbox.
 - Link the aggregate transformation to the DATA READER Destination.
 - Double click the link to view the data flow editor and select Data Viewers.



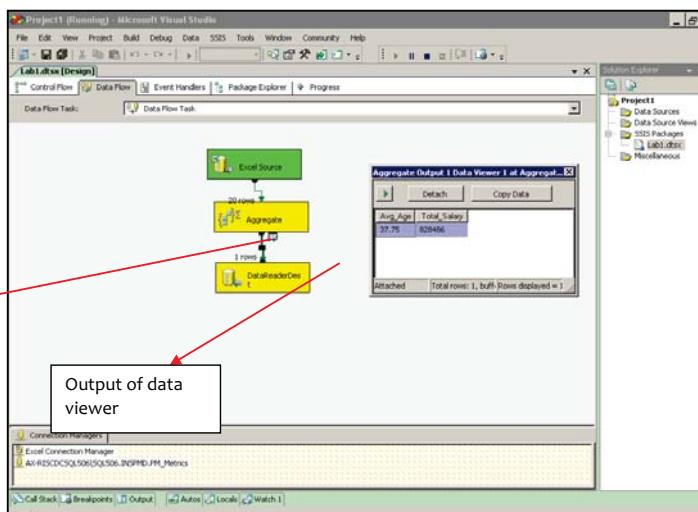
→ Add a new data viewer and select Grid and click OK twice.



4) A small icon of GRID will be visible near the joining arrow.

5) Save and execute the DATA FLOW TASK.

6) On successful execution a grid with output will be shown as:

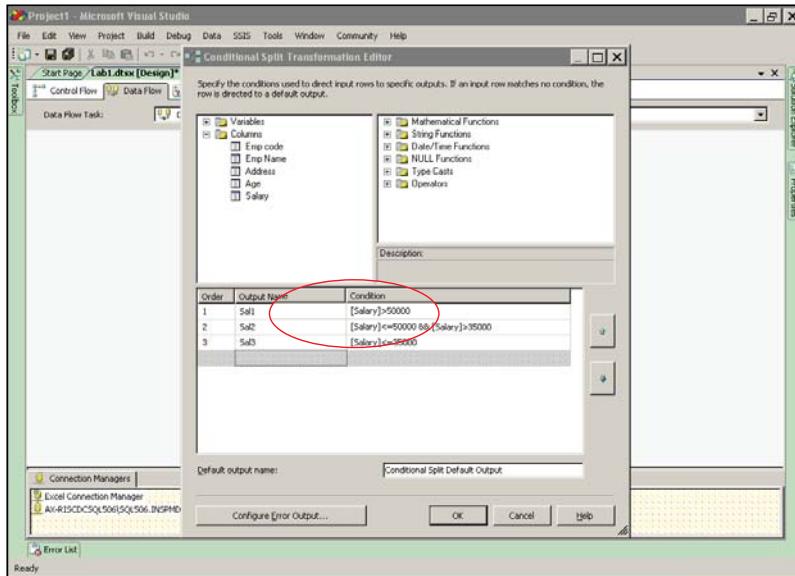


7) As soon as the data viewer is closed the execution gets completed.

Lab 4- Conditional split transformation

Objective	To load different tables with specific salary range (>50000, >35000 and <50000, <=35000) using conditional split and derived column transformation
Lab Setup	<ul style="list-style-type: none"> SSDT tool Existing SSIS project and Package

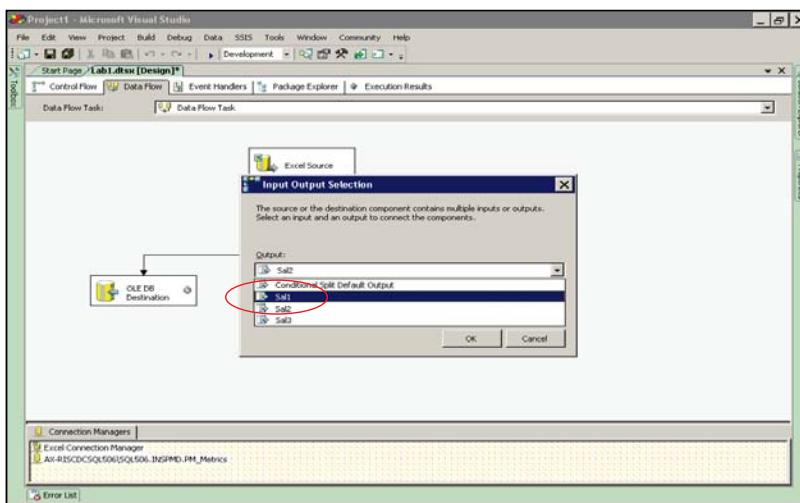
1)In the Data flow Task after selecting the source table, select conditional split transformation from the tool box and double click to get the editor.



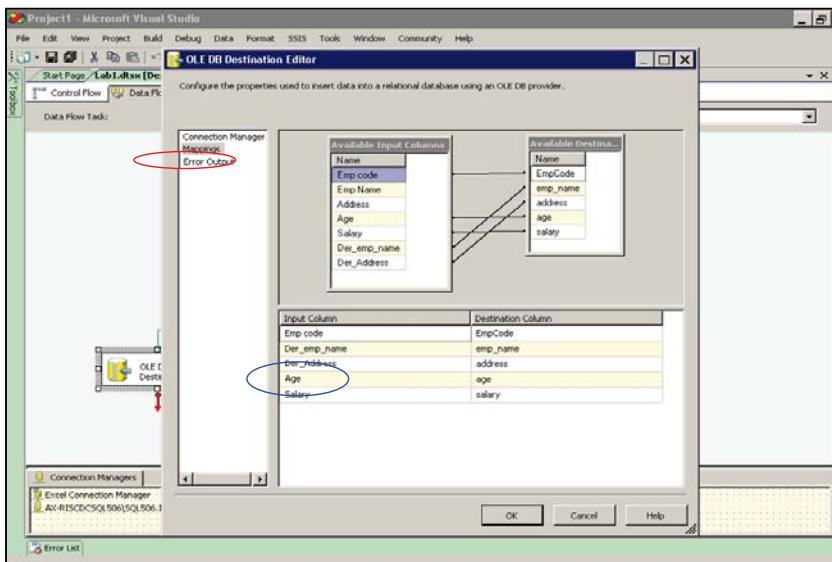
2)In the editor use different conditions according to which you want to divide the data.

3)Insert a Derived column transformation between source and conditional split and change the data type to name and address.

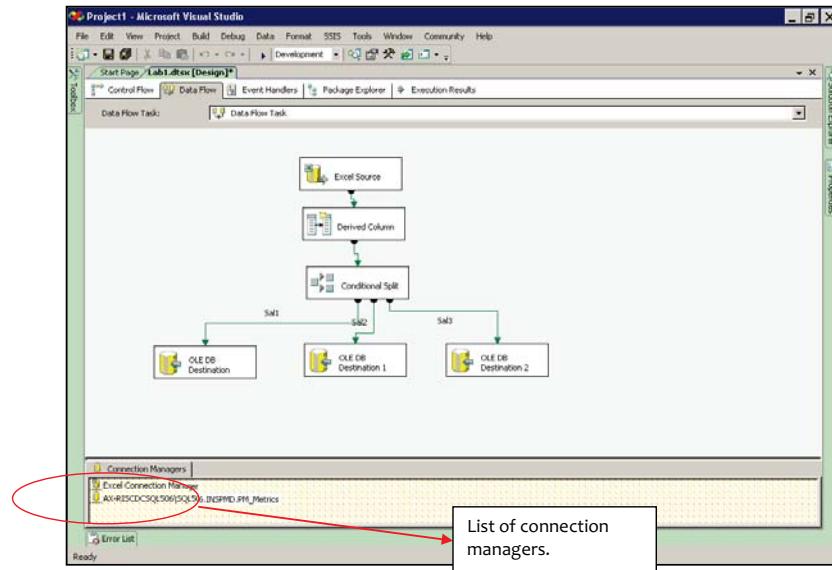
- Select target tables (OLE DB) and join with the transformation (select valid output).



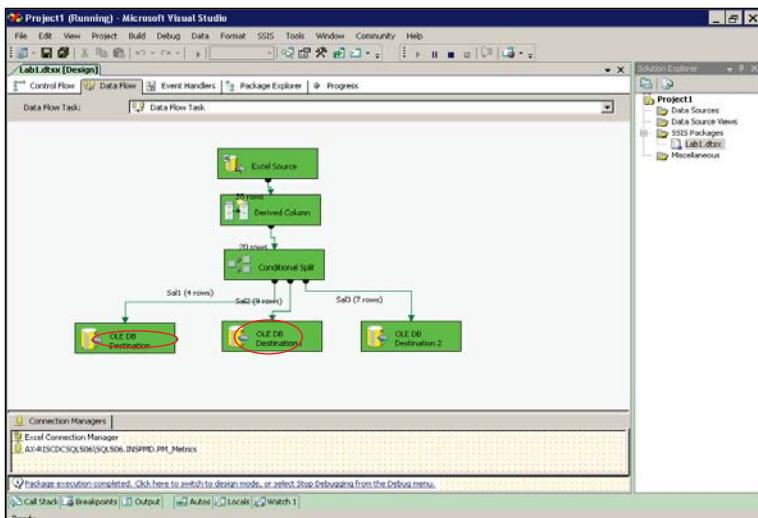
- You can use the existing connection manager or can create a new connection manager for the target table.
- Use DATA ACCES MODE as –Table or view.
- Select appropriate mapping.



4) Repeat for all the target tables.



- 5) Save and execute.



6)Using "SELECT" statement in SQL server.

Destination 1 :

Emp_code	emp_name	Address	Age	salary
102	b	abc	43	56000
104	d	asd	56	59000
108	h	jkl	55	70000
115	o	rtyu	50	58000

Destination 2 :

Emp_code	emp_name	Address	Age	salary
103	c	pqr	34	40000
105	e	abc	33	39000
107	g	ghi	47	49000
110	j	lop	38	36000
111	k	stu	49	50000
114	n	qwe	40	49875
116	p	fnfd	39	39785
119	s	mnb	36	36987
120	t	afgh	47	46987

Destination 3

Emp_code	emp_name	Address	Age	salary
101	a	xyz	23	20000
106	f	def	27	29000
109	i	mno	28	30000
112	l	vwx	29	31000
113	m	yz	30	34698
117	q	sfh	25	26000
118	r	jkg	26	27154

Lab 5- Look Up Transformation

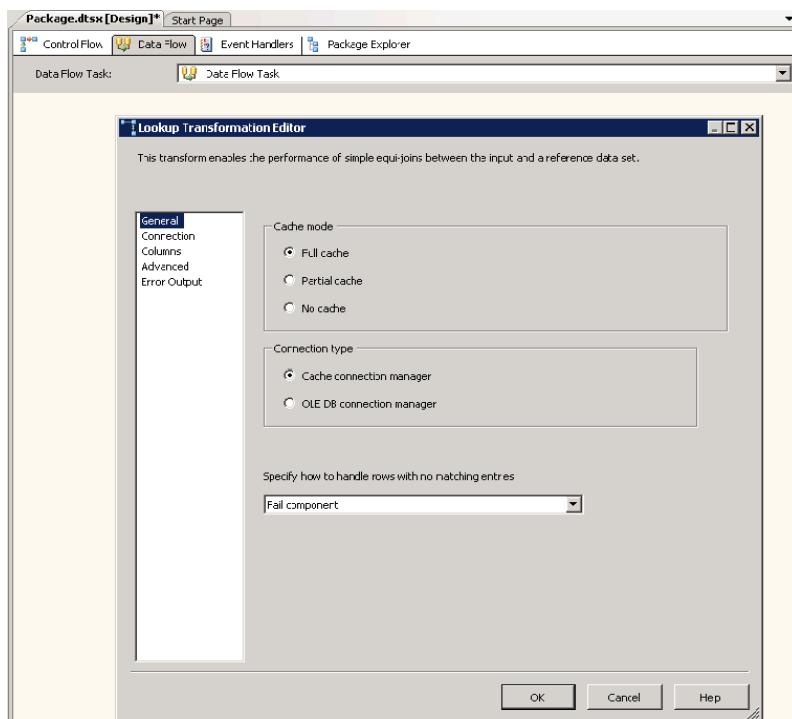
Objective	To check in the database whether there is any change in the address of the employee or if any new employee details are added then load the data in the database.
Lab Setup	<ul style="list-style-type: none">• SSDT tool• Existing SSIS project and Package

1)In the Data flow Task after selecting the source table, drag and drop Data conversion transformation and double click to view the editor.

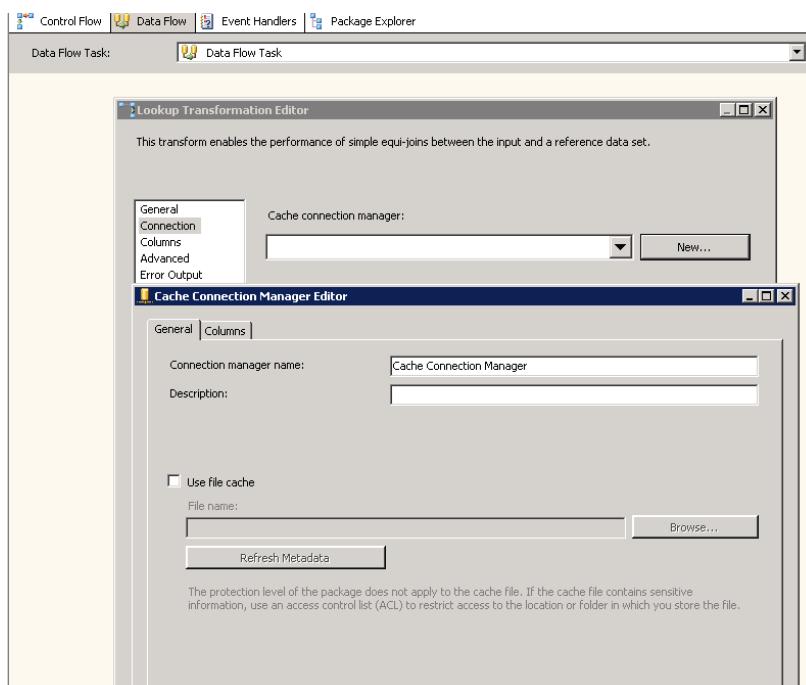
- Convert the data type of the fields like emp_name, emp_code, address and salary compatible with the target table.

2)Drag and drop the Look up transformation and then double click to view the editor.

3) In the General tab, you can select the Cache Mode and the Connection Type:

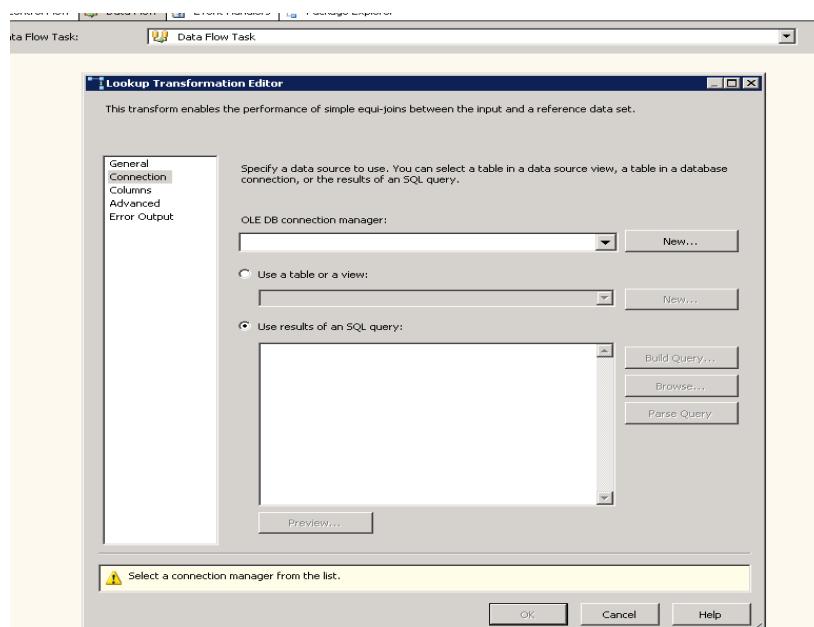


4) After selecting Cache Connection Manager in the Connection Type, You can select the Cache Connection Manager which you have created in the Cache Transformation created before the Lookup Transformation.

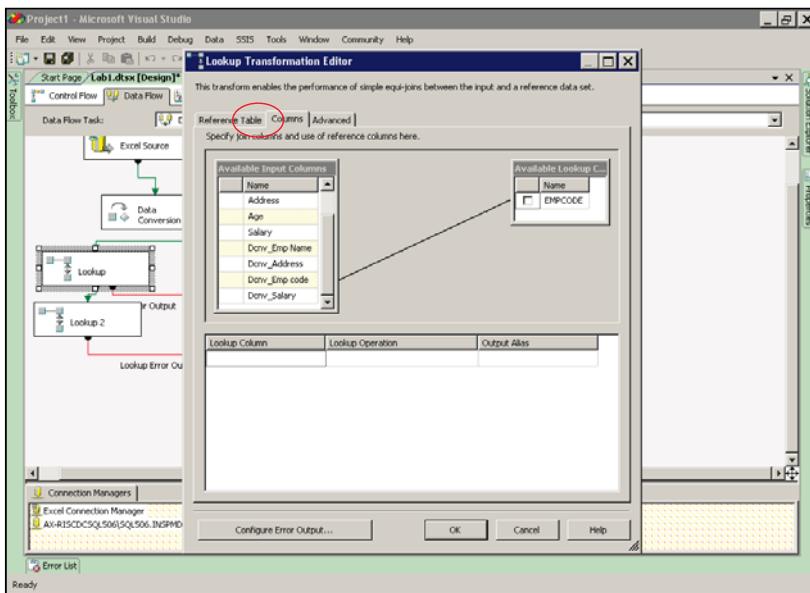


- 5) After selecting OLE DB Connection Manager in the Connection Type, In the editor you can either use a table or use SQL statement to look up the data.
- 6) Select the connection manager and use a SQL statement to extract empcode.

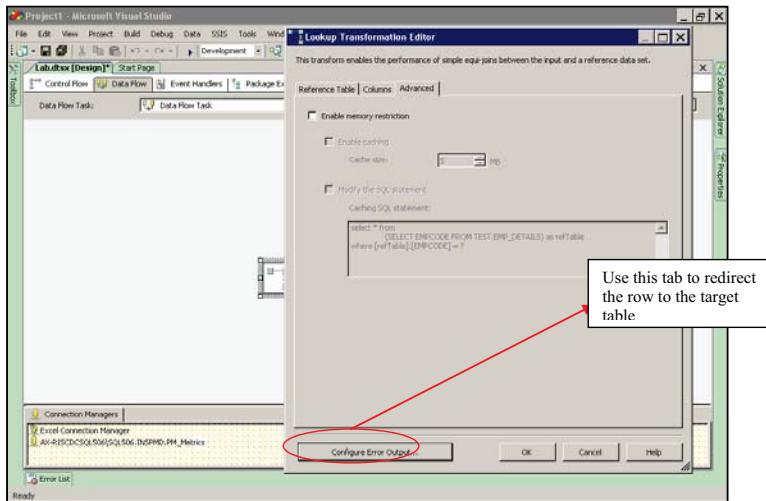
(Select empcode from table name)

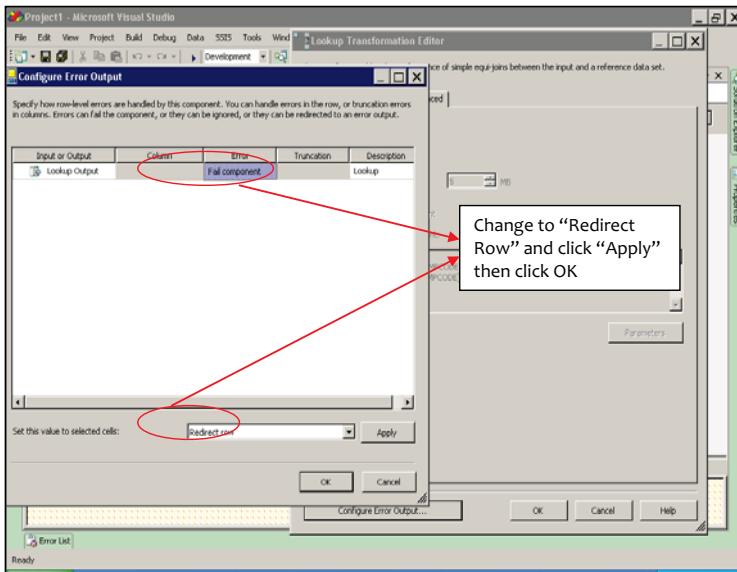


- 7) On the column tab join the empcode to find a match for the same.

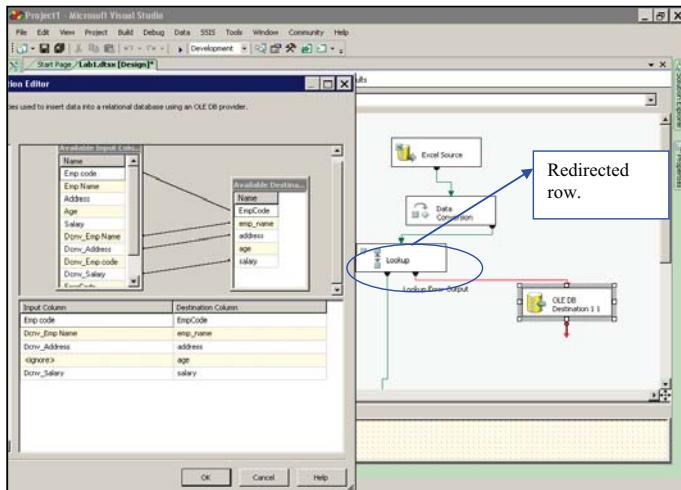


- 8) If the lookup finds any new data then it redirects to the target table.





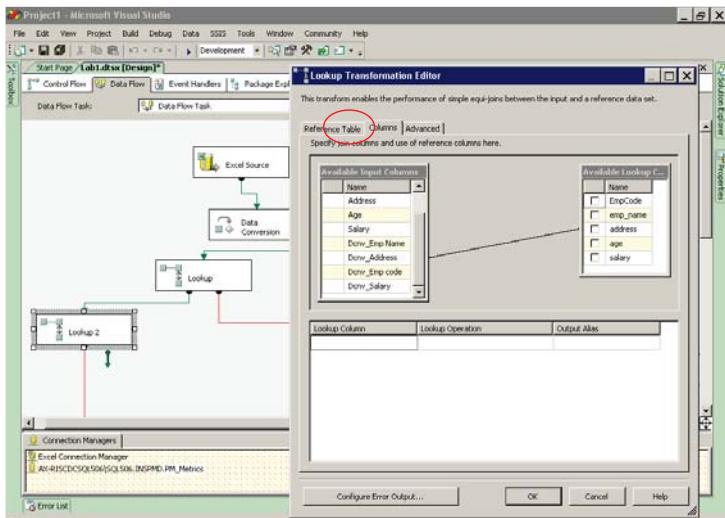
9) Use one OLE DB destination to redirect the new row to target table. In the OLE DB destination set the connection properties and map the columns.



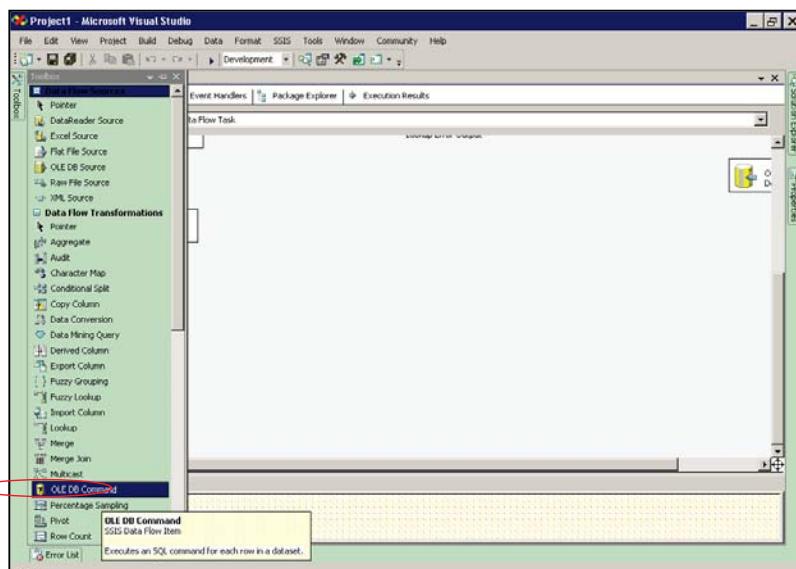
10) Use another look up to check if there is any change in address of the data that is already present in the target table.

→ To use the result of an SQL query use the following query:
`SELECT * FROM TEST.EMP_DETAILS`

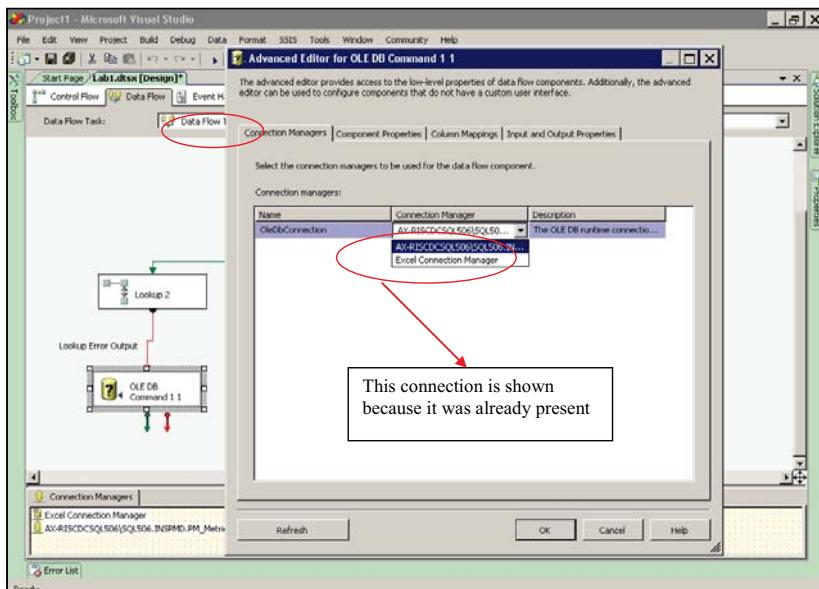
11) In the Columns tab match the address columns to check.



- 12) Now select OLE DB command from the toolbox.



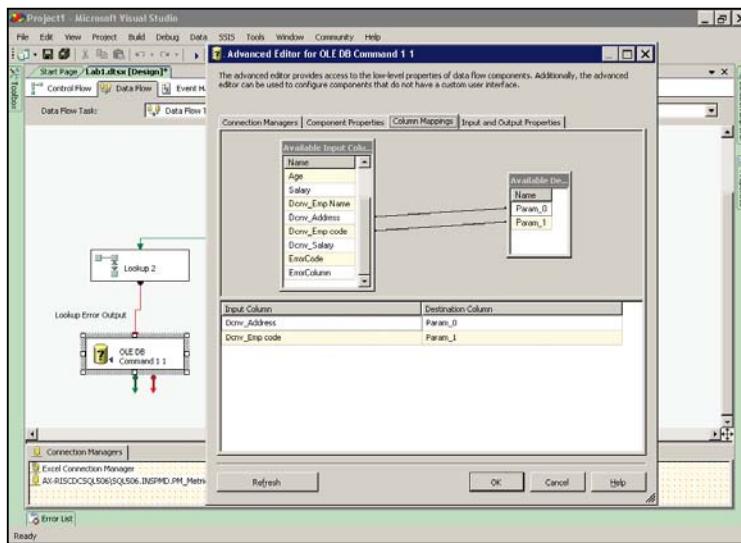
13) Set the connection as available:



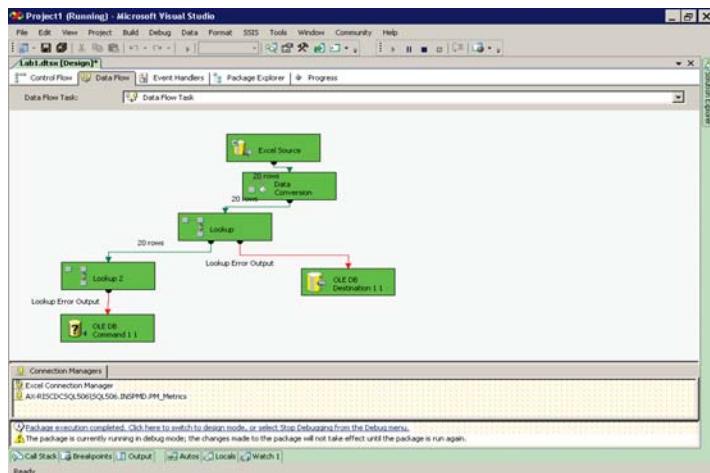
- In the component properties tab select the SqlCommand and write the query as:
**UPDATE TEST.EMP_DETAILS
SET ADDRESS=?
WHERE EMPCODE=?**

[The two question marks create parameters which are used to check the data (Old and updated). The first question mark refer to first and second refer to the second paramet]

14) In the column Mappings tab map the columns with the parameters:



15)Now save and execute the Data flow.



Original data:

Emp_code	emp_name	Address	Age	salary
101	a	xyz	23	20000
102	b	abc	43	56000
103	c	pqr	34	40000
104	d	asd	56	59000
105	e	abc	33	39000
106	f	def	27	36254
107	g	ghi	47	49000
108	h	jkl	55	70000
109	i	mno	28	30000
110	j	lop	38	36000
111	k	stu	49	50000
112	l	vwx	29	31000
113	m	yz	30	32000
114	n	qwe	40	49875
115	o	rtyu	50	58000
116	p	fnfd	39	39785
117	q	sfh	25	26000
118	r	jkg	26	27154
119	s	mnb	36	35647
120	t	afgh	47	46987

Final data:

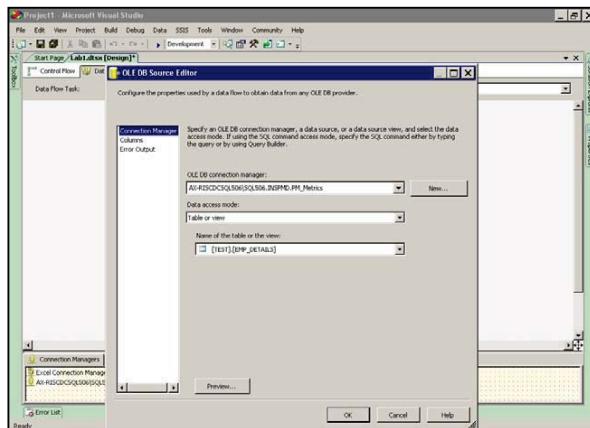
Emp_code	emp_name	Address	Age	salary
101	a	klasfnx	23	20000
102	b	abc	43	56000
103	c	pqr	34	40000
104	d	asd	56	59000
105	e	abc	33	39000
106	f	perghjsd	27	36254
107	g	ghi	47	49000
108	h	jkl	55	70000
109	i	mno	28	30000
110	j	lop	38	36000
111	k	jshfkag	49	50000
112	l	vwx	29	31000
113	m	yz	30	32000

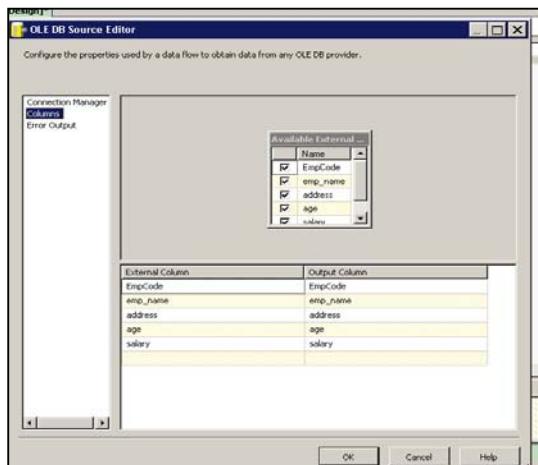
114	n	qwe	40	49875
115	o	rtyu	50	58000
116	p	fndf	39	39785
117	q	sdfjfgosf	25	26000
118	r	jkg	26	27154
119	s	mnb	36	35647
120	t	afgh	47	46987

Lab 6- Copy Column Transformation

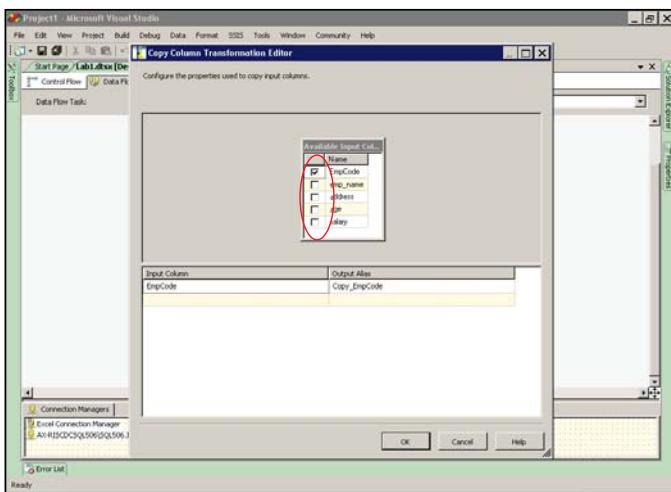
Objective	To create a copy of a column.
Lab Setup	<ul style="list-style-type: none"> • SSDT tool • Existing SSIS project and Package

1)Select OLE DB source and double click to view the editor, set the connection manager and select the required columns.



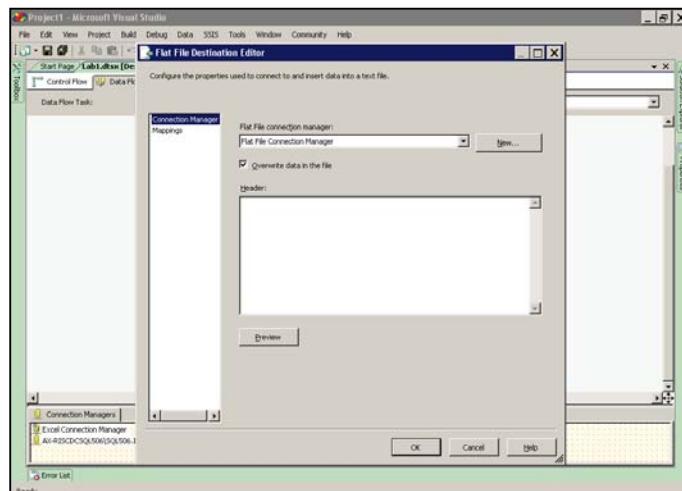


2) Select copy column transformation and double click to view the editor.

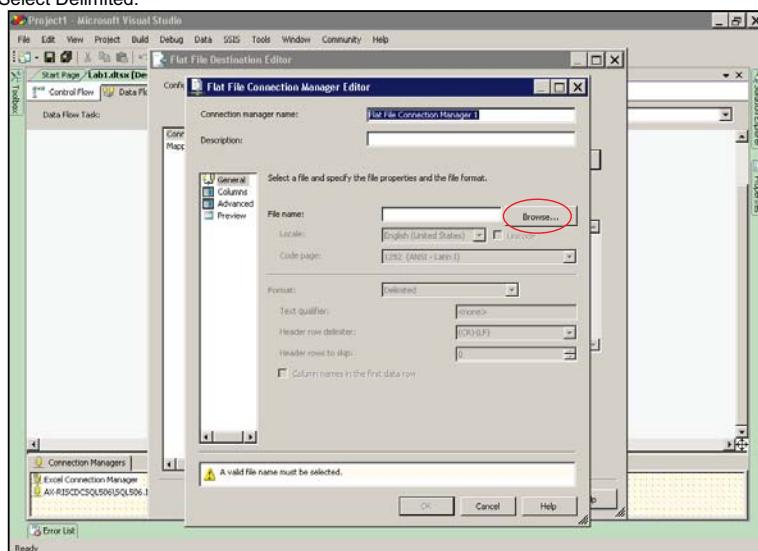


3)Select the column for which you need a copy and click OK.

4)Select a Flat file destination and double click to view editor.

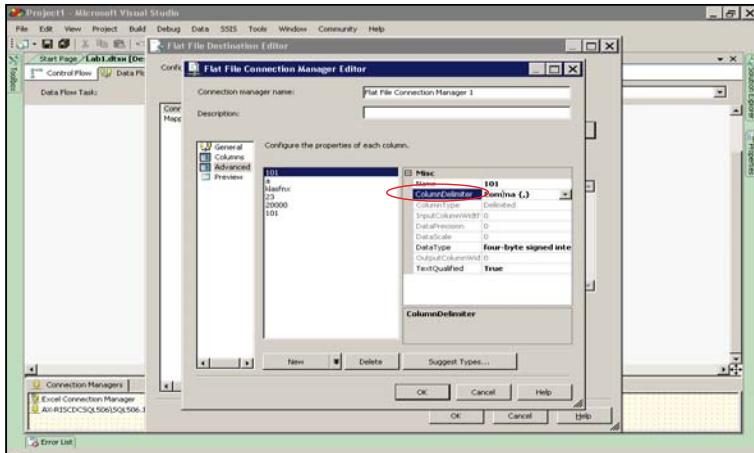


- 5)Click on NEW to get the format.
6)Select Delimited.

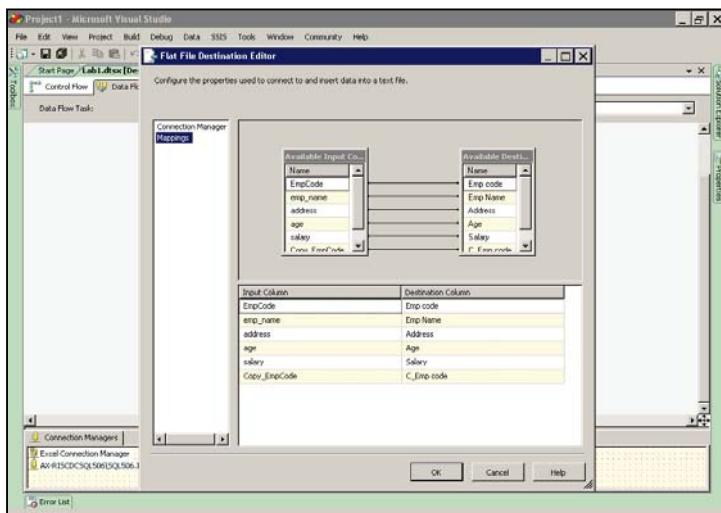


7)Browse for a file where you want to load the data.

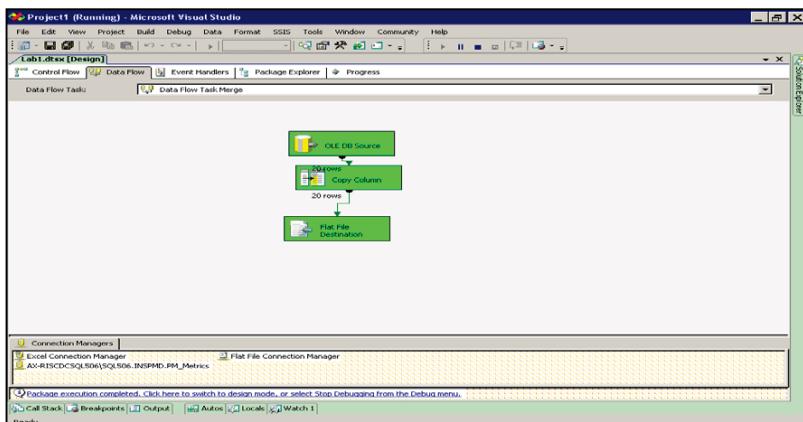
- ➔ Usually the column delimiter is Comma (,) which can be changed in Columns or advanced option.



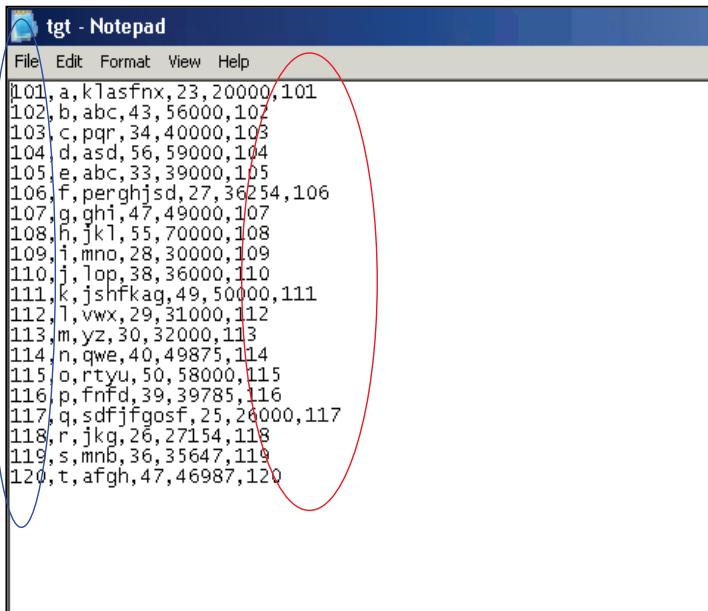
8)Select mapping and map the corresponding columns.



9)Save and execute the data flow.



Output:



tgt - Notepad

File Edit Format View Help

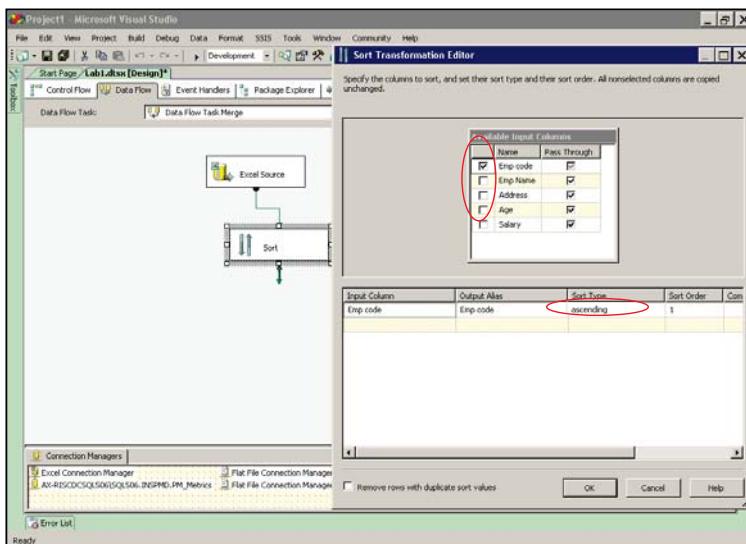
```
101,a,klasfnx,23,20000,101
102,b,abc,43,56000,102
103,c,pqr,34,40000,103
104,d,asd,56,59000,104
105,e,abc,33,39000,105
106,f,perghjsd,27,36254,106
107,g,ghi,47,49000,107
108,h,jkl,55,70000,108
109,i,mmo,28,30000,109
110,j,top,38,36000,110
111,k,jshfkag,49,50000,111
112,l,vwx,29,31000,112
113,m,yz,30,32000,113
114,n,qwe,40,49875,114
115,o,rtyu,50,58000,115
116,p,fnfd,39,39785,116
117,q,sdfjfgosf,25,26000,117
118,r,jkg,26,27154,118
119,s,mnb,36,35647,119
120,t,afgh,47,46987,120
```

Lab 7- Merge Transformation

Objective	To merge data from two different sources into one target.
Lab Setup	<ul style="list-style-type: none"> SSDT tool Existing SSIS project and Package

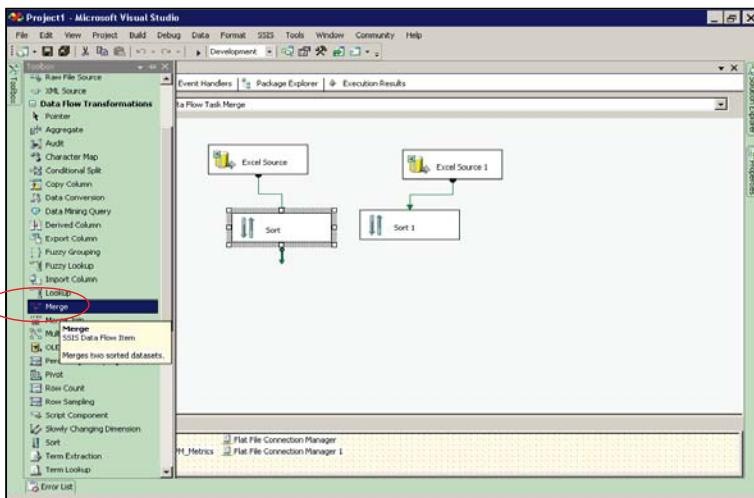
1)Select two excel file as source and edit the connection manager.

2)Select two sort transformations and edit the settings.

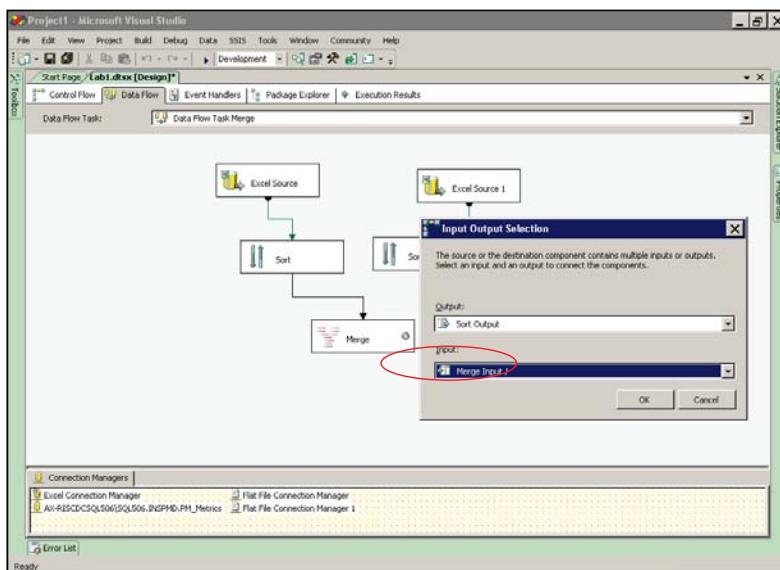


3)Sort both the source on the same column.

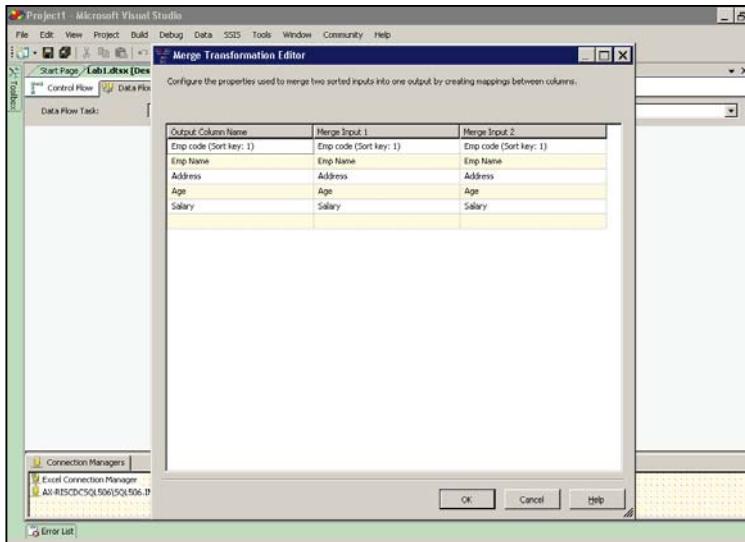
4)Then select merge transformation and join the sorted source to it.



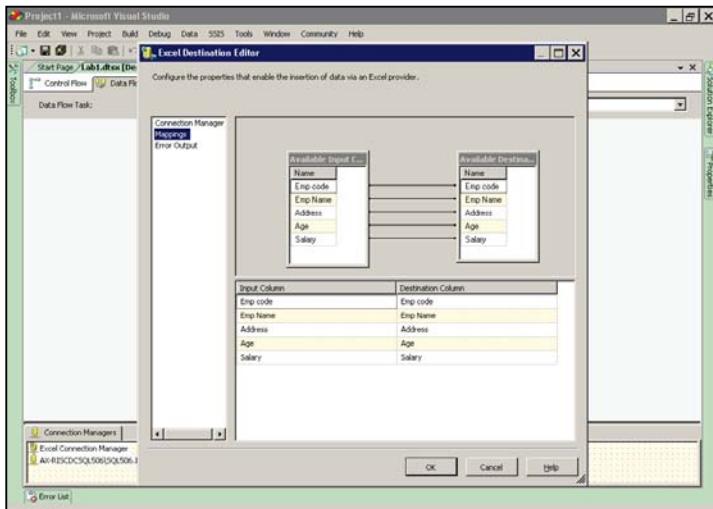
5)After joining the first SORT to the merge transformation, the window opens as:



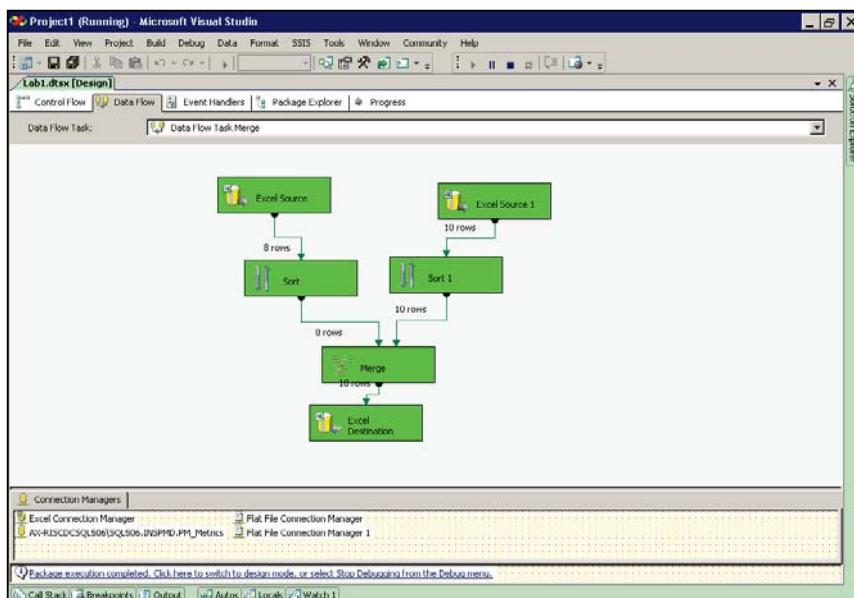
- 6)Select Sort Output and Merge Input 1
7)And then join the second sort to the Merge transformation.
8) Double click the merge transformation to view the editor.



- 9)Select a destination file (here we have taken excel as target)
→ Set the connection manager and map the columns in the Columns tab.



10)Save and execute the data flow task.



Data 1:

Emp code	Emp Name	Address	Age	Salary
112 l	vwx		29	31000
113 m	yz		30	32000
114 n	qwe		40	49875
101 a	klasfnx		23	20000
102 b	abc		43	56000
103 c	pqr		34	40000
104 d	asd		56	59000
105 e	abc		33	39000

Data 2:

Emp code	Emp Name	Address	Age	Salary
101 a	klasfnx		23	20000
102 b	abc		43	56000
103 c	pqr		34	40000

104 d	asd	56	59000
105 e	abc	33	39000
119 s	mnb	36	35647
120 t	afgh	47	46987
112 l	vwx	29	31000
113 m	yz	30	32000
114 n	qwe	40	49875

Output data:

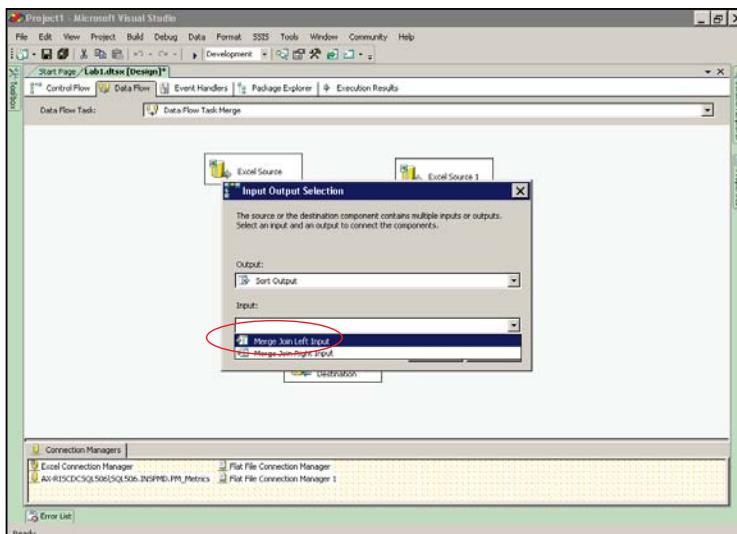
Emp code	Emp Name	Address	Age	Salary
101 a	klasfnx		23	20000
101 a	klasfnx		23	20000
102 b	abc		43	56000
102 b	abc		43	56000
103 c	pqr		34	40000
103 c	pqr		34	40000
104 d	asd		56	59000
104 d	asd		56	59000
105 e	abc		33	39000
105 e	abc		33	39000
112 l	vwx		29	31000
112 l	vwx		29	31000
113 m	yz		30	32000
113 m	yz		30	32000
114 n	qwe		40	49875
114 n	qwe		40	49875
119 s	mnb		36	35647
120 t	afgh		47	46987

Lab 8- Merge join Transformation

Objective	To merge data from two data sources and load into a target table using join condition.
Lab Setup	<ul style="list-style-type: none"> SSDT tool Existing SSIS project and Package

1) Repeat the steps from selecting the source till sorting of data and then drag and drop Merge Join Transformation.

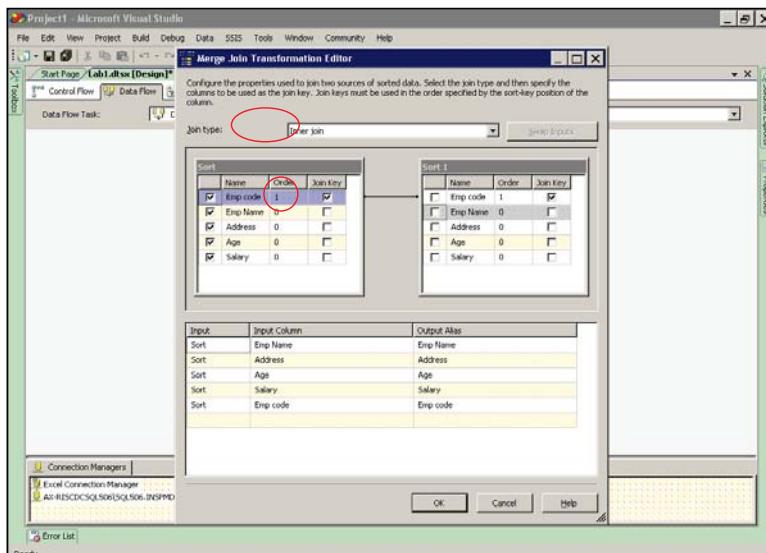
- Join the sort output to the transformation.
- The window opens as:



- Select the valid input type.
- Connect the second input to the transformation.

2) Double click merge join transformation to view the editor.

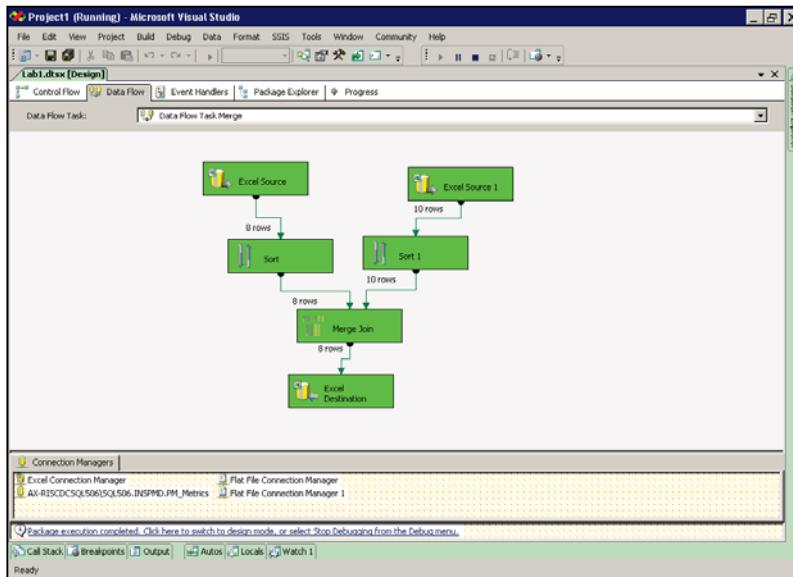
3) Select the required join type and the columns.



4) Connect the transformation to destination.

5) Edit the connection manager with valid connections and map the columns.

6) Save and execute the data flow.



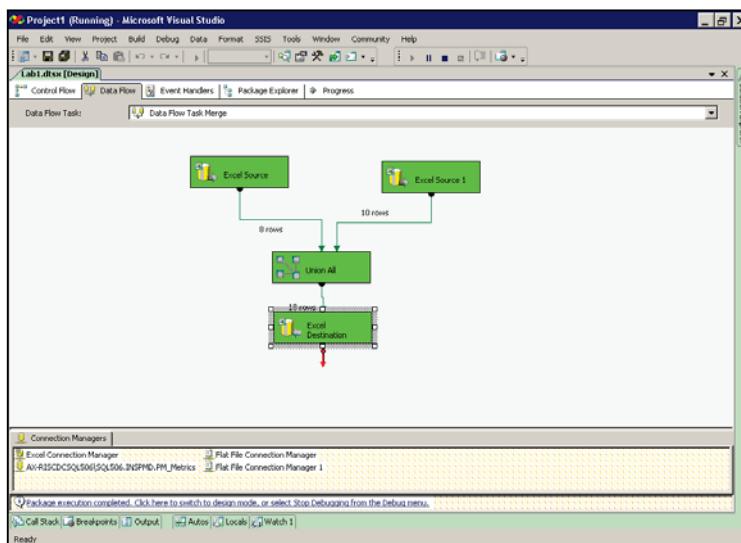
Output:

Emp Name	Address	Age	Salary	Emp code
a	klasfnx	23	20000	101
b	abc	43	56000	102
c	pqr	34	40000	103
d	asd	56	59000	104
e	abc	33	39000	105
f	vwx	29	31000	112
m	yz	30	32000	113
n	qwe	40	49875	114

Lab 9- Union All Transformation

Objective	To merge two data sources into one target table.
Lab Setup	<ul style="list-style-type: none"> SSDT tool Existing SSIS project and Package

- 1)Select union all from the toolbox and connect it to the sources.
- 2)Select destination and join the union transformation to it. Edit the connection manager mapping.
- 3)Save and execute the data flow.



Output:

Emp code	Emp Name	Address	Age	Salary
112 l	vwx		29	31000
113 m	yz		30	32000
114 n	qwe		40	49875
101 a	klasfnx		23	20000
102 b	abc		43	56000
103 c	pqr		34	40000
104 d	asd		56	59000
105 e	abc		33	39000

101 a	klasfnx	23	20000
102 b	abc	43	56000
103 c	pqr	34	40000
104 d	asd	56	59000
105 e	abc	33	39000
119 s	mnb	36	35647
120 t	afgh	47	46987
112 l	vwx	29	31000
113 m	yz	30	32000
114 n	qwe	40	49875

Notes:

The Merge Join transformation provides an output that is generated by joining two sorted datasets using a FULL, LEFT, or INNER join.

The Merge transformation combines two sorted datasets into a single dataset. The rows from each dataset are inserted into the output based on values in their key columns.

The Merge transformation is similar to the Union All transformations. Use the Union All transformation instead of the Merge transformation in the following situations:

- The transformation inputs are not sorted.
- The combined output does not need to be sorted.
- The transformation has more than two inputs.

Lab 10-Slowly changing Dimension

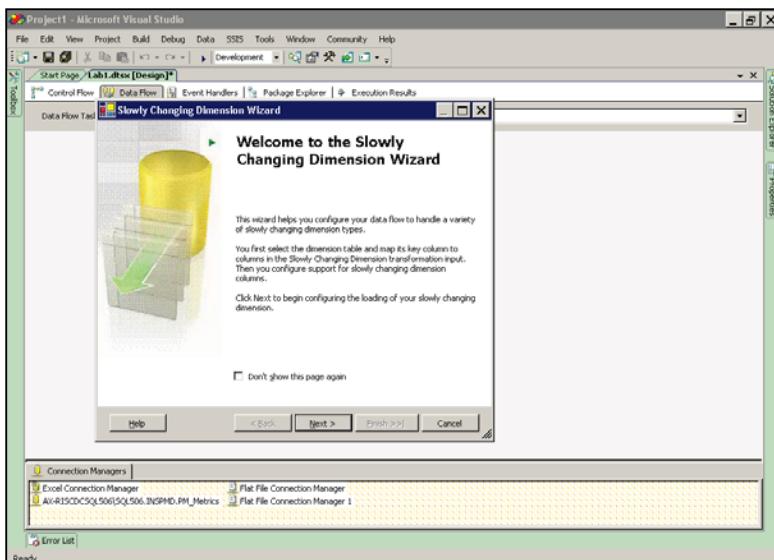
Objective	To maintain the database in such a way that if any change occurs in the old data then the previous data is also shown along with the updated new data (Using Wizard)
Lab Setup	<ul style="list-style-type: none"> • SSDT tool • Existing SSIS project and Package

This transformation is used to maintain historical data.

1)Select a source (here we have used OLE DB)

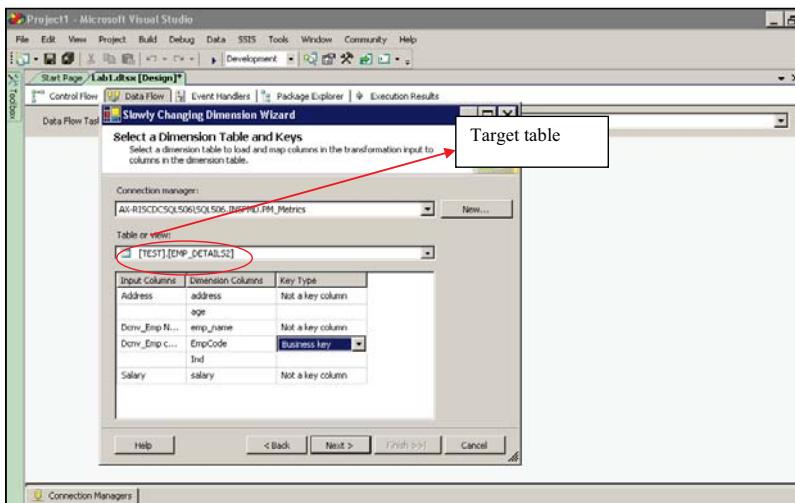
2)Select Slowly Changing dimension transformation, Connect the source to it.

- Double click the transformation to start the wizard.

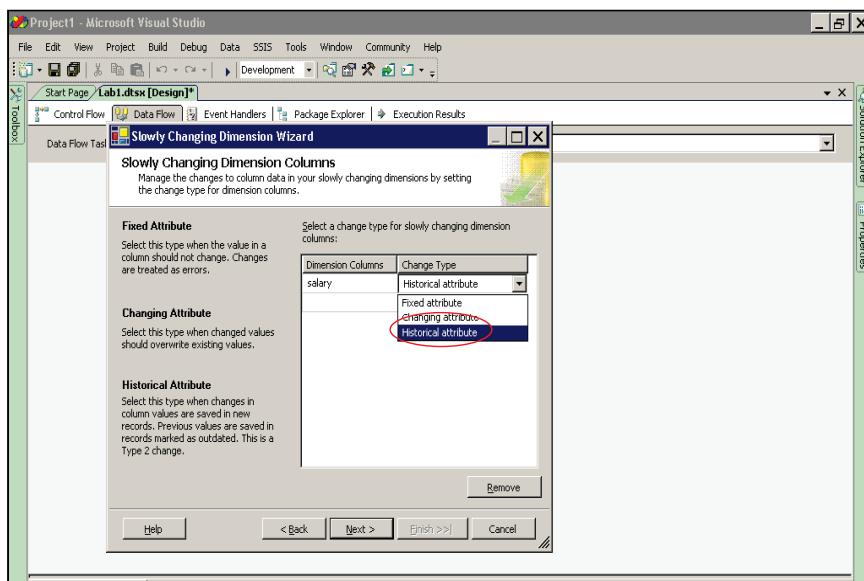


- Click on NEXT .

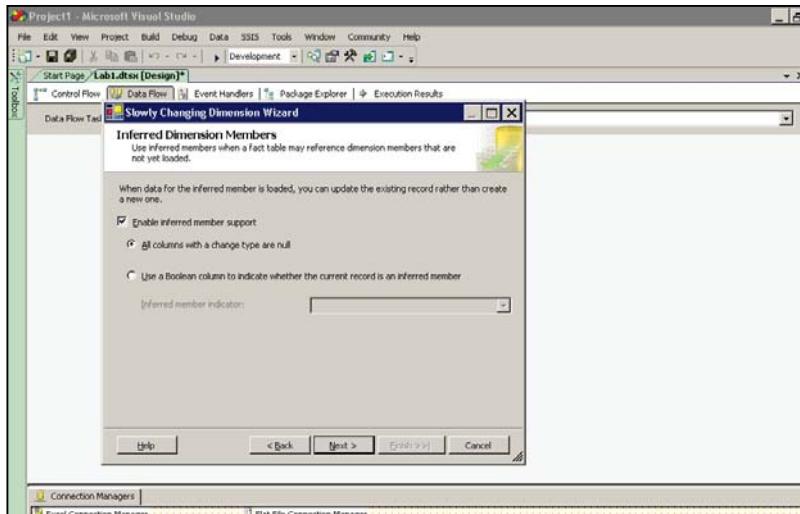
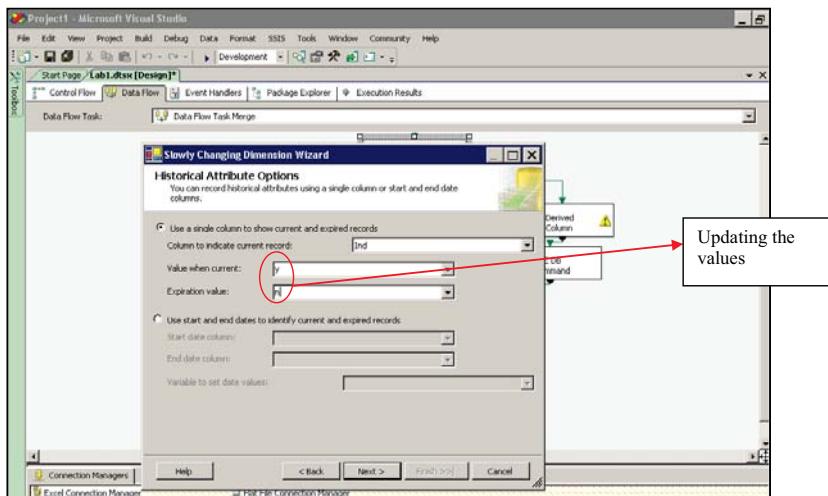
- Select a target table and then select input columns and use emp code as business key.



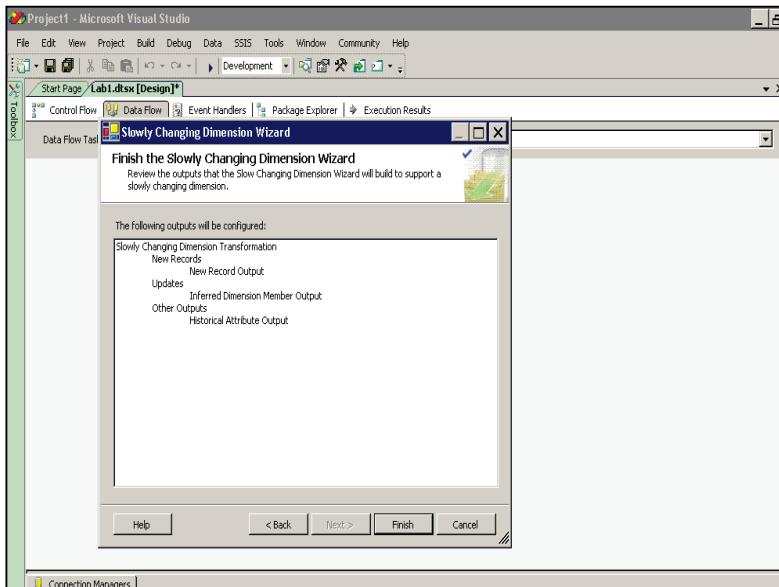
- Now click on NEXT, select the column where the data changes and change type to **historical attribute**.



- ➔ Click NEXT.
- ➔ Use a column to differentiate new and old values.
Set the required fields and then click NEXT.

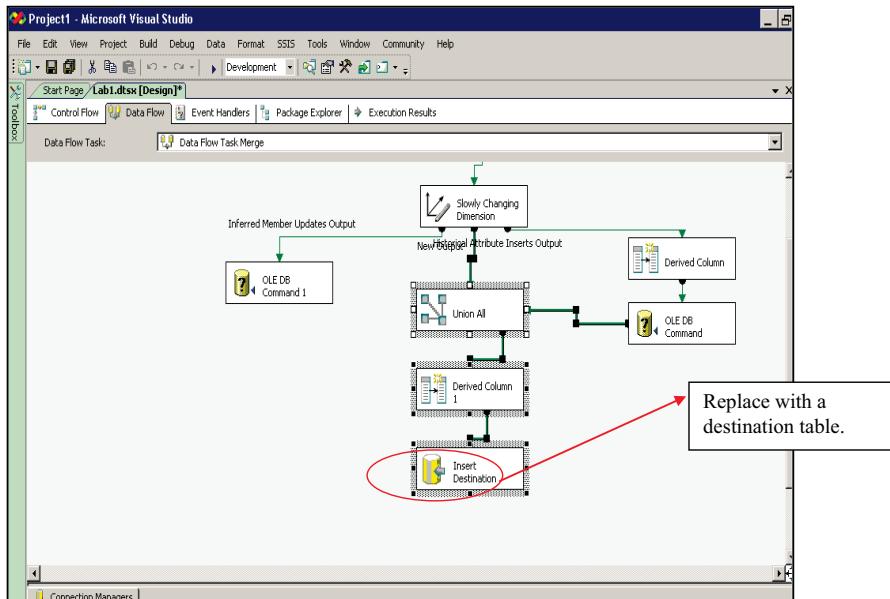


➔ Click NEXT



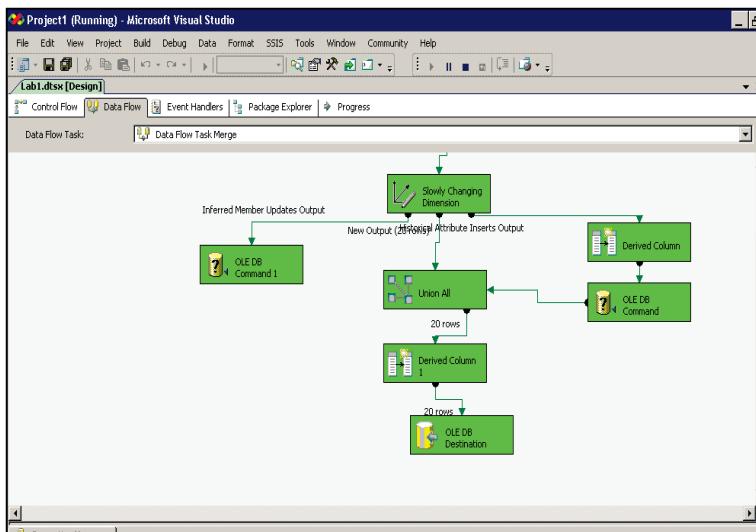
➔ Click FINISH.

The output for the wizard will be like:



3) Update the destination with the target table.

4) Save and run the data flow.


Initial Load will show the data as:

Emp_code	emp_name	Address	Age	Salary	IND
101	a	klasfnx	23	20000	y
102	b	abc	43	56000	y
103	c	pqr	34	40000	y
104	d	asd	56	59000	y
105	e	abc	33	39000	y
106	f	perghjsd	27	36254	y
107	g	ghi	47	49000	y
108	h	jkl	55	70000	y
109	i	mno	28	30000	y
110	j	lop	38	36000	y
111	k	jshfkag	49	50000	y
112	l	vwx	29	31000	y
113	m	yz	30	32000	y
114	n	qwe	40	49875	y
115	o	rtyu	50	58000	y
116	p	fnfd	39	39785	y
117	q	sdfjfgosf	25	26000	y
118	r	jkg	26	27154	y
119	s	mnb	36	35647	y

120 t afgh 47 46987 y

Updating the rows:

```

update test EMP_DETAILS
SET salary=23465 where EmpCode=101
update test EMP_DETAILS
SET salary=50200 where EmpCode=107
update test EMP_DETAILS
SET salary=26987 where EmpCode=117

```

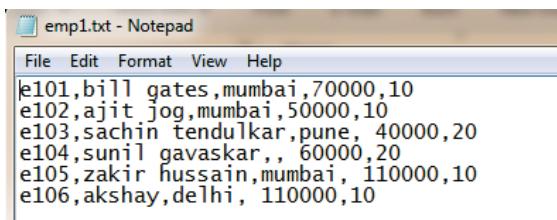
After changes in salary:

Emp_code	emp_name	Address	Age	Salary	IND
101 a	klasfnx		23	23564	n
102 b	abc		43	56000	y
103 c	pqr		34	40000	y
104 d	asd		56	59000	y
105 e	abc		33	39000	y
106 f	perghjsd		27	36254	y
107 g	ghi		47	50000	n
108 h	jkl		55	70000	y
109 i	mno		28	30000	y
110 j	lop		38	36000	y
111 k	jshfkag		49	50000	y
112 l	vwx		29	31000	y
113 m	yz		30	32000	y
114 n	qwe		40	49875	y
115 o	rtyu		50	58000	y
116 p	fnfd		39	39785	y
117 q	sdfjfgosf		25	27564	n
118 r	jkg		26	27154	y
119 s	mnb		36	35647	y
120 t	afgh		47	46987	y
101 a	klasfnx		23	23465	y
107 g	ghi		47	50200	y
117 q	sdfjfgosf		25	26987	y

Lab 11- Extracting, Transforming and Loading Data

Objective	To learn <ol style="list-style-type: none"> 1. How to extract data from flat file 2. How to Transform Data <ol style="list-style-type: none"> a. Add new Derived Column b. Change Data Types c. Perform a Data Lookup d. Conditionally route and load data into relational table 3. Load data into relational table
Lab Setup	<ul style="list-style-type: none"> • SSDT tool • Existing SSIS project and connection to database

- 1)Create a separate folder “Demo2” for placing input data feed files
2)Place a emp1.txt data file in this folder with following data



```
emp1.txt - Notepad
File Edit Format View Help
e101,bill gates,mumbai,70000,10
e102,ajit jog,mumbai,50000,10
e103,sachin tendulkar,pune, 40000,20
e104,sunil gavaskar,, 60000,20
e105,zakir hussain,mumbai, 110000,10
e106,akshay,delhi, 110000,10
```

IN SSMS:

- 3)Create “mumbaiemp” table in sql server database with following structure

```
CREATE TABLE [dbo].[mumbaiemp](
[empno] [varchar](7)PRIMARYKEY,
[empname] [varchar](20)NULL,
[city] [varchar](20)NULL,
[sal] [money] NULL,
[da] [money] NULL,
[hra] [money] NULL,
[deptno] [int] NULL,
[dname] varchar(20)
)
```

- 4)Create 2 more tables “puneemp” and “otheremp” with exact same structure as above.
5)Create dept table as below:

```
createtable dept
(
deptno intprimarykey,
deptname varchar(20)
)
```

- 6)Add the following records to the dept table:

	deptno	deptname
1	10	finance
2	20	sales
3	30	accounts
4	40	purchase
5	50	IT

IN SSIS:

- 1)In SSDT ,create New SSIS Package
 - Rename it as "FF_To_DB_With_Transformation.dtsx"
- 2)In the package designer, goto data flow tab and click the link in the middle to create a new data flow task.
- 3)Same as was done in Lab 1, drag a flat file source task from toolbox onto designer surface and make it point to emp1.txt file placed in Demo2 folder.
- 4)Next, drag derived column task from transformation section of toolbox
 - Connect the flat file task green connector to this task
 - Double click this task and configure the additional derived columns as shown below:

Derived Column Name	Derived Column	Expression	Data Type	Length
FF_Col_DA	<add as new column>	FF_Col_EmpSal * 0.20	numeric [DT_NUMERIC]	
FF_Col_HRA	<add as new column>	FF_Col_EmpSal * 0.25	numeric [DT_NUMERIC]	

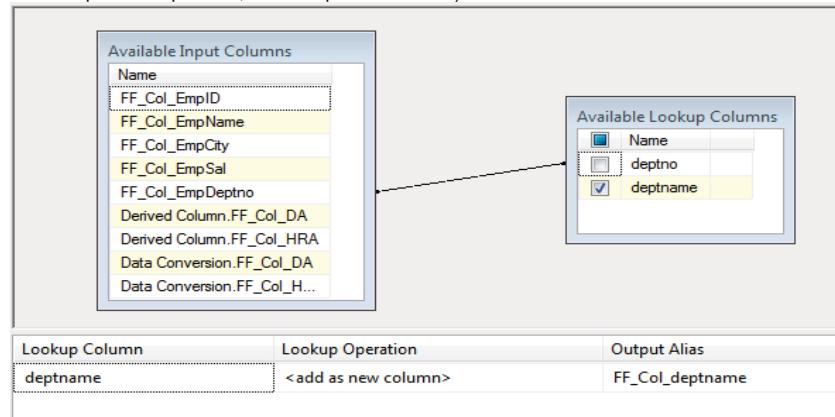
- 5)Drag a Data Conversion task and connect the Derived Column task green connector to this task.
 - Double click it and change the data type of the following to columns from DT_Numeric to DT_CY as shown below:

Input Column	Output Alias	Data Type	Length
FF_Col_DA	FF_Col_DA	currency [DT_CY]	
FF_Col_HRA	FF_Col_HRA	currency [DT_CY]	

- 6)Drag a lookup transform task and connect the data conversion task to it.
 - Double click lookup task, go to connection tab, create a new connection manager which will point to the sql server database which has "Dept" table created earlier in this lab.

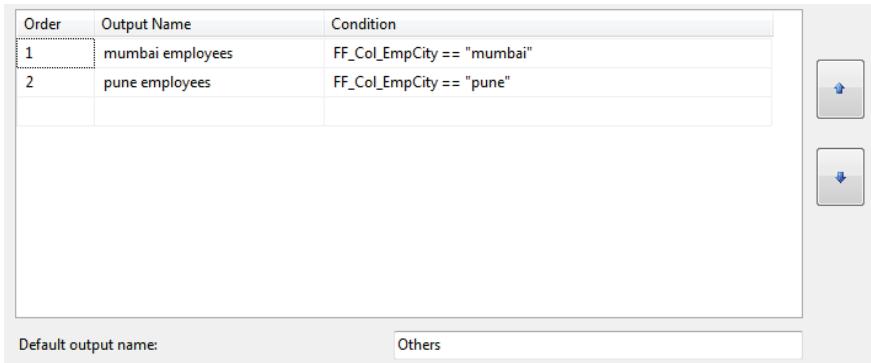


- Goto columns tab, and configure this tab as shown below. (connect FF_Col_EmpDeptno to Deptno in Dept Table, select deptname column)



7)Drag a conditional split task onto surface rename it as “Split Employee records based on city”, connect the green connector of lookup task to this task a dialog will popup, from drop down select “lookup match output”.

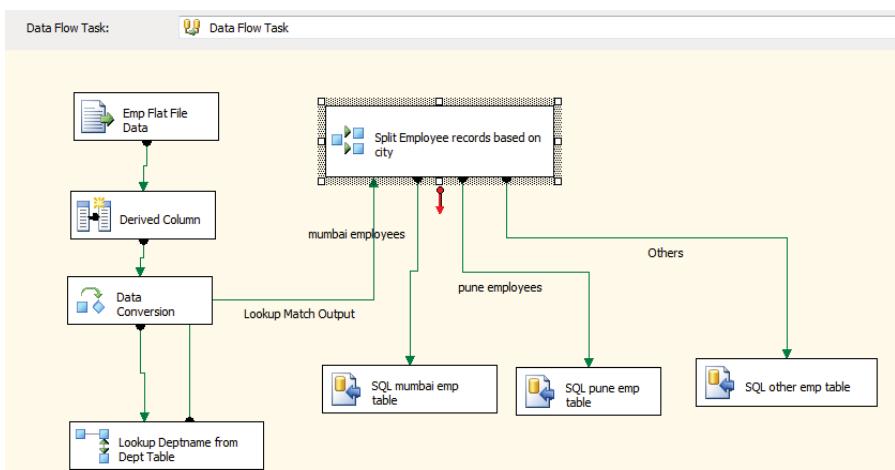
- Double click conditional split task and configure as below: (there are 2 named outputs which will send only those rows that satisfy the specified condition, and there is default output as well called “others”)



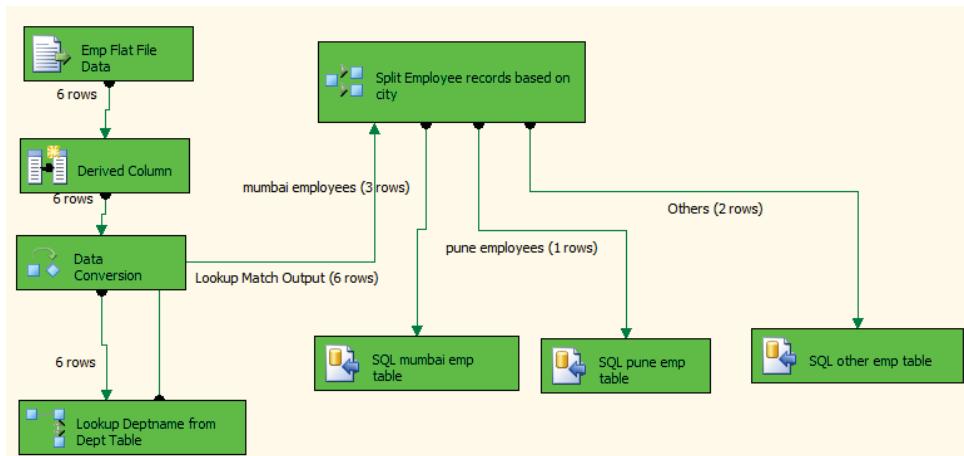
8) Drag 3 SQL Server Destinations tasks from data flow destinations section of toolbox. Rename them as "SQL Mumbai emp table", SQL Pune Emp Table, SQL Other emp table.

9) Connect conditional split task to the first "SQL Mumbai emp table" sql server destination task. A dialog box will popup select the output name to connect as "mumbai employees". Similarly connect the remaining outputs to the corresponding tasks.

10) Configure each of the these tasks by double clicking them so that they point to appropriate tables in the sql server. The complete SSIS package is shown below:

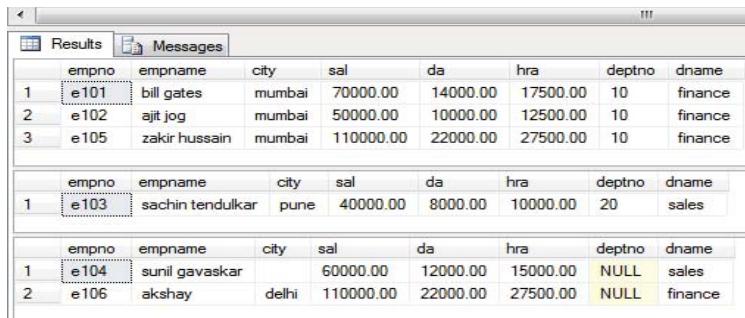


11) Run the package, when the entire package executes successfully it will look as shown below:



12)Stop the Package. 

13)Connect to sql server using SSMS and check the table data



The screenshot shows the SSMS Results pane displaying three tables of employee data:

	empno	empname	city	sal	da	hra	deptno	dname
1	e101	bill gates	mumbai	70000.00	14000.00	17500.00	10	finance
2	e102	ajit jog	mumbai	50000.00	10000.00	12500.00	10	finance
3	e105	zakir hussain	mumbai	110000.00	22000.00	27500.00	10	finance

	empno	empname	city	sal	da	hra	deptno	dname
1	e103	sachin tendulkar	pune	40000.00	8000.00	10000.00	20	sales

	empno	empname	city	sal	da	hra	deptno	dname
1	e104	sunil gavaskar		60000.00	12000.00	15000.00	NULL	sales
2	e106	akshay	delhi	110000.00	22000.00	27500.00	NULL	finance

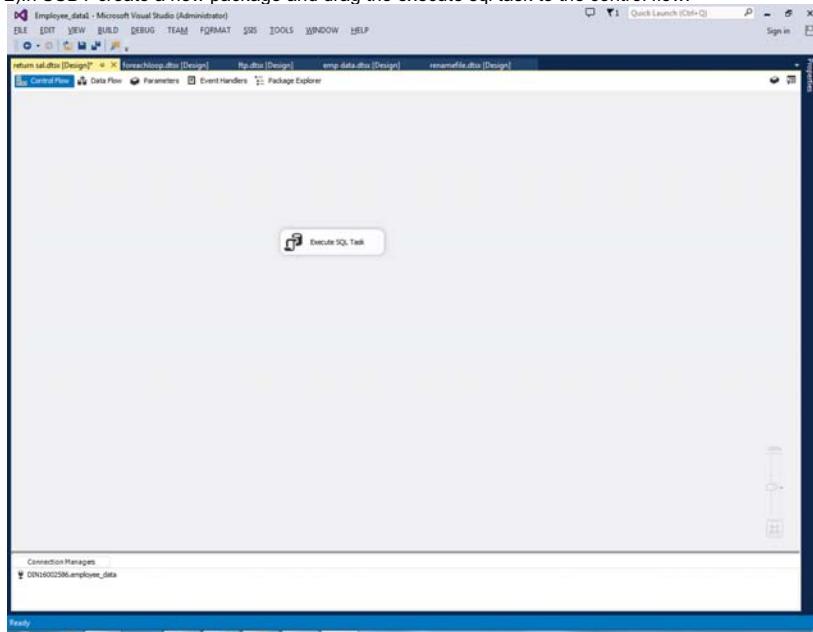
Lab 12- Execute SQL Task& Script task

Objective	Execute a stored procedure from SSIS to get employee salary and display output in message box
Lab Setup	<ul style="list-style-type: none"> • SSDT tool • Existing SSIS project and connection to database

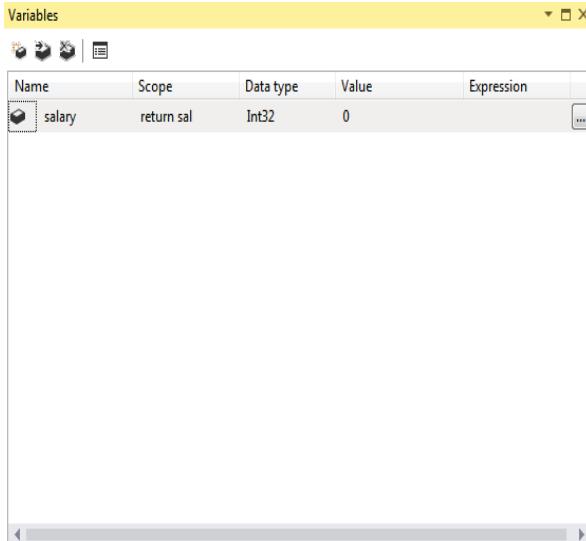
1)As per below create a stored procedure to get employee salary as outcome when passing salary name as input parameter.

```
createprocedure [dbo].[emp_name9](
@emp_name varchar(20),
@salary intoutput
)
as
begin
select @salary=[salary] from [dbo].[empdata] where [emp_name]=@emp_name
end
```

2)In SSDT create a new package and drag the execute sql task to the control flow.

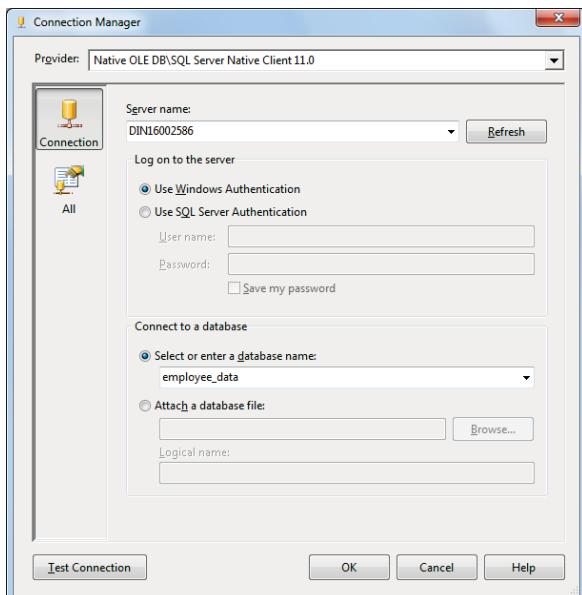


3)Create a variable as per below. You can create variables by right clicking on the control flow or from the SSIS menu above the control flow.

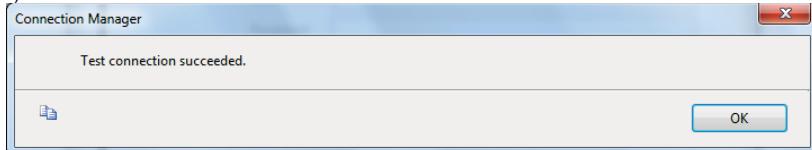


Name	Scope	Data type	Value	Expression
salary	return sal	Int32	0	[...]

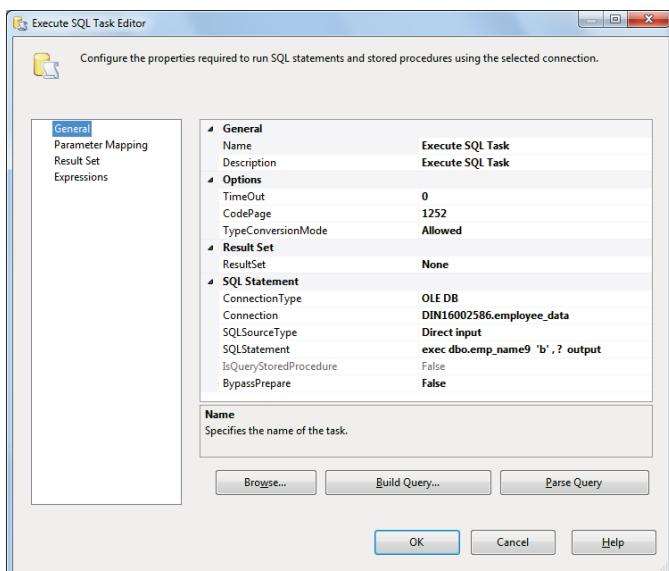
4)Create a new connection string as per below. If connection string already exists no need to create new connection string, you can use the same connection string.



- 5)Choose the provider as SQL Server Native Client 11.0.
- 6)Choose your server name and database which holds your employee_data table.
- 7)Click the Test Connection tab and test the connection.



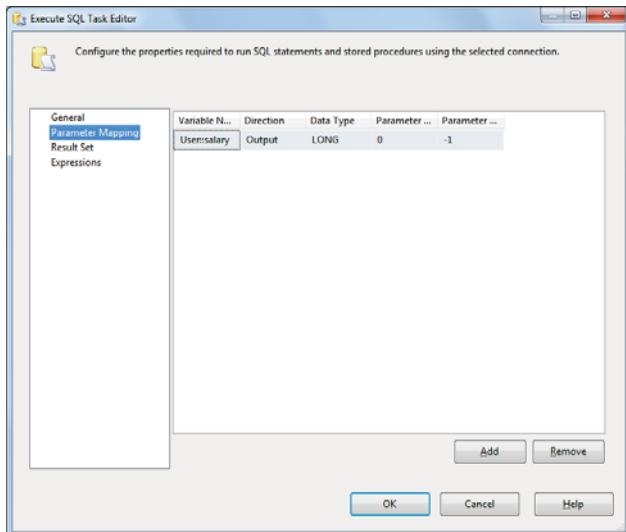
- 8)Edit the Execute SQL task and choose the connection string which you have created and SQL Source type as Direct Input as per below.



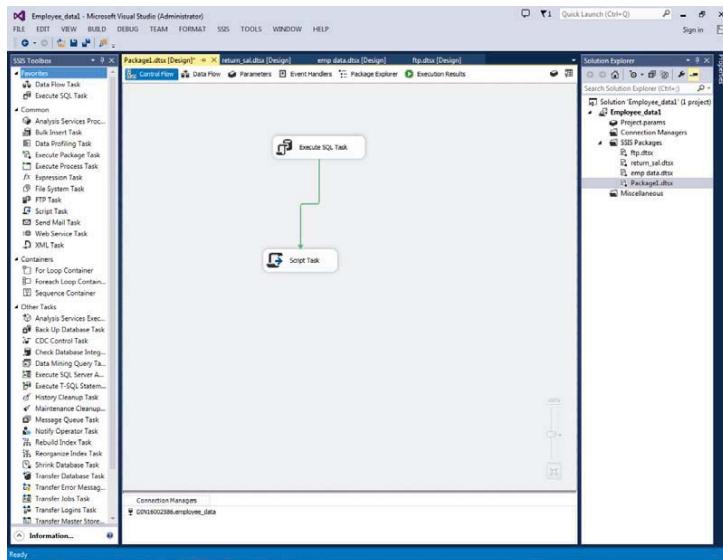
5)Provide the execute statement in SQLStatement field as per below.

SQL Statement:exec dbo.emp_name9 'p', ? output

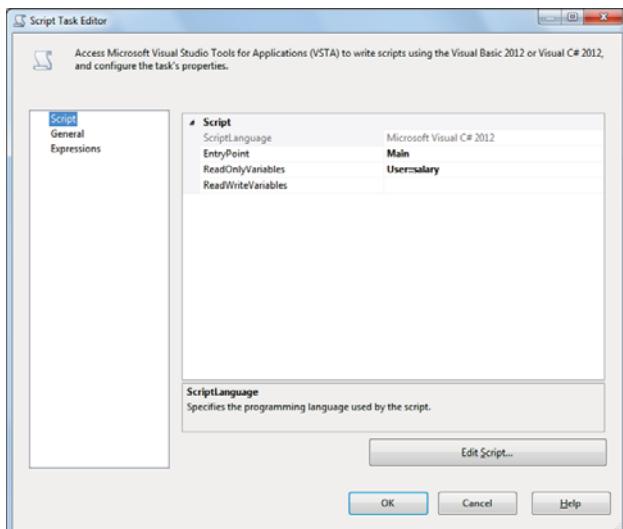
6)In parameter mapping choose the parameter you have created and give the direction as output and parameter name as 0.



- 7)Now will use Script task to display salary of the Employee that we have stored in output variable.
- 8)Drag a Script task to the control flow and connect the green arrow from execute sql task to the script task.



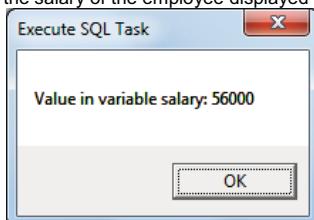
9)Now edit the script task and in the read only variable choose the output parameter **User::salary**



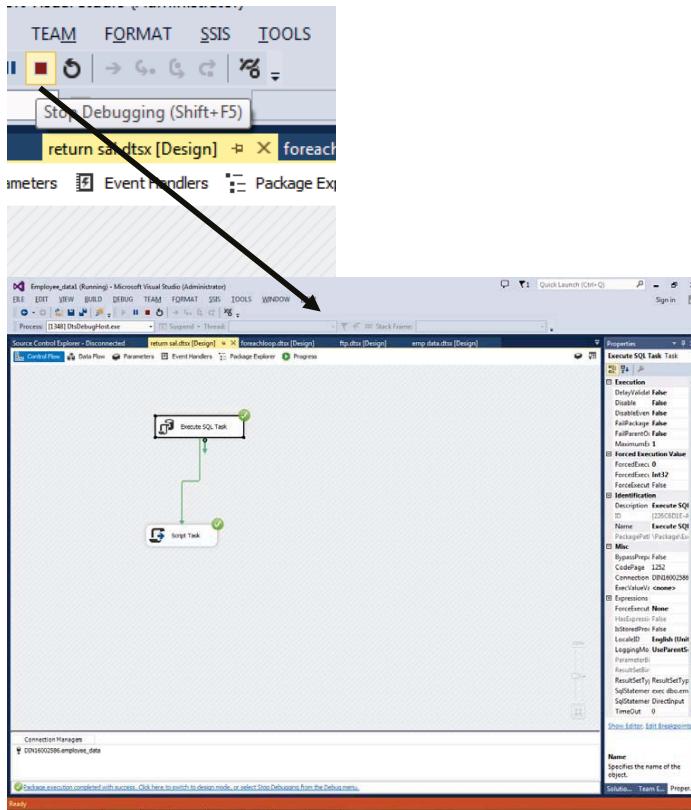
10)In Edit Script and add the below code in main function to display the salary of the employee.

```
MessageBox.Show(String.Format("Value in variable salary: {0}",  
Dts.Variables["User::salary"].Value), "Execute SQL Task");
```

11)Now execute the package by right clicking the project name in the solution explorer and you can see the salary of the employee displayed in the message box.



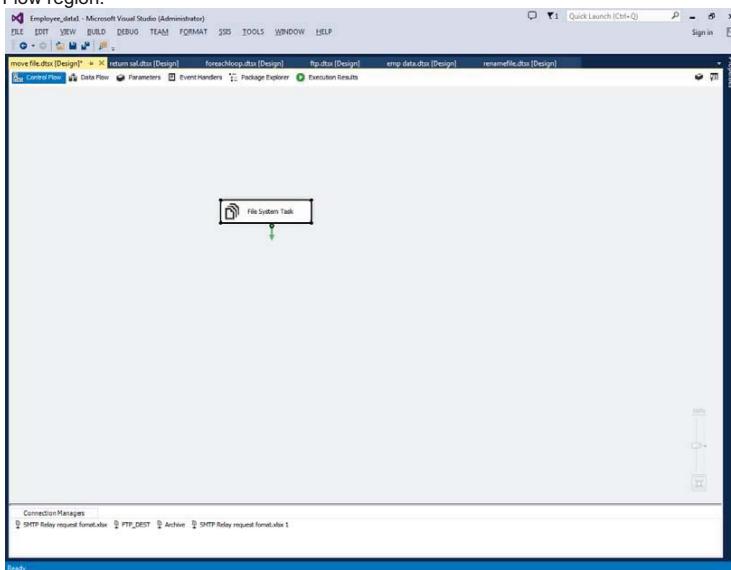
12)After executing stop debugging to return to the design mode.



Lab 13-File System Task

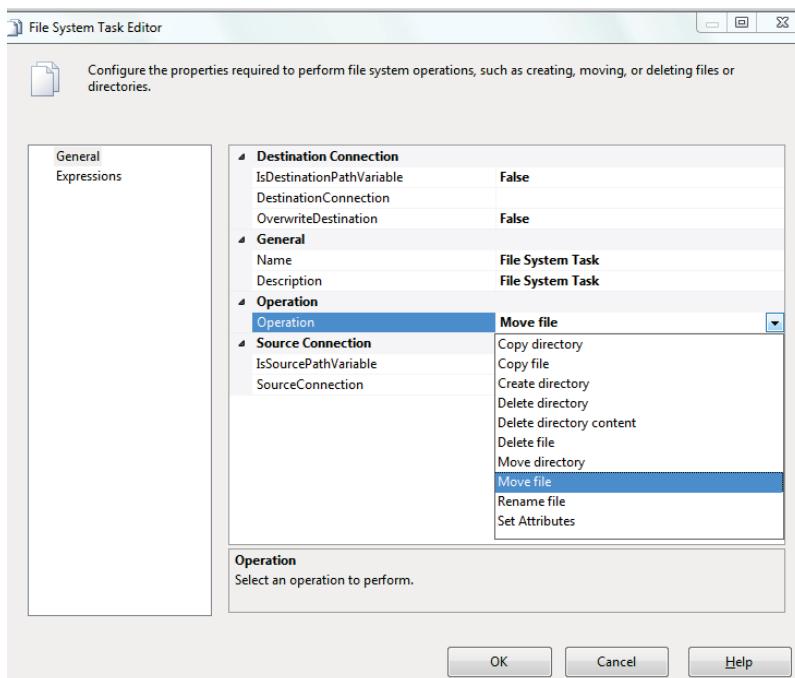
Objective	To learn 1.How to move file from one location to another location.
Lab Setup	<ul style="list-style-type: none">• SSDT tool• Create package

1)In order to move file using file system task , drag and drop the File System Task into the Control Flow region.



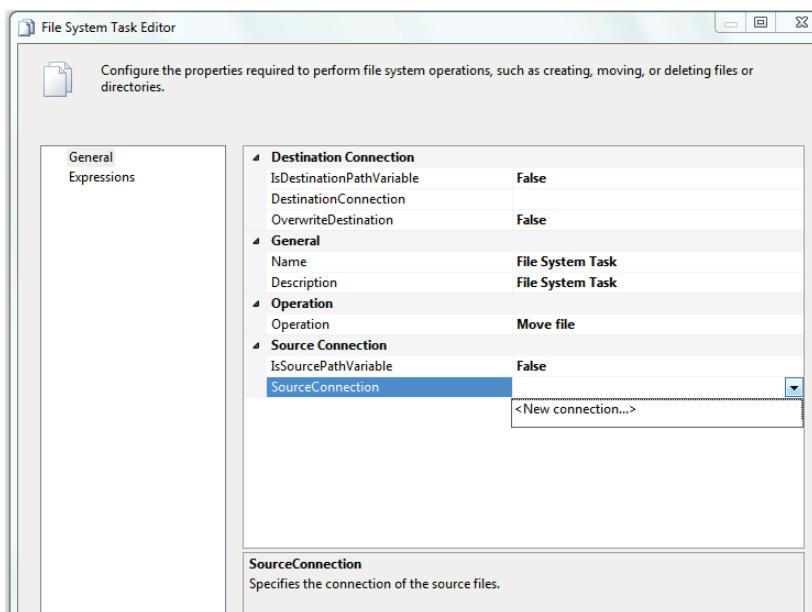
2) Double click on it will open the File System Task Editor to configure it.

3)In this example, We are Moving single file so, Please change the **operation** property to **Move File** as shown in the below screenshot.

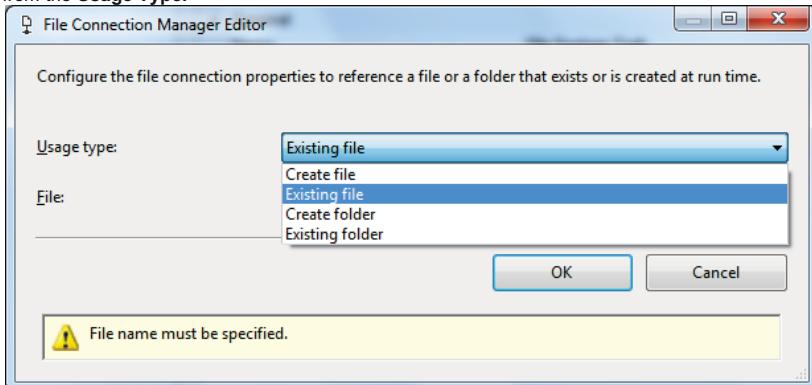


4) Let us configure the Source Connection by selecting the **Source Connection** property.
5) If you already created the File Connection Manager before then select the created one or If you stored the Source Connection in the Variable then, please change the **IsSourcePathVariable** property to TRUE and select the Variable Name.

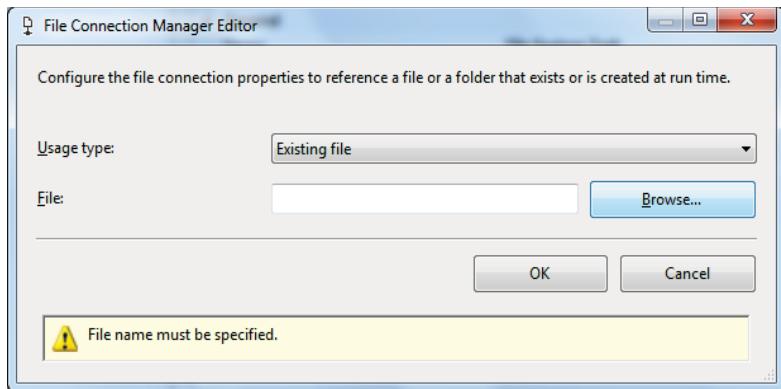
6) Here, We haven't created any connection Manager before so, We are selecting <New Connection...>



7)Once you click on the <New Connection..> option, File Connection Manager Editor will be opened to configure it. In this example we are Moving existing file so we are selecting Existing File option from the Usage Type.

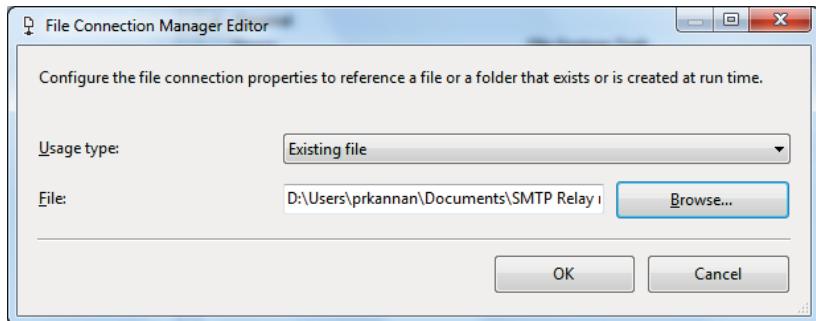


8)Click on the Browse button to select the Existing File from the file system.



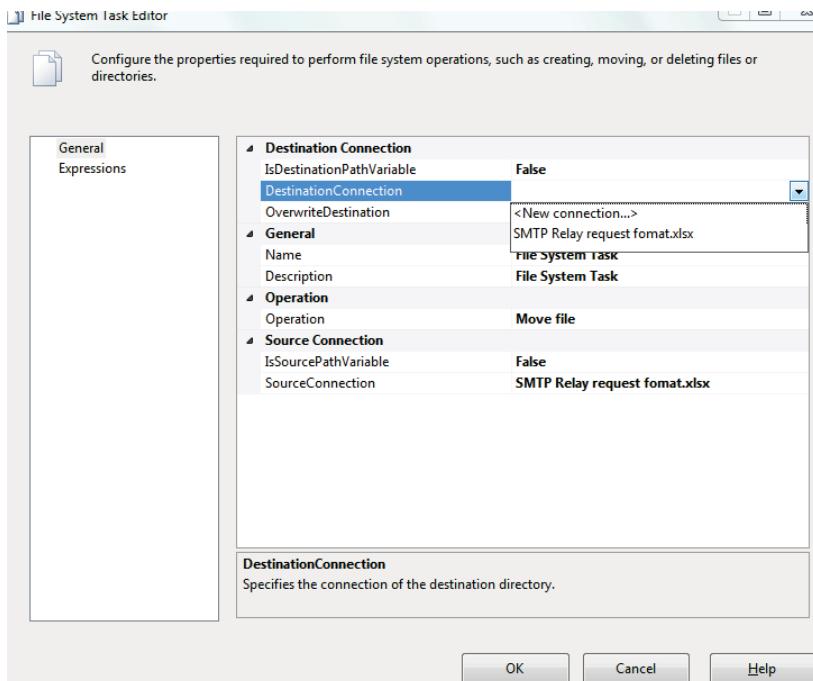
9)

From the above screenshot you can observe that, We selected the SMTP Relay request format.xlsx file inside the File System Task Folder.

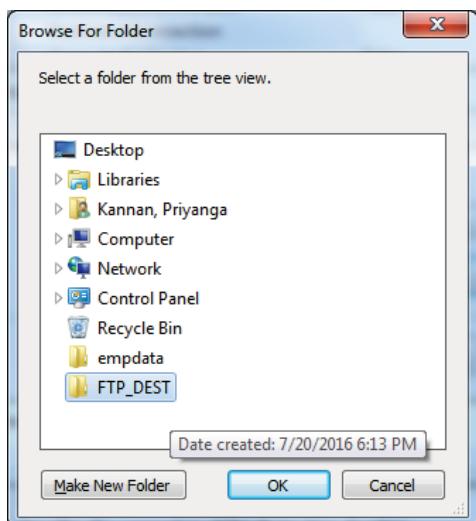


10)Now we have to configure the Destination Connection so, Please select the **DestinationConnection** property.

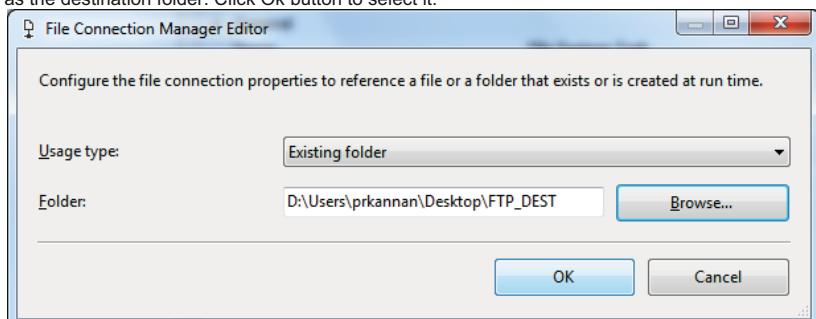
11)We haven't created any connection Manager before so, We are selecting <New Connection..>



12)In this example we are Moving existing file to the already existing folder so, we are selecting Existing Folder option from the **Usage Type**.



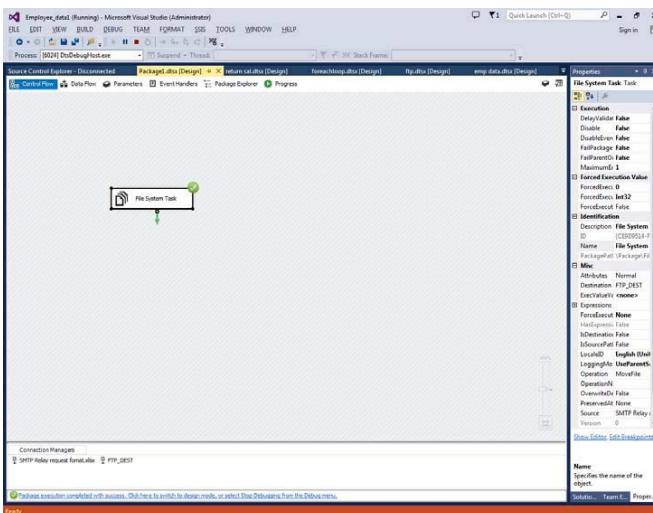
13) From the above the above screenshot you can observe that, We selected the Copied FTP_DEST as the destination folder. Click Ok button to select it.



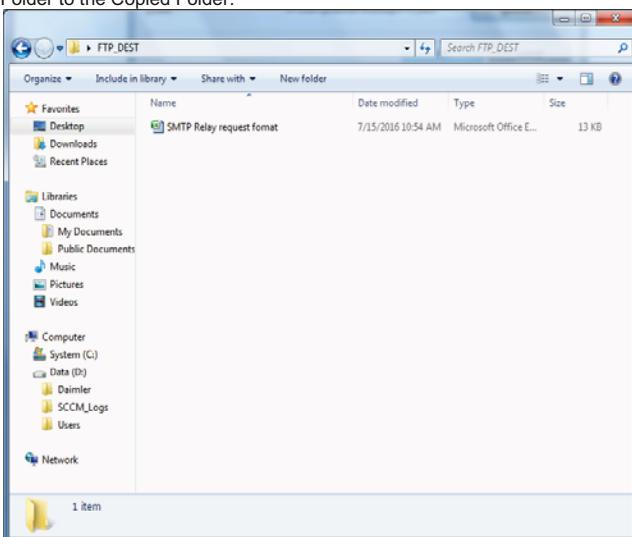
14) Click Ok to finish configuring the File Connection Manager for the destination.

15) Click Ok to finish configuring Move File using File System Task in SSIS package.

16) Click Ok to finish configuring Move File using File System Task in SSIS package. Let's run and see whether we successfully Moved the Customers.txt file using the File System Task or Not.



- 17) We successfully Moved the SMTP Relay request format.xlsx file present in the File System Task Folder to the Copied Folder.



Lab 14-For each loop container with Execute SQL task and Script task

Objective	To learn <ol style="list-style-type: none"> 1. How to make use of control flow tasks 2. Use of For Each Loop container task as result set iterator 3. Calling Stored Proc 4. Using a Custom Script Task Component 5. Using DTS Variables
Lab Setup	<ul style="list-style-type: none"> • SSDT tool • Existing project and connection to database

- 1.Create a separate folder “Demo3”. Create a script CreateTab.sql and put the following T-SQL code in it:
2.Connect to SQL server using SSMS and create the following tables

```

CREATE TABLE [dbo].[Department](
    [DepartmentID] [int] PRIMARY KEY,
    [Name] [nvarchar](50) NOTNULL,
    [Budget] [money] NOTNULL
)
CREATE TABLE [dbo].[Course](
    [CourseID] [int] PRIMARY KEY,
    [Title] [nvarchar](100) NOTNULL,
    [DepartmentID] [int] NOTNULL
)

```

- 3.Add the following records into the above tables:

Results			Messages	
	DepartmentID	Name	Budget	
1	1	Engineering	350000.00	
2	2	English	120000.00	
3	4	Economics	200000.00	
4	7	Mathematics	250000.00	
	CourseID	Title	DepartmentID	
1	1045	Calculus	7	
2	1050	Chemistry	1	
3	1061	Physics	1	
4	2021	Composition	2	
5	2030	Poetry	2	
6	2042	Literature	2	
7	3141	Trigonom...	7	
8	4022	Microeco...	4	
9	4041	Macroeco...	4	
10	4061	Quantitative	4	

4.Create a folder "Demo3", and create a script "createproc.sql" in this folder. Put the following code in the script

```

ifOBJECT_ID('getcourses')isnotnull
dropprocedure GetCourses
go
createprocedure GetCourses
(
    @dno int,
    @coursecnt intoutput
)
as
    select @coursecnt =count(*)
        from course where departmentid = @dno
go

```

5.Add a new ssis package, Project => New SSIS package, name it "ExecuteSQLParametersResultSets.dtsx".

6.In the control flow tab designer, drag a "Execute SQL Task". Rename it as "Create Stored Proc"

- double click task, set the following properties:
 - connection type: oledb
 - sql source type: file connection.
 - connection: create new connection and connect to your sql server.
 - file connection: create a new connection which points to the "createproc.sql" script file created in step (4).

General	
Name	Create Stored Proc
Description	Execute SQL Task
Options	
TimeOut	0
CodePage	1252
Result Set	
ResultSet	None
SQL Statement	
ConnectionType	OLE DB
Connection	pkplaptop_2285.ssisdemos
SQLSourceType	File connection
FileConnection	createproc.sql
IsQueryStoredProcedure	False
BypassPrepare	True

7. Select the package, by single clicking the package designer surface in free area, Open Variable Window, View => Other Windows => Variables

- b. Create a variable "DeptRS" clicking top left small button in the variables window, set datatype to "Object"
- c. Create 3 more variables Deptno Datatype Int32, DeptName datatype: string, CourseCnt datatype int32

Variables			
Name	Scope	Data Type	Value
DeptName	ExecuteSQLParametersResultSets	String	
Deptno	ExecuteSQLParametersResultSets	Int32	0
DeptRS	ExecuteSQLParametersResultSets	Object	System....
CourseCnt	ExecuteSQLParametersResultSets	Int32	0

8. Drag another "Execute SQL Task". Rename it as "Execute Department Query"

- d. Double click task, set the following properties:
 - i. ResultSet: Full Result Set
 - ii. connection type: oledb
 - iii. sql source type: direct input
 - iv. SQL Statement: SELECT * FROM department

Configure Execute SQL Task Properties:

General	
Name	Execute Department Query
Description	Execute SQL Task
Options	
TimeOut	0
CodePage	1252
Result Set	
ResultSet	Full result set
SQL Statement	
ConnectionType	OLE DB
Connection	pkplaptop_2285.ssisdemos
SQLSourceType	Direct input
SQLStatement	SELECT * FROM department
IsQueryStoredProcedure	False
BypassPrepare	True

9.Goto ResultSet TAB, click "Add" button and configure as below and click ok

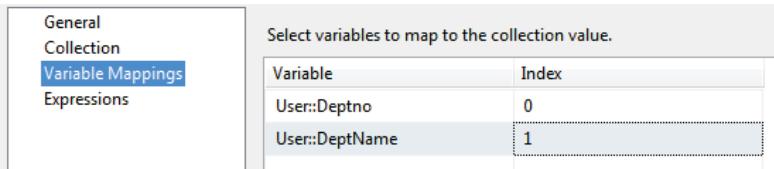
General	Result Name	Variable Name
Parameter Mapping	0	User::DeptRS
Result Set		
Expressions		

10.Drag a For Each Loop Container Task. Rename it “Iterate the Department Rows”

- e. Double click the task, goto collection tab, and set enumerator property to “ForEach Ado Enumerator”, set ado object source variable to User::DeptRS

General	Foreach Loop Editor
Collection	Enumerator
Variable Mappings	Foreach ADO Enumerator
Expressions	Expressions
Enumerator	
Specifies the enumerator type.	
Enumerator configuration	
ADO object source variable:	
User::DeptRS	

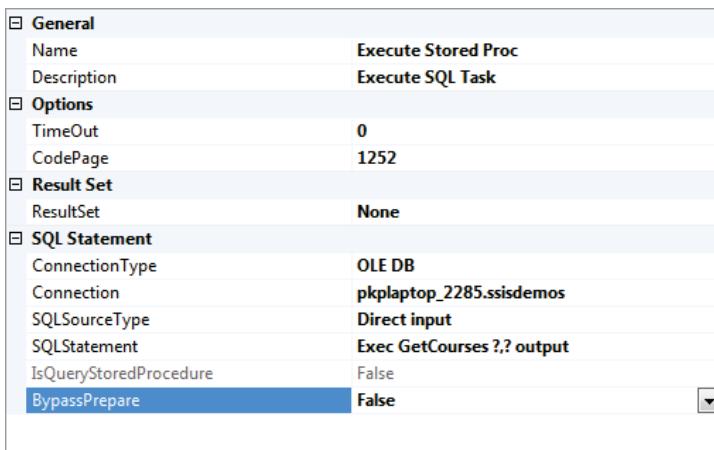
11.Goto Variable Mappings tab and configure as below and then click Ok



Variable	Index
User::Deptno	0
User::DeptName	1

12. Drag a Execute SQL Task inside the For Each Loop Container, Rename the task as “Execute Stored Proc”

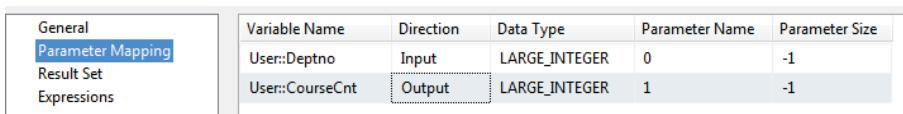
- f. Double click task, set the following properties:
 - i. ResultSet: None
 - ii. connection type: oledb
 - iii. connection: specify a connection to your sql server
 - iv. sql source type: direct input
 - v. SQL Statement: Exec GetCourses ?,? output
 - vi. ByPass Prepare: false



Name	Execute Stored Proc
Description	Execute SQL Task
TimeOut	0
CodePage	1252
ResultSet	None
SQL Statement	
ConnectionType	OLE DB
Connection	pkplaptop_2285.ssisdemos
SQLSourceType	Direct input
SQLStatement	Exec GetCourses ?,? output
IsQueryStoredProcedure	False
BypassPrepare	False

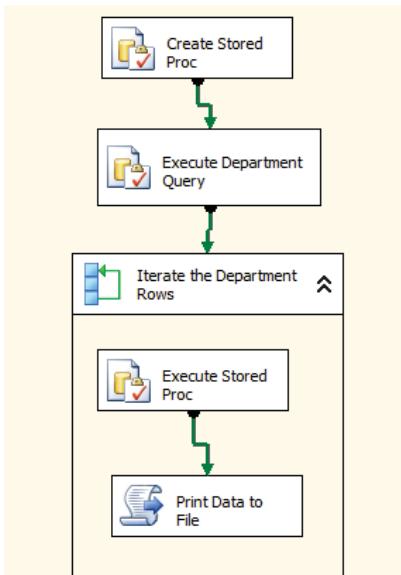
2. Goto Parameter Mapping Tab and configure it as below and click Ok

Note: Click Add button to add variable entries as parameters



Variable Name	Direction	Data Type	Parameter Name	Parameter Size
User::Deptno	Input	LARGE_INTEGER	0	-1
User::CourseCnt	Output	LARGE_INTEGER	1	-1

3. Drag a script task inside the For Each Loop container task and rename it as “Print Data to File”
4. Connect the various tasks that are dragged sequentially as shown below:



5. In the package designer right click the “Connection Managers” section at the bottom in free space and select “new file connection”
 - a. Set Usage Type: Create File and
 - b. Browse and specify filename as Results.txt in Demo3 folder created at the start of this lab. Click ok
6. Double the script task and in the ellipses for readonly variables property, select User::CourseCnt, User::DeptName, User::Deptno Variables.
7. Click Edit Script button
8. Define the following code in the main function

```

publicvoid Main()
{
    int? dno,coursecnt;
    string dname;

    string FileName = (string)Dts.Connections["Results.txt"].AcquireConnection(null);
    dno = (int)Dts.Variables["Deptno"].Value;
    if (Dts.Variables["CourseCnt"].Value == null)
        coursecnt = 0;
    else
        coursecnt = (int)Dts.Variables["CourseCnt"].Value;
    dname = (string)Dts.Variables["DeptName"].Value;
    using (StreamWriter sw = File.AppendText(FileName))
    {
  
```

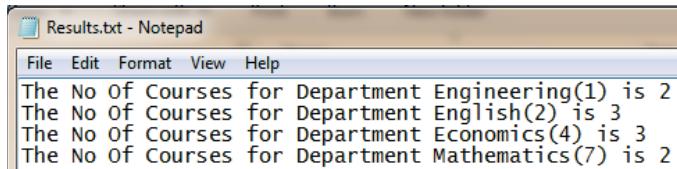
```

        sw.WriteLine("The No Of Courses for Department " + dname + "(" +
dno.ToString() +") is " + coursecnt.ToString());
        sw.Close();
    }
    Dts.TaskResult = (int)ScriptResults.Success;
}

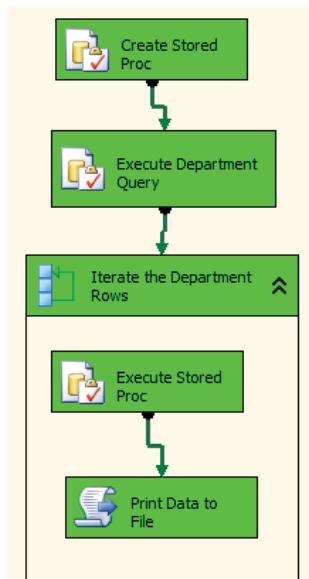
```

9. Build the script Ctrl+Shift+B. Once succeeds exit the script editor. Click ok on the script task dialog box to go to the package designer.

10. Run the package and open the Results.txt file as shown below and then stop the package



The No Of Courses for Department Engineering(1) is 2
The No Of Courses for Department English(2) is 3
The No Of Courses for Department Economics(4) is 3
The No Of Courses for Department Mathematics(7) is 2



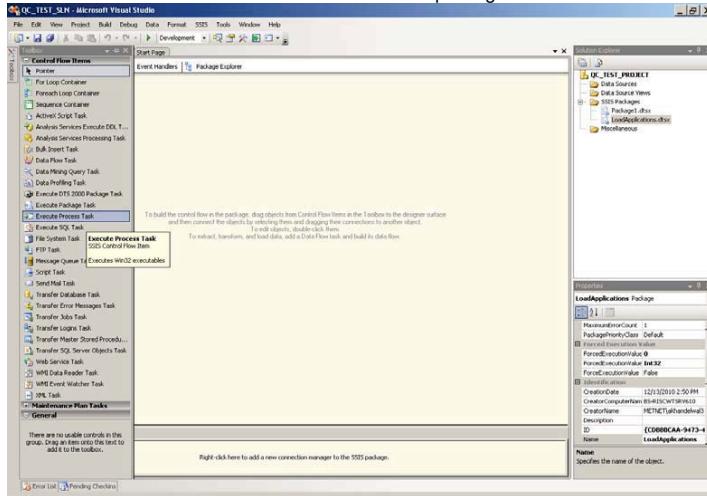
Open a different application based on day of week

Lab 15- Execute Process Task

Objective	The purpose of this Lab is to show an example of property expressions in a fringe operation, using the Execute Process task to open other applications.
Lab Setup	<ul style="list-style-type: none"> SSDT tool Create package.

1) Save and Close all other open packages and create a new package as shown below and then rename it:

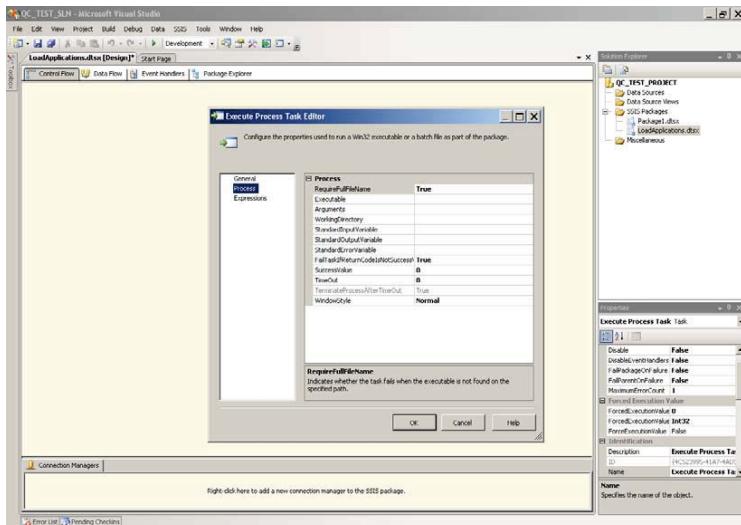
2) Add an Execute Process task to the control flow of this new package:



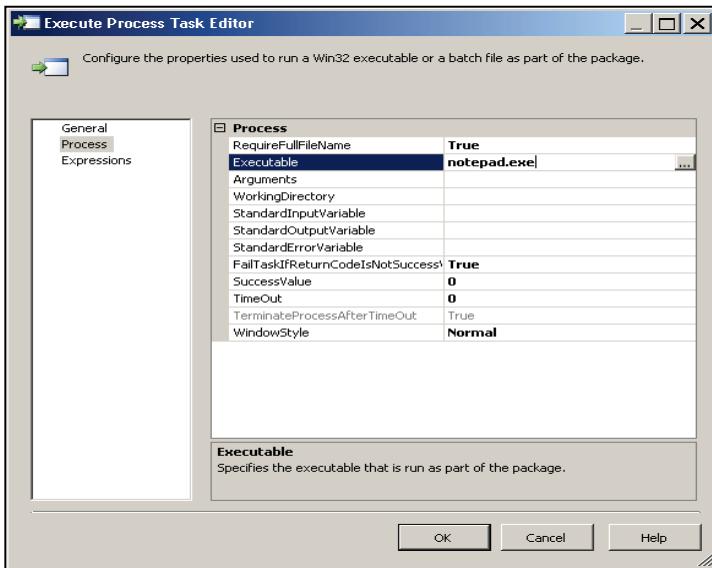
Note:

[Using property expressions we can configure the single task to open a different application based on the day of the week. Open either notepad.exe or mspaint.exe depending on day of week.]

3) Now double click on the Execute Process task to go to editor and click on the Process Page:

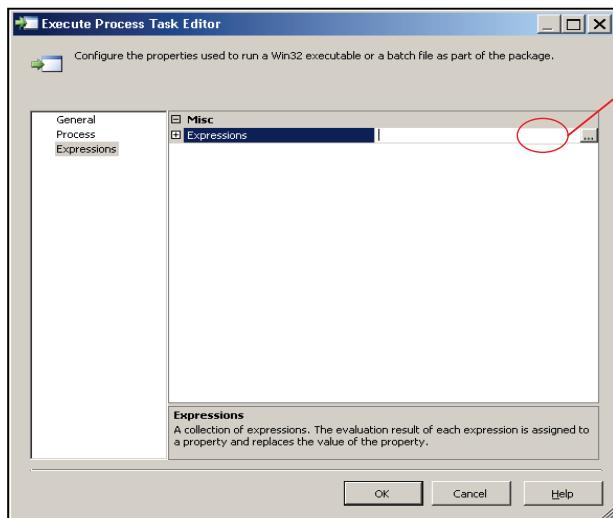


4)Now enter "notepad.exe" for the 'Executable' property.



5)Now click the “Expressions” page. In the right pane click in the empty row for “Expressions” and then press the ellipse button which will take you to the Property Expressions Editor.

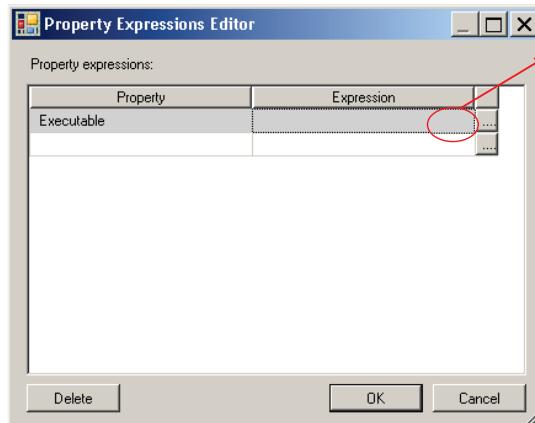
Press this
ellipse
button



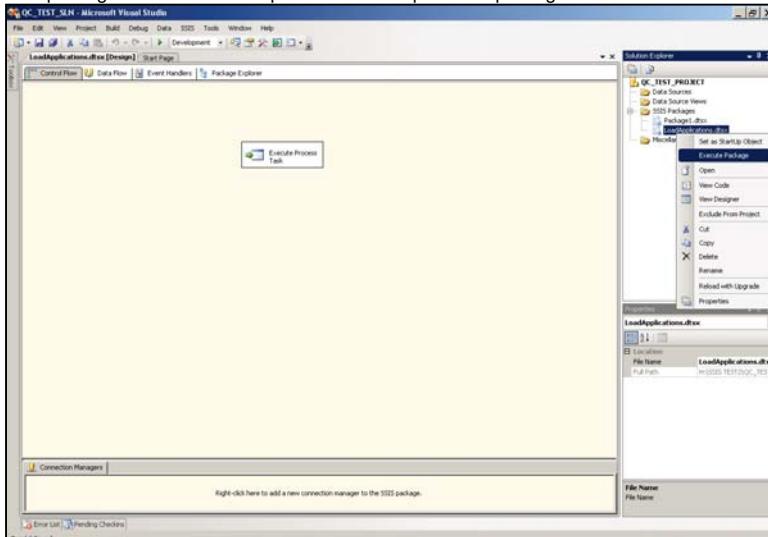
6)Now choose the "Executable" property and either copy/paste the following expression or press the other ellipse button to go into the expression builder and build this yourself and then evaluate the expression:

Expression:

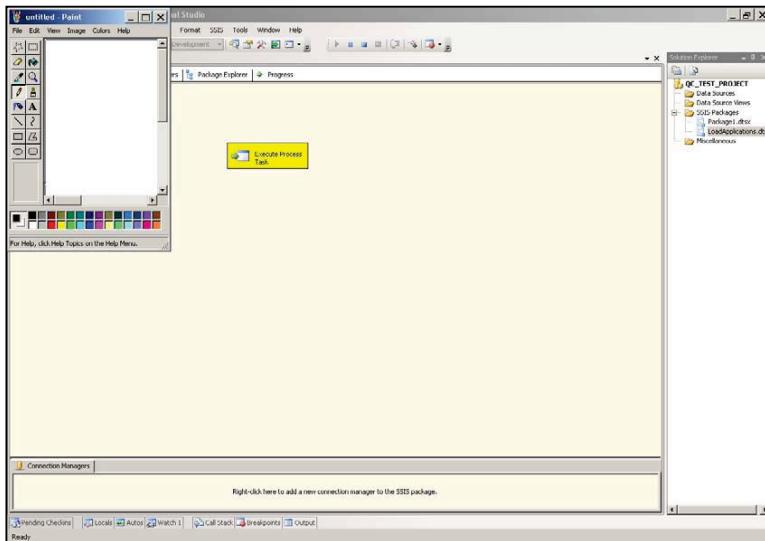
DATEPART("weekday", GETDATE()) ==5?"notepad.exe":"mspaint.exe"



7)Save the package. In the solution explorer select this particular package and select "Execute Package".

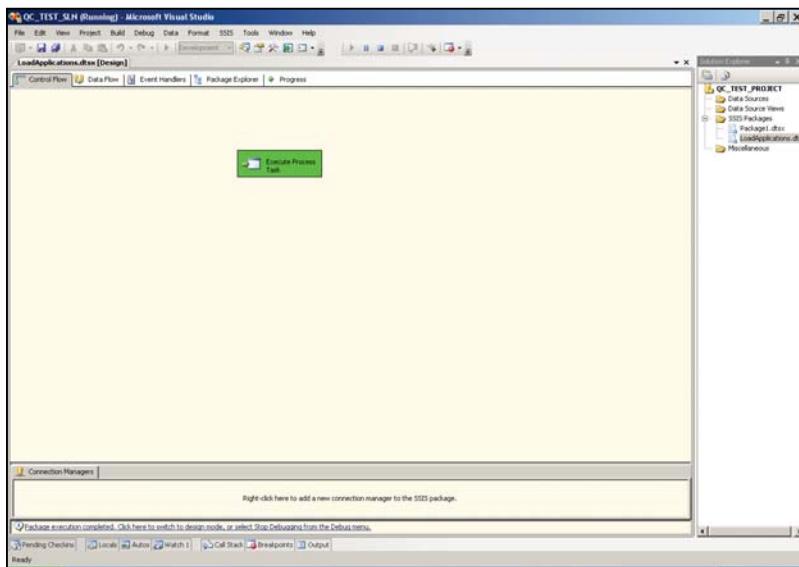


8)IF the current day is Thursday (5) then Windows Notepad will open, otherwise Paint will open as shown below:



Note: The Execute Process task is still yellow and the package is still considered 'running' until you close the application that opened, then the task will turn green.

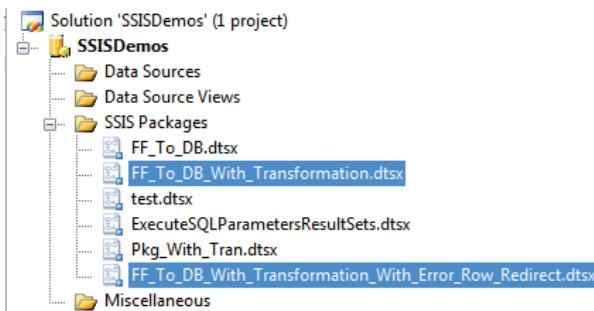
9) After closing the Paint application this task turned green.



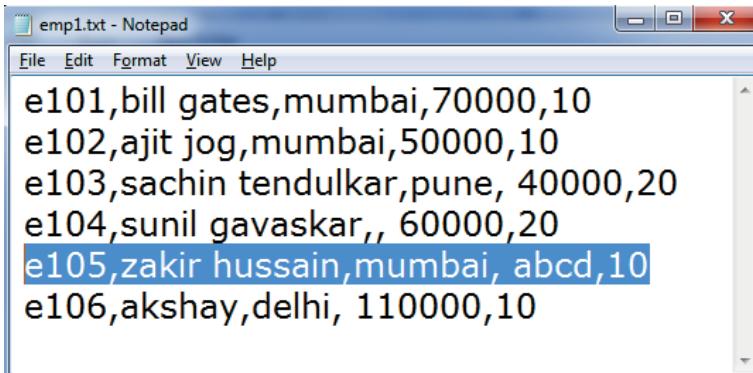
Lab 16- Using Error Redirect Row

Objective	To learn 1. How to use error redirect output to identify the rows having processing errors and prevent the entire ELT process from halting
Lab Setup	<ul style="list-style-type: none"> SSDT tool Login details for connecting to the database and Create package.

1)In Solution Explorer copy and paste the “FF_To_DB_With_Transformation.dtsx” and rename as “FF_To_DB_With_Transformation_With_Error_Row_Redirect.dtsx”. Click “Yes” to the dialog box that will come up to complete renaming.



2)Goto Demo2 folder created for Lab 2 open “emp1.txt” file and change the salary value of one the employee to character as shown and save changes

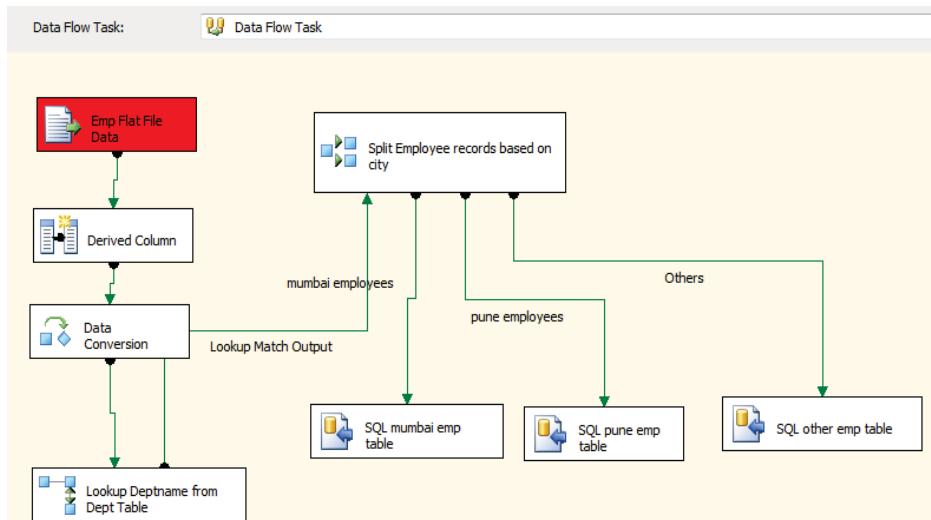


```
e101,bill gates,mumbai,70000,10
e102,ajit jog,mumbai,50000,10
e103,sachin tendulkar,pune, 40000,20
e104,sunil gavaskar,, 60000,20
e105,zakir hussain,mumbai, abcd,10
e106,akshay,delhi, 110000,10
```

3) Prior delete all the records from the tables

```
deletefrom mumbaiemp
deletefrom puneemp
deletefrom otheremp
```

4) Open “FF_To_DB_With_Transformation.dtsx” package and execute the package. There will be error in the package execution as shown:



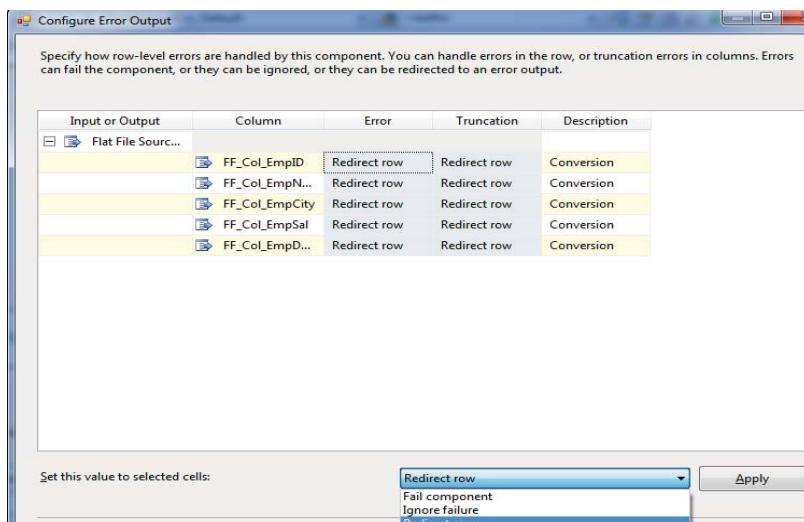
5) Stop the Package execution from VS Studio.

6) Check the SQL tables, no data would be transferred.

7) Open “FF_To_DB_With_Transformation_With_Error_Row_Redirect.dtsx” goto data flow tab, from toolbox drag “Flat File Destination” task.

8) Join the Red Error Output Connector of the “Emp Flat File Data” Flat File Source Task to it.

9) In the dialog box select all the error and truncation column entries. (Use: “shift” + click) select “redirect row” from dropdown below and click appl. Click ok

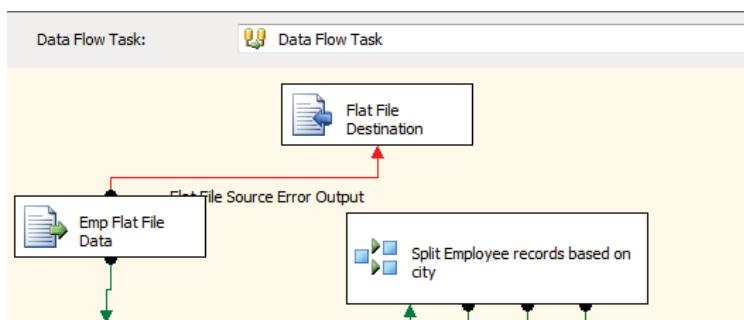


10) Double Click the “**Flat File Destination**” task click new “New” button to create a new Connection Manager. Select “Delimited” in the dialog box that pops. Name the connection manager as “FFCM_Error_Output_File”.

11) In filename specify the file as “emperr.txt” in the same directory Demo2

12) For eg: C:\Users\jogajitm\Desktop\Project 1\Folders\Demo2\emperr.txt

13) Click Ok.



14) Save Changes to package. Run it.

15) The package will succeed. Stop it. The error row will come into the emperr.txt file rest of the rows will be successfully transferred to the SQL tables.

Lab 17- Transaction support and Checkpoint

Objective	To learn <ol style="list-style-type: none"> How to use Transaction support available for atomicity How to use Checkpoint feature to resume the package from the point where it was abandoned
Lab Setup	<ul style="list-style-type: none"> SSDT tool Login details for connecting to the database and Create package.

- 1)Create Demo5 folder for this lab. Create "In", "Out" and "Checkpoint" 3 folders under it.
2)Connect to sql server and create 2 tables as below:

```

CREATE TABLE staginginvoices
(
    invno int PRIMARY KEY,
    invamt numeric(18, 0) NULL,
    invdate smalldatetime NULL,
)

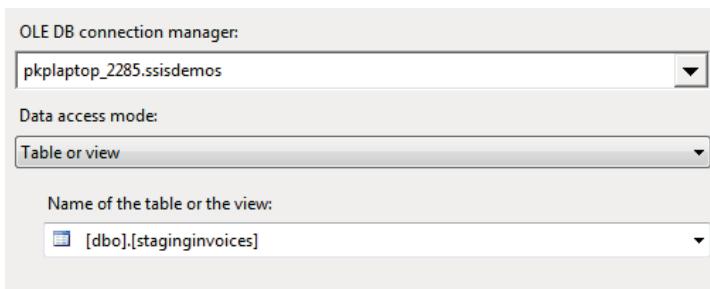
CREATE TABLE invoices
(
    invno int PRIMARY KEY,
    invamt numeric(18, 0) NULL,
    invdate smalldatetime NULL,
)

```

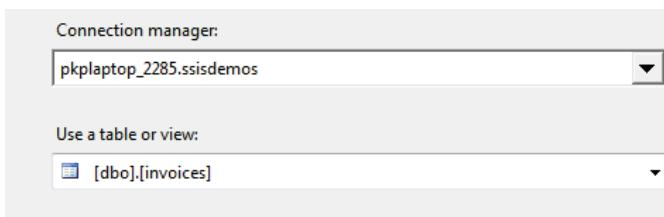
Add few records into staging invoices table for eg as below:

Results			Messages	
	invno	invamt	invdate	
1	1	10000	2012-01-01 00:00:00	
2	2	25000	2012-01-03 00:00:00	
3	3	40000	2012-01-03 00:00:00	

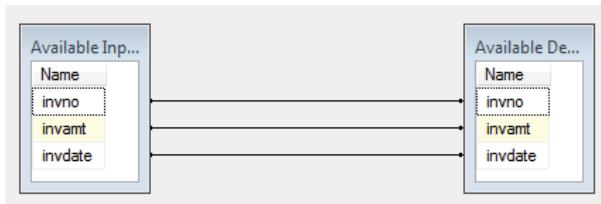
- 3)Add a new ssis package and name it as "**Pkg_with_Tran_and_CheckPoint_Demo.dtsx**".
4)In the control flow, drag a sequence container task. Rename it "Transfer Invoices to Table and File".
5)Drag a data flow task inside this sequence container. Rename data flow task to "Transfer Invoices to SQL Table".
6)double click the above data flow task.
7)Drag a oledb data source task from data flow sources section. Rename as "Staging Table" Double click and point it to the sql server database where staginginvoices table is created as shown below. Click ok



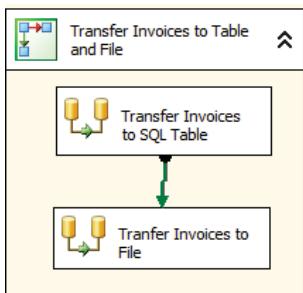
8) Drag a Sql server destination task. Rename it "Invoices Table". Connect the oledb datasource task to the destination task. Double click destination task and point it to invoices table created above.



9) Go to mappings tab and map the corresponding columns. Click ok



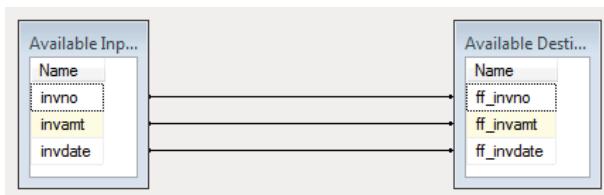
10) In control flow, drag another data flow task in the sequence container, below the earlier data flow task and connect them. Rename this task as "Transfer Invoices to File".



- 11) Double click above data flow task to invoke data flow designer,
- 12) Drag a oledb data source and point it to staginginvoices.
- 13) Drag a flat file destination, rename it "Invoices Flat File".
- 14) Connect the oledb source of step 12 to it. double click flat file destination and through a new flat file connection point it to a file in "In" folder of step 1.

For eg: C:\Users\jogajitm\Desktop\Project 1\Folders\Demo5\In\invoices.txt

- 15) Set the mappings as below:



- 16) Back to control flow, drag a Execute SQL Task outside sequence container, rename as "Empty Staging table". Double click it and configure it as below:

General	
Name	Empty Staging table
Description	Execute SQL Task
Options	
TimeOut	0
CodePage	1252
Result Set	
ResultSet	None
SQL Statement	
ConnectionType	OLE DB
Connection	pkplaptop_2285.ssisdemos
SQLSourceType	Direct input
SQLStatement	delete from staginginvoices
IsQueryStoredProcedure	False
BypassPrepare	True

Note: Connection: is the connection manager which points to staginginvoices sql server database.

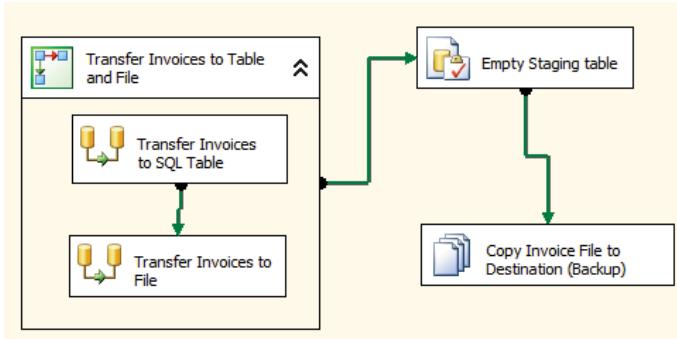
- 17) Drag a File System task on to control flow and rename as "Copy Invoice File to Destination (Backup)".
18) Double click the above task and configure as below:

Destination Connection	
IsDestinationPathVariable	False
DestinationConnection	invoices.txt
OverwriteDestination	False
General	
Name	Copy Invoice File to Destination (Backup)
Description	File System Task
Operation	
Operation	Copy file
Source Connection	
IsSourcePathVariable	False
SourceConnection	FF_Invoices_CM

Note:

- ➔ **Destination connection** is a flat file connection manager which points to some file in "Out" folder created in step 1.
For eg: C:\Users\jogajitm\Desktop\Project 1\Folders\Demo5\Out\invoices.txt
- ➔ **Source Connection** is the flat file manager created above in step 14 which points to the file in "In" folder
- ➔ The **operation property** is set to Copy File.

19) Connect the control flow tasks as shown below



Configure Transaction Support:

- 20) Select the sequence container task, and set the transaction option of the task to "required".
21) Set the transaction option of both the data flow tasks within sequence container to "supported"

Configure Checkpoint Support:

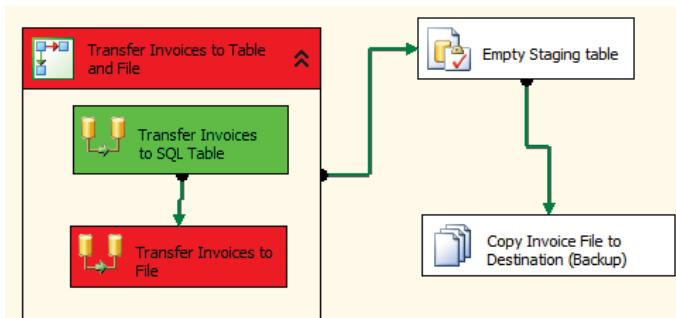
- 22) Select the control flow designer , by clicking it on free surface.
23) Set the following properties

Property Name	Value
Checkpoint File Name	To a file name in checkpoint folder created in step 1 For example something like: C:\Users\jogajitm\Desktop\Project 1\Folders\Demo5\CheckPoint\ChkPnt.txt
CheckPoint Usage	If Exists
Save Checkpoints	True

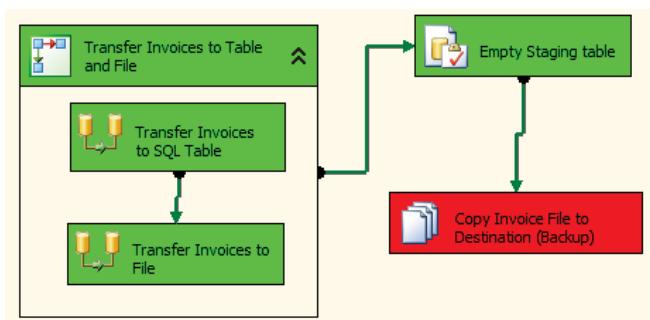
- 24) Make use that for all tasks in the control flow viz: sequence container, sub tasks of sequence container, execute sql task, file system task, the properties "FailPackageOnFailure" and "FailParentOnFailure" properties are true.

→ Now we ready to run the package and test the features

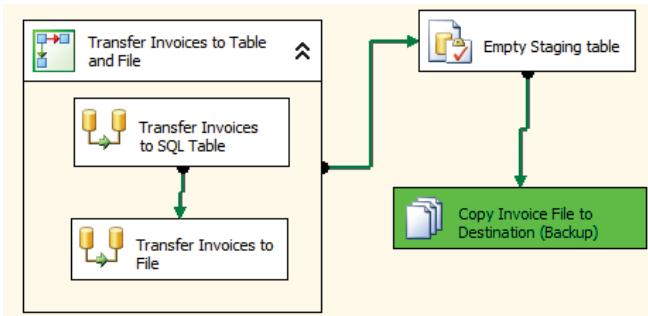
- 25) Before we run delete the "In" folder for time being or rename it to something else.
26) Run the Package, the package will fail in the sequence container because "In" folder is not found while transferring invoices details to flat file as below:



- 27) Go to SSMS and check that the work of first data flow task is rolled back. The data is not successfully transferred to Invoices online table from staging table. Stop the package
28) Now recreate the "In" folder, but now delete the "Out" folder from Demo5 folder for the time being.
29) Now run the package, the package fails at the File System Task "Copy Invoice File to Destination (Backup)" but the earlier tasks are completed as below:



- 30) Stop the package, and check that:
→ Rows have been copied to invoices table from staging table.
→ The flat file is created in "In" folder
→ A checkpoint has been created in checkpoint folder
31) Stop the package and restore or recreate the "Out" folder and rerun the package.



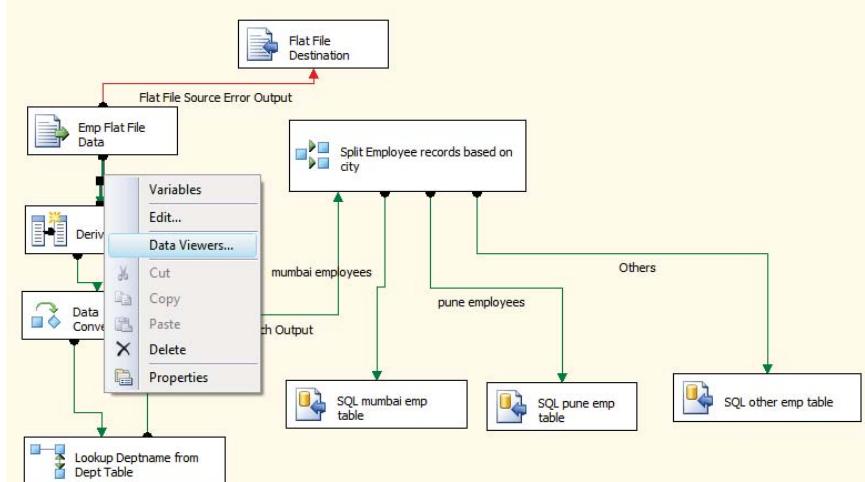
- 32)The package will run directly run the last file system task and the checkpoint file will be automatically removed from its location and the file will be copied to the "Out" folder.
33)Stop the package.

Lab 18- Using Data Viewer

Objective	To learn 1. How to use data viewer
Lab Setup	<ul style="list-style-type: none"> SSDT tool Create package.

1)Open the Package “FF_To_DB_With_Transformation_With_Error_Row_Redirect.dtsx” created in Lab 4

2)Right Click the data flow path between the “Emp Flat File Data” and “Derived Column” data flow tasks.



3)Click Data Viewers

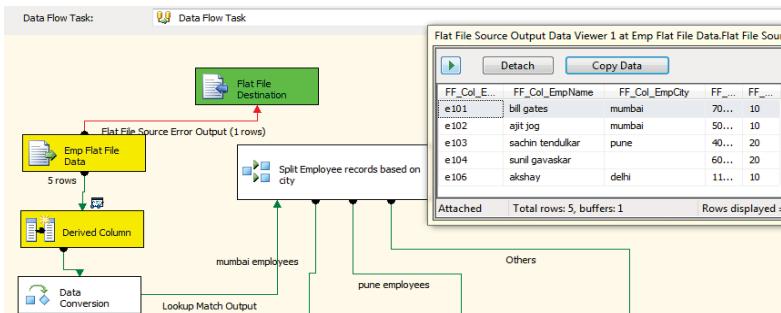
4)In the Data Viewer dialog box click “Add” button.

5)Select Grid from options. Click ok. Again click Ok.

6)delete the data from mumbaiemp, puneemp and otheremp tables.

7)Execute the package.

8)A data viewer window will automatically popup and will show the records retrieved from the flat data file. The package execution will also pause for a moment.



9)Resume the package execution click  button on data viewer window and complete the execution.

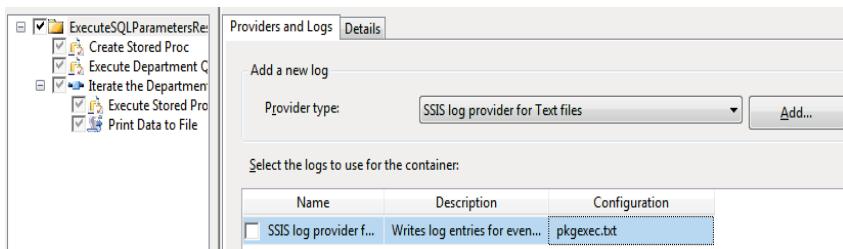
Lab 19- Using Package Logging

Objective	To learn <ol style="list-style-type: none"> How to enable package logging
Lab Setup	<ul style="list-style-type: none"> SSDT tool Create project.

1)Open the Package "ExecuteSQLParametersResultSets.dtsx" of Lab 3.

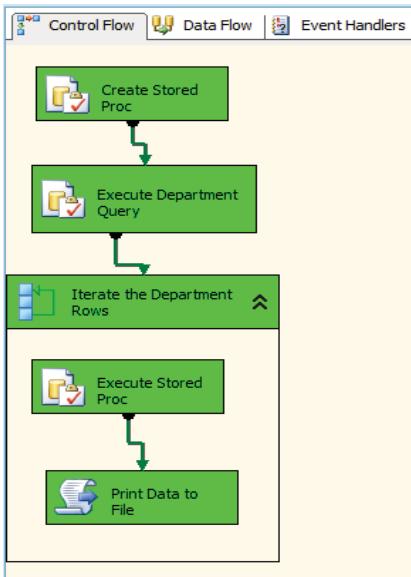
2)To enable logging for this package go to SSIS => Logging Menu option.

- In the logging dialog box, select the root tree node of the package in left pane.
- In right pane, select "SSIS log provider for text file" and click add. A row will be added.
- Go to configuration column of the row, Click new connection.
- Select Create File, and give a file path, where you want your package execution log file to be created.
- Click ok



- a. Now go to details tab, select "onpreexecuted", "onpostexecuted" and "ontaskfailed" events.
- b. Click ok.

3)Execute the package.



4)Stop the execution on the package.

5)Open the package log file. The package file contents will look as below:

```

pkgenec.hvt - Notepad
File Edit Format View Help
#Fields: event,computer,operator,source,sourced,executionid,starttime,endtime,datacode,databytes,message
PackageStart,PKPLAPTOP_2285,IGATECORP\jogajitm,ExecuteSQLParametersResultSets,{941EE059-313A-4604-9982-5D48A71AF57F},{D3FEF66-6C59-4520-8E4F-5AD85A5C1643},27-01-2012 11:11:43,27-01-2012 11:11:43,0,0x,Beginning of package execution.

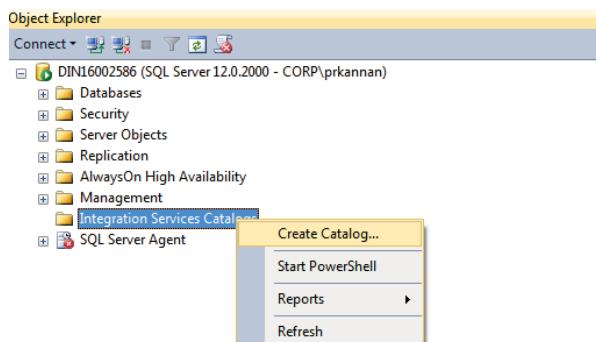
OnPreExecute,PKPLAPTOP_2285,IGATECORP\jogajitm,ExecuteSQLParametersResultSets,{941EE059-313A-4604-9982-5D48A71AF57F},{D3FEF66-6C59-4520-8E4F-5AD85A5C1643},27-01-2012 11:11:43,27-01-2012 11:11:43,0,0x,(null)
OnPreExecute,PKPLAPTOP_2285,IGATECORP\jogajitm>Create Stored Proc,{25BC5D01-0292-49C3-B010-8397BE6553D9},{D3FEF66-6C59-4520-8E4F-5AD85A5C1643},27-01-2012 11:11:43,27-01-2012 11:11:43,0,0x,(null)
OnPostExecute,PKPLAPTOP_2285,IGATECORP\jogajitm,Create Stored Proc,{25BC5D01-0292-49C3-B010-8397BE6553D9},{D3FEF66-6C59-4520-8E4F-5AD85A5C1643},27-01-2012 11:11:43,27-01-2012 11:11:43,0,0x,(null)
OnPostExecute,PKPLAPTOP_2285,IGATECORP\jogajitm,Execute Department Query,{6F7E51B9-9861-439D-A6EE-015498388215},{D3FEF66-6C59-4520-8E4F-5AD85A5C1643},27-01-2012 11:11:43,27-01-2012 11:11:43,0,0x,(null)
OnPostExecute,PKPLAPTOP_2285,IGATECORP\jogajitm,Execute Department Query,{6F7E51B9-9861-439D-A6EE-015498388215},{D3FEF66-6C59-4520-8E4F-5AD85A5C1643},27-01-2012 11:11:43,27-01-2012 11:11:43,0,0x,(null)
OnPostExecute,PKPLAPTOP_2285,IGATECORP\jogajitm,iterate the Department Rows,{7857942A-D374-4E89-8BD8-57E83FA3715F},{D3FEF66-6C59-4520-8E4F-5AD85A5C1643},27-01-2012 11:11:43,27-01-2012 11:11:43,0,0x,(null)
OnPostExecute,PKPLAPTOP_2285,IGATECORP\jogajitm,execute Stored Proc,{38EB36EA-461F-433E-83D5-BAB39F28FECC},{D3FEF66-6C59-4520-8E4F-5AD85A5C1643},27-01-2012 11:11:43,27-01-2012 11:11:43,0,0x,(null)
OnPostExecute,PKPLAPTOP_2285,IGATECORP\jogajitm,Execute Stored Proc,{38EB36EA-461F-433E-83D5-BAB39F28FECC},{D3FEF66-6C59-4520-8E4F-5AD85A5C1643},27-01-2012 11:11:43,27-01-2012 11:11:43,0,0x,(null)
  
```

Lab20-Project deployment

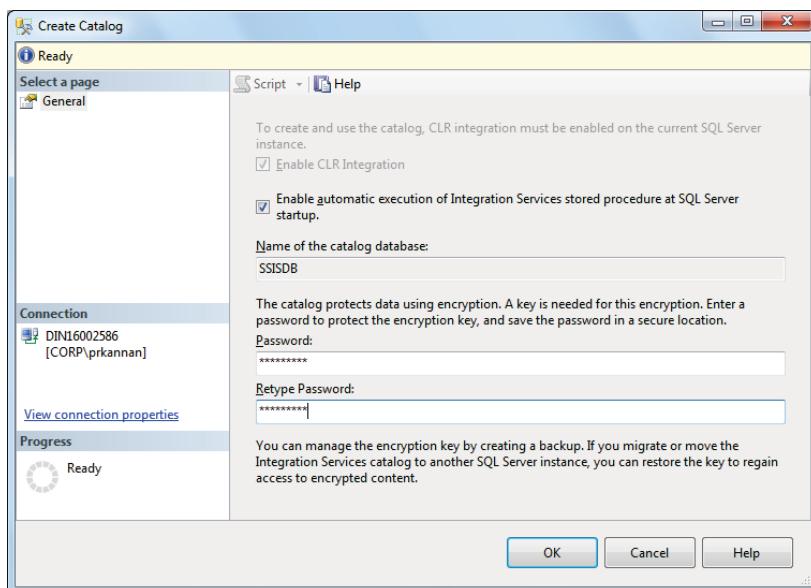
Objective	To learn 1.How to deploy a project.
Lab Setup	<ul style="list-style-type: none"> • SSDT tool • Login details for connecting to the database

IN SSMS:

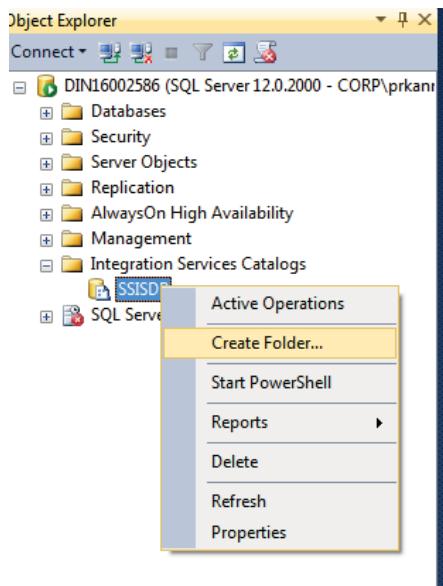
- 1)Before we can deploy any projects to a SQL Server instance, we first need to create the Integration Services Catalog. Think of this catalog as the container for all deployed projects, their settings and historical versions.
- 2)There is only one Catalog per SQL Server instance; it is represented by a separate SQL Server database called SSISDB, which contains the deployment's versioning, settings, and even statistical performance data.
- 3) In order to create the catalog, we need to connect to the SQL Server instance and right-click on the 'Integration Services Catalogs' and click 'Create Catalog'.



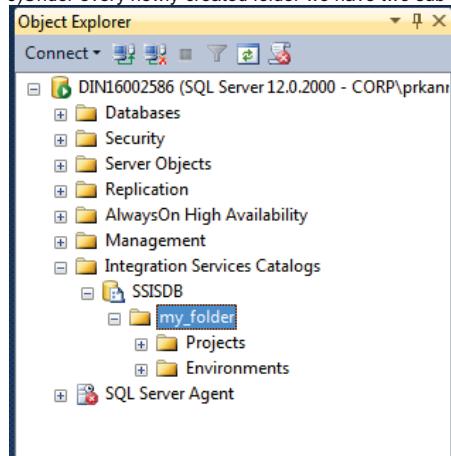
- 4) In the next screen we are asked for the encryption password and whether we would like to run the **Catalog_Startup** procedure every time the SQL Server instance starts (the **Catalog.Startup** stored procedure fixes the status of the packages in the catalog if there were packages running when the SQL Server instance went down):



5) After we have created the Catalog, we can create a new folder under it, which will contain our first project. To create the folder, simply right-click on the SSISDB catalog and click 'Create Folder'.

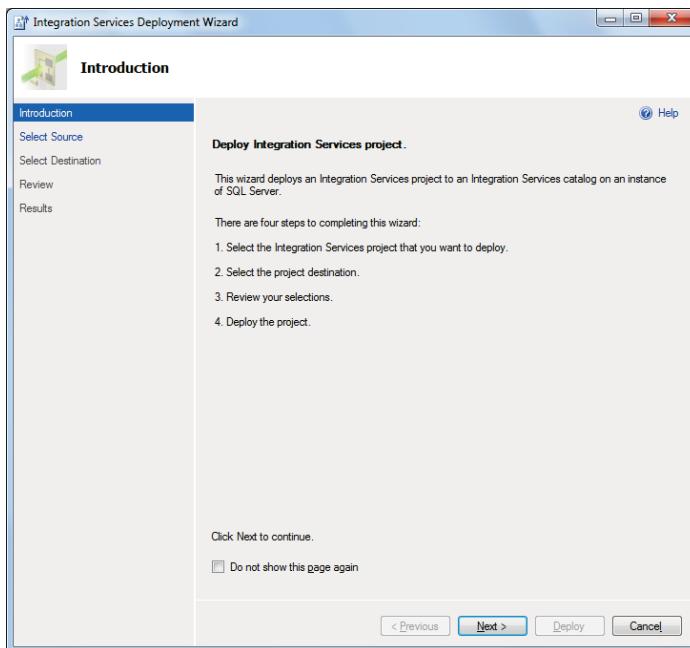


6) Under every newly-created folder we have two sub-folders: 'Projects' and 'Environments'.

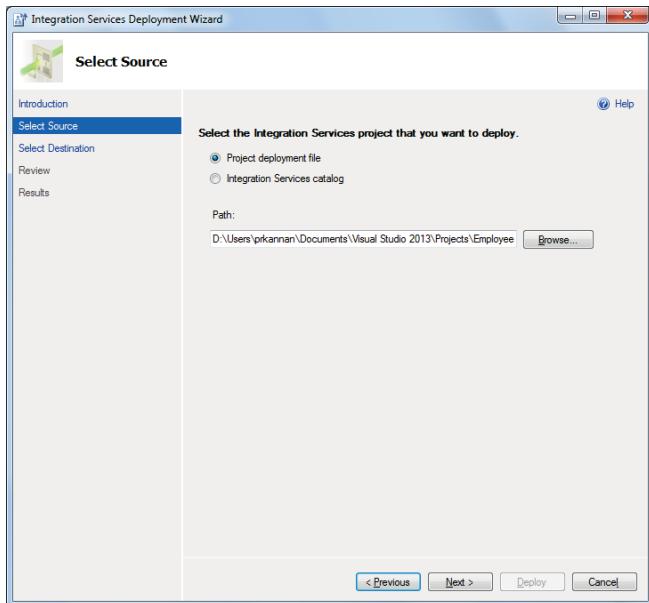


IN SSDT:

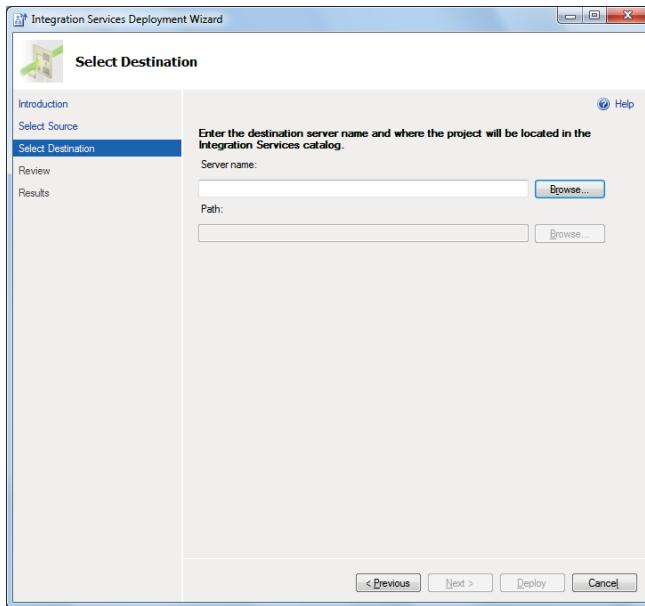
7) Now let's deploy the project to our SQL Server instance. By right-clicking on the project solution in SSDT and clicking Deploy, we get to the following wizard:



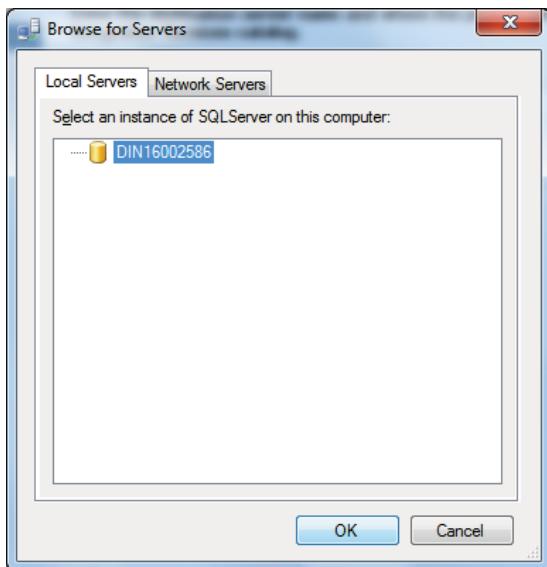
8)Select source page will appears like this:



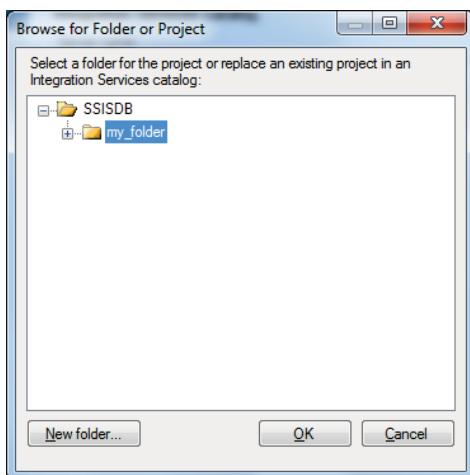
9)Select Destination server looks like the below screenshot:



10)Click browse button to choose your destination server name like below screenshot and click ok.



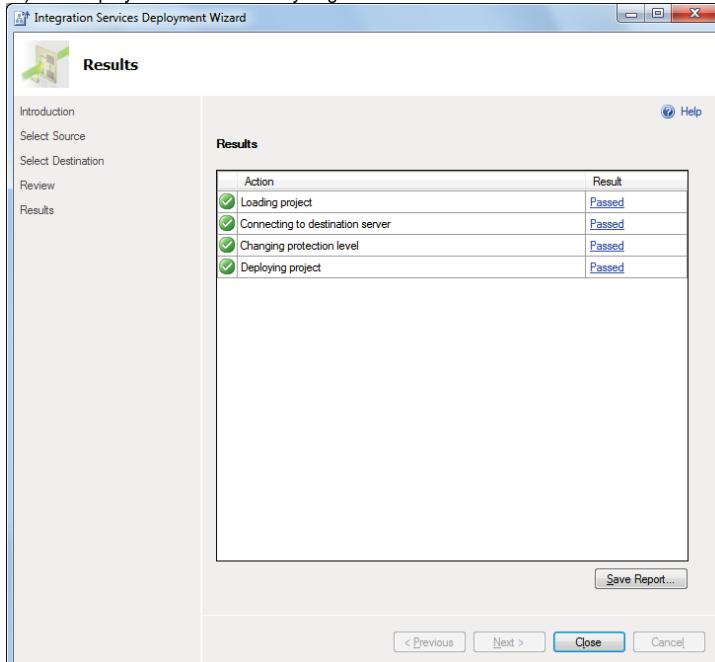
- 11)Now browse path where the project should be deployed as given below.
12)You will get the below screen when you click the browse button.



13)Choose the folder which you have created under Integration services catalog and click ok and give next.

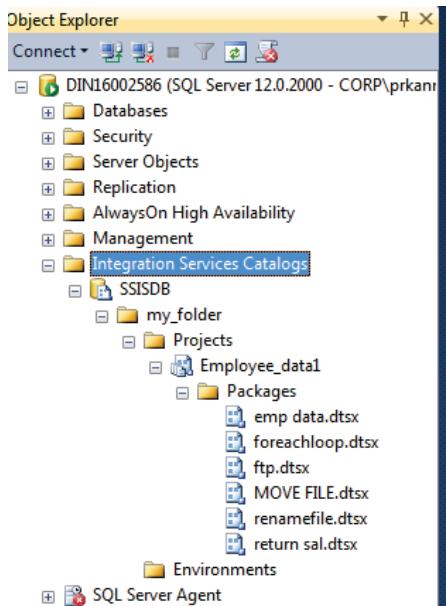
14)Check once for your source and destination and deploy.

15)If the deployment is successful you get like the below screenshot.



16)If you need, you can save the report.

17)The deployed project now looks like this:

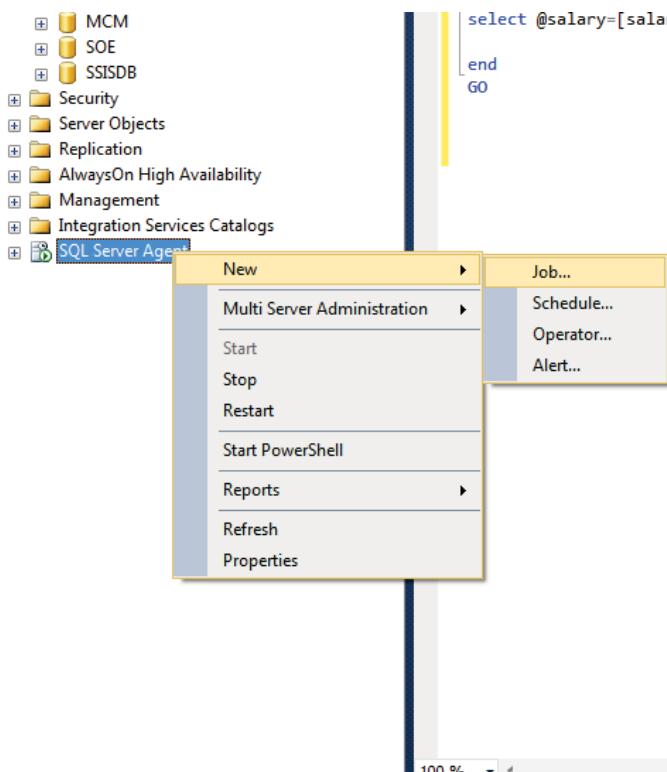


18)From this point on, we can execute the deployed package by scheduling an SQL Server Agent Job and using the package in it.

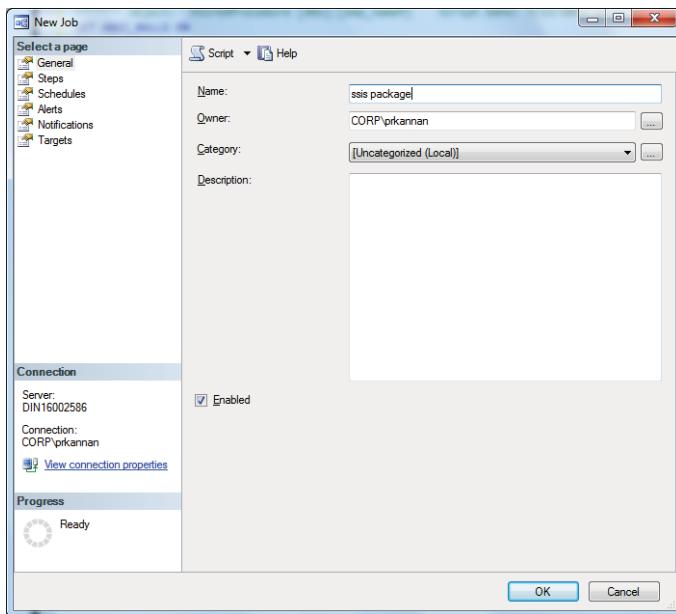
Lab21-Package Execution using SQL Server Agent

Objective	Package execution from SQL Server Agent
Lab Setup	Login details for connecting to the database

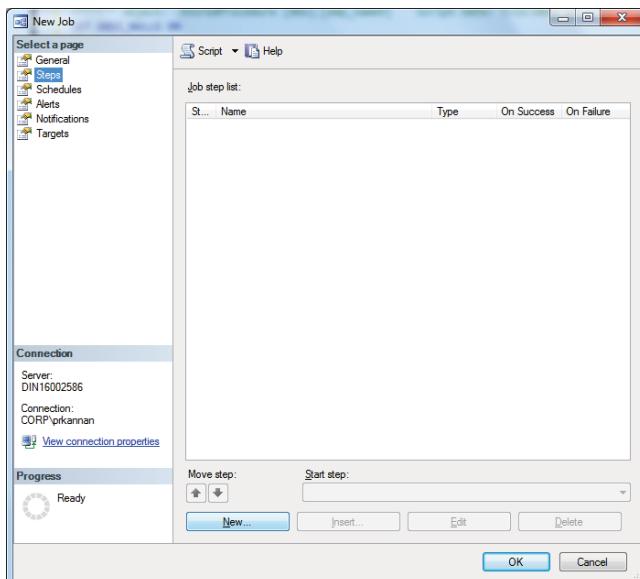
1)To execute a package from SSMS , right click on the SQL Server Agent which will be under your server as per below.



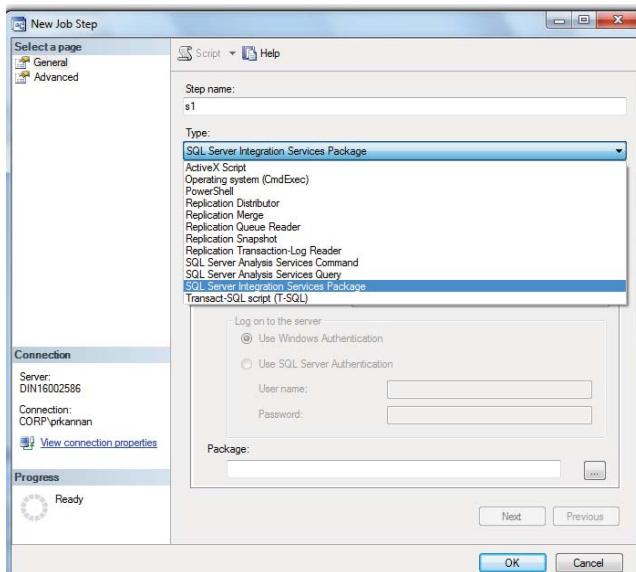
2)Click on new job you will get the below screen. Give name for your job.



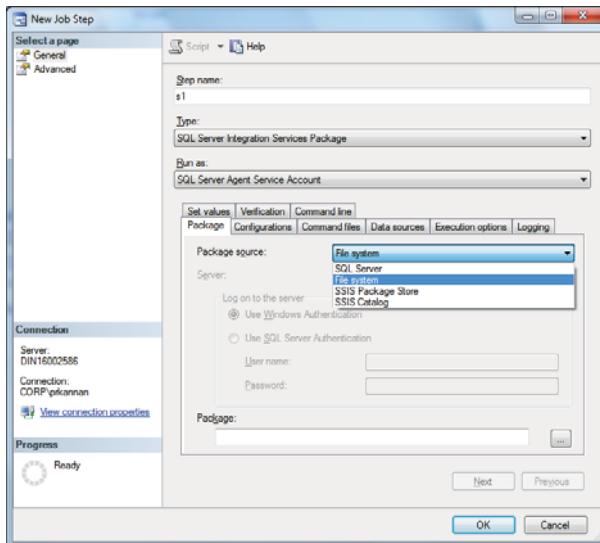
3) Click on the "Steps" under Select a page. You will get a screen like below where you should choose the **New...** option.



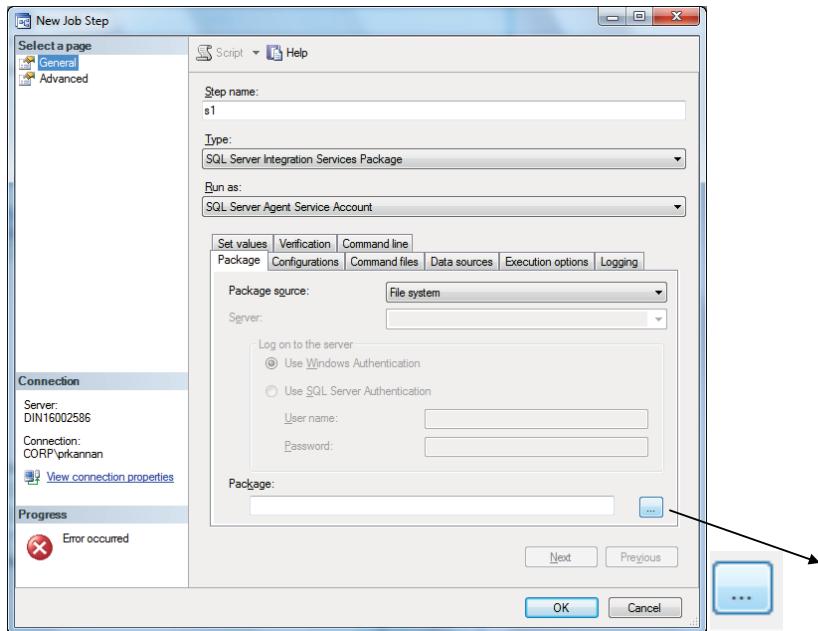
- 4) After selecting new , you will be prompted to this page. Give a name to your step and choose SQL Server Integration Services under type.

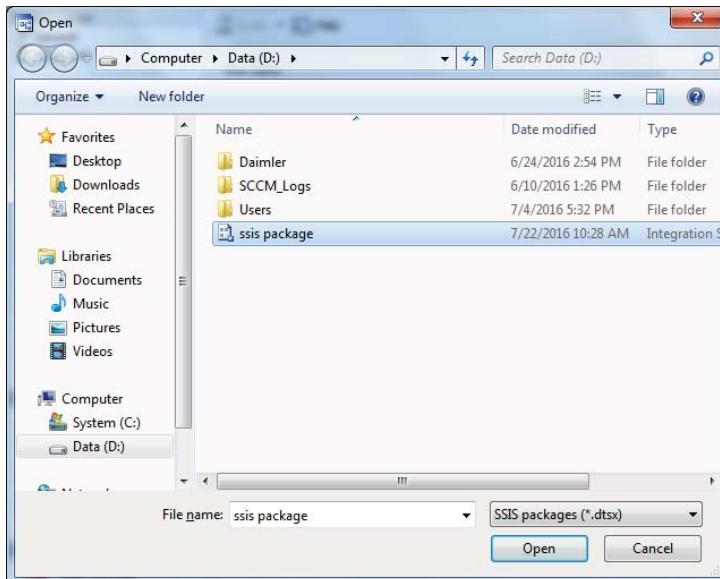


5)Select the package source as File System.

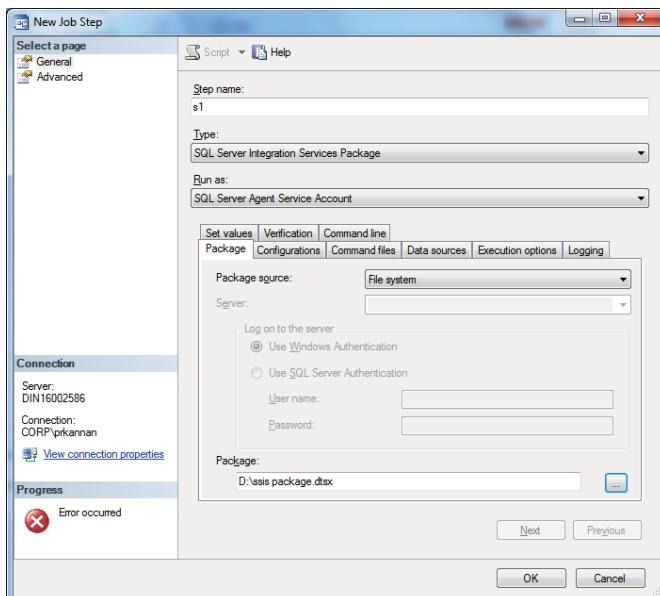


6)Select the package which you want to execute using the browse button as per below.



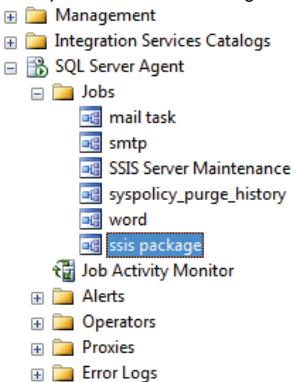


7)After selecting the package click open.

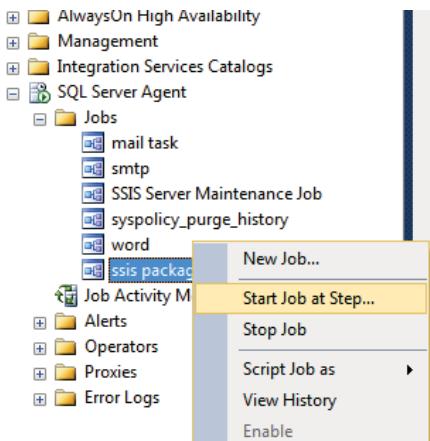


8) Then click ok in the new job step page and new job page.

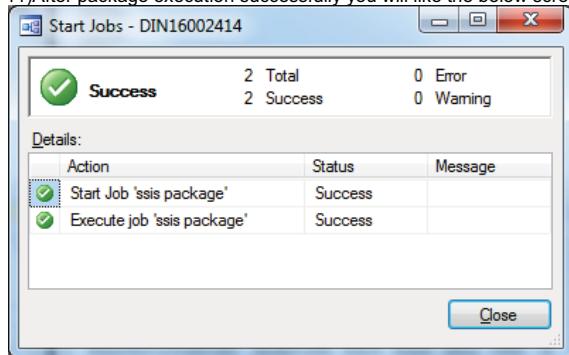
9) Now expand the SQL Server Agent and now you can see your job under server agent.



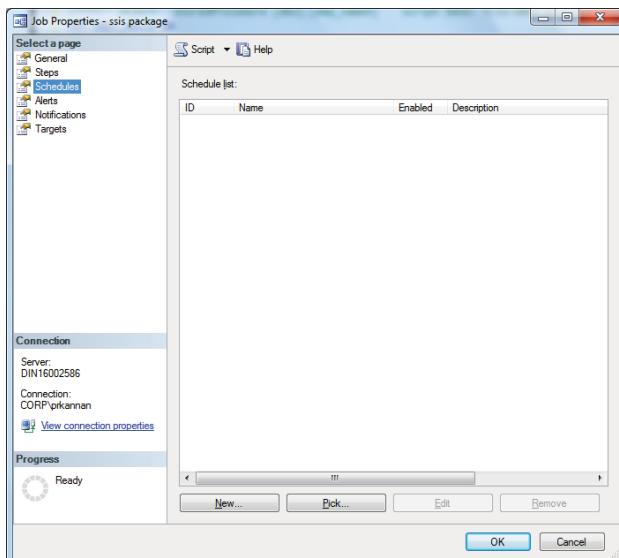
10) Now right click on the package on click Start job at Step.



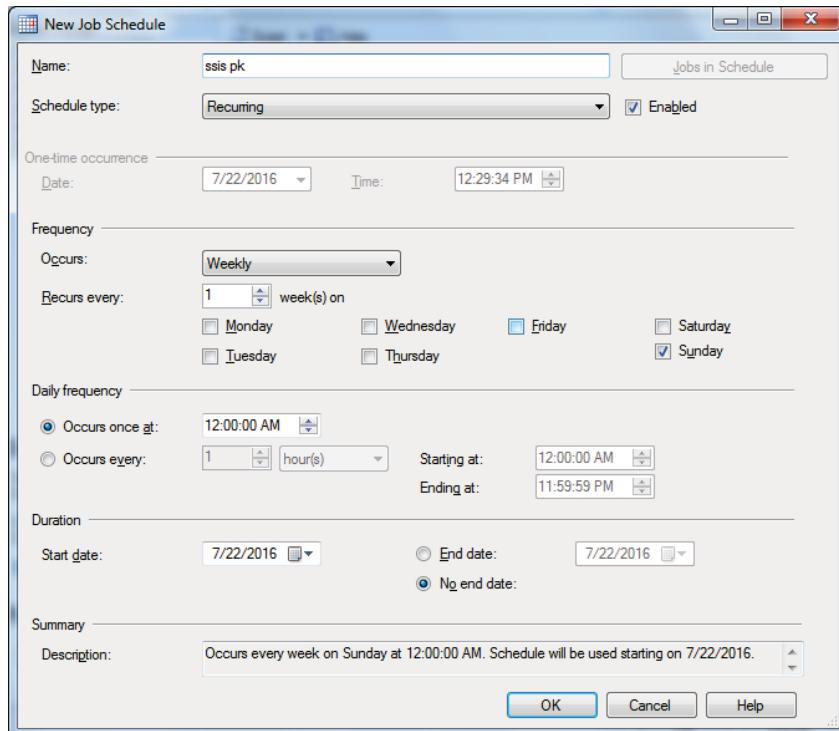
11) After package execution successfully you will like the below screenshot.



12) After executing your package by sql server agent you can schedule your package to run daily on at a frequency you wish. To schedule the job ,double click the package under sql server agent you will get the below screen.



13) Click new at present at the bottom of the screen. You will get the below screen.



14)Here you can specify the schedule type, frequency and daily frequency for executing your job.