

# Table of Contents

## Overview

[What's New in Integration Services in SQL Server 2016](#)

[What's New in Integration Services in SQL Server 2017](#)

[Integration Services Features Supported by the Editions of SQL Server](#)

[Integration Services Backward Compatibility](#)

## Install or upgrade

## Development and management Tools

## Projects and solutions

## User interface

[SSIS Designer](#)

[Advanced Editor](#)

[Group or Ungroup Components](#)

[Use Annotations in Packages](#)

[SSIS Toolbox](#)

[General Page of Integration Services Designers Options](#)

## Packages

[Create Packages in SQL Server Data Tools](#)

[Add Copy of Existing Package](#)

[Set Package Properties](#)

[View Package Objects](#)

[Copy a Package in SQL Server Data Tools](#)

[Copy Package Objects](#)

[Save Packages](#)

[Reuse Control Flow across Packages by Using Control Flow Package Parts](#)

[Reuse of Package Objects](#)

[Delete Packages](#)

[dtutil Utility](#)

[Package Upgrade Wizard F1 Help](#)

[Package and Project Parameters](#)

[Connections](#)

[Control flow](#)

[Data flow](#)

[Variables](#)

[Variables Window](#)

[System Variables](#)

[Expressions](#)

[Event Handlers](#)

[Queries](#)

[Transactions](#)

[Catalog and Server](#)

[SSIS Catalog](#)

[Integration Services \(SSIS\) Catalog Transact-SQL Reference](#)

[View the List of Packages on the Integration Services Server](#)

[Project Properties Dialog Box](#)

[Project Versions Dialog Box](#)

[Browse All Principals Dialog Box](#)

[Package Properties Dialog Box](#)

[Validate Dialog Box](#)

[Configure Dialog Box](#)

[Set Parameter Value Dialog Box](#)

[Folder Properties Dialog Box](#)

[Scale Out](#)

[Service \(legacy\)](#)

[Security](#)

[Performance](#)

[Troubleshooting](#)

[System views](#)

[System stored procedures](#)

[System function - dm\\_execution\\_performance\\_counters](#)

[Errors and Events Reference](#)

[Integration Services Error and Message Reference](#)

## Scripting and programming

[Integration Services Programming Overview](#)

[Understanding Synchronous and Asynchronous Transformations](#)

[Working with Connection Managers Programmatically](#)

[Extend packages with scripting](#)

[Extend packages with custom objects](#)

[Build packages programmatically](#)

[Run and manage packages programmatically](#)

## Integration Services Language Reference

[Azure Feature Pack for Integration Services \(SSIS\)](#)

[Hadoop and HDFS Support in Integration Services \(SSIS\)](#)

[Microsoft Connectors for Oracle and Teradata by Attunity](#)

[Import and export data](#)

[Change data capture](#)

## Microsoft Connector for SAP BW

[Installing the Microsoft Connector for SAP BW](#)

[Microsoft Connector for SAP BW Components](#)

[Microsoft Connector for SAP BW F1 Help](#)

[Certification by SAP](#)

## Tutorials

[Create a Package](#)

[Lesson 1: Create a Project and Basic Package](#)

[Lesson 2: Add Looping](#)

[Lesson 3: Add Logging](#)

[Lesson 4: Add Error Flow Redirection](#)

[Lesson 5: Add Package Configurations for the Package Deployment Model](#)

[Lesson 6: Using Parameters with the Project Deployment Model](#)

## Deploy Packages

[Lesson 1: Preparing to Create the Deployment Bundle](#)

[Lesson 2: Create the Deployment Bundle](#)

[Lesson 3: Install Packages](#)

# SQL Server Integration Services

5/11/2017 • 1 min to read • [Edit Online](#)

For content related to previous versions of SQL Server, see [SQL Server Integration Services](#).

Microsoft Integration Services is a platform for building enterprise-level data integration and data transformations solutions. You use Integration Services to solve complex business problems by copying or downloading files, sending e-mail messages in response to events, updating data warehouses, cleaning and mining data, and managing SQL Server objects and data. The packages can work alone or in concert with other packages to address complex business needs. Integration Services can extract and transform data from a wide variety of sources such as XML data files, flat files, and relational data sources, and then load the data into one or more destinations.

Integration Services includes a rich set of built-in tasks and transformations; tools for constructing packages; and the Integration Services service for running and managing packages. You can use the graphical Integration Services tools to create solutions without writing a single line of code; or you can program the extensive Integration Services object model to create packages programmatically and code custom tasks and other package objects.

# What's New in Integration Services in SQL Server 2016

4/19/2017 • 16 min to read • [Edit Online](#)



**Need help?** [MSDN Forum](#), [Stackoverflow](#), [Connect](#)

This topic describes the features that have been added or updated in SQL Server 2017 Integration Services.

## Improvements grouped by category

### • **Manageability**

- Better deployment
  - [SSISDB Upgrade Wizard](#)
  - [Support for Always On in the SSIS Catalog](#)
  - [Incremental package deployment](#)
  - [Support for Always Encrypted in the SSIS Catalog](#)
- Better debugging
  - [New ssis\\_logreader database-level role in the SSIS catalog](#)
  - [New RuntimeLineage logging level in the SSIS catalog](#)
  - [New custom logging level in the SSIS catalog](#)
  - [Column names for errors in the data flow](#)
  - [Expanded support for error column names](#)
  - [Support for server-wide default logging level](#)
  - [New IDTSComponentMetaData130 interface in the API](#)
- Better package management
  - [Improved experience for project upgrade](#)
  - [AutoAdjustBufferSize property automatically calculates buffer size for data flow](#)
  - [Reusable control flow templates](#)
  - [New templates renamed as parts](#)

### • **Connectivity**

- Expanded connectivity on premises
  - [Support for OData v4 data sources](#)
  - [Explicit support for Excel 2013 data sources](#)
  - [Support for the Hadoop file system \(HDFS\)](#)
  - [Expanded support for Hadoop and HDFS](#)

- HDFS File Destination now supports ORC file format
- ODBC components updated for SQL Server 2016
- Explicit support for Excel 2016 data sources
- Connector for SAP BW for SQL Server 2016 released
- Connectors v4.0 for Oracle and Teradata released
- Connectors for Analytics Platform System (PDW) Appliance Update 5 released
- Expanded connectivity to the cloud
  - Azure Storage connectors and Hive and Pig tasks for HDInsight - [Azure Feature Pack for SSIS released for SQL Server 2016](#)
  - [Support for Microsoft Dynamics online resources released in Service Pack 1](#)
  - [Support for Azure Data Lake Store released](#)
  - [Support for Azure SQL Data Warehouse released](#)
- **Usability and productivity**
  - Better install experience
    - [Upgrade blocked when SSISDB belongs to an Availability Group](#)
  - Better design experience
    - [SSIS Designer creates and maintains packages for SQL Server 2016, 2014, or 2012](#)
    - Multiple designer improvements and bug fixes.
  - Better management experience in SQL Server Management Studio
    - [Improved performance for SSIS Catalog views](#)
  - Other enhancements
    - [Balanced Data Distributor transformation is now part of SSIS](#)
    - [Data Feed Publishing Components are now part of SSIS](#)
    - [Support for Azure Blob Storage in the SQL Server Import and Export Wizard](#)
    - [Change Data Capture Designer and Service for Oracle for Microsoft SQL Server 2016 released](#)
    - [CDC components updated for SQL Server 2016](#)
    - [Analysis Services Execute DDL Task updated](#)
    - [Analysis Services tasks support tabular models](#)
    - [Support for Built-in R Services](#)
    - [Rich XML validation output in the XML Task](#)

## Manageability

### Better deployment

#### SSISDB Upgrade Wizard

Run the SSISDB Upgrade Wizard to upgrade the SSIS Catalog database, SSISDB, when the database is older than the current version of the SQL Server instance. This occurs when one of the following conditions is true.

- You restored the database from an older version of SQL Server.
- You did not remove the database from an Always On Availability Group before upgrading the SQL Server instance. This prevents the automatic upgrade of the database. For more info, see [Upgrading SSISDB in an availability group](#).

For more info, see [SSIS Catalog \(SSISDB\)](#).

#### Support for Always On in the SSIS Catalog

The Always On Availability Groups feature is a high-availability and disaster-recovery solution that provides an enterprise-level alternative to database mirroring. An availability group supports a failover environment for a discrete set of user databases known as availability databases that fail over together. For more information, see [Always On Availability Groups](#).

In SQL Server 2017, SSIS introduces new capabilities that let you easily deploy to a centralized SSIS Catalog (i.e. SSISDB user database). In order to provide high availability for the SSISDB database and its contents - projects, packages, execution logs, and so on - you can add the SSISDB database to an Always On Availability Group, just like any other user database. When a failover occurs, one of the secondary nodes automatically becomes the new primary node.

For a detailed overview and step-by-step instructions for enabling Always On for SSISDB, see [SSIS Catalog](#).

#### Incremental package deployment

The Incremental Package Deployment feature lets you deploy one or more packages to an existing or new project without deploying the whole project. You can incrementally deploy packages by using the following tools.

- Deployment Wizard
- SQL Server Management Studio (which uses the Deployment Wizard)
- SQL Server Data Tools (Visual Studio) (which also uses the Deployment Wizard)
- Stored procedures
- The Management Object Model (MOM) API

For more info, see [\[Deploy Integration Services \(SSIS\) Projects and Packages\]\(../integration-services/packages/deploy-integration-services-ssis-projects-and-packages.md\)](#).

#### Support for Always Encrypted in the SSIS Catalog

SSIS already supports the Always Encrypted feature in SQL Server. For more info, see the following blog posts.

- [SSIS with Always Encrypted](#)
- [Lookup transformation with Always Encrypted](#)

#### Better debugging

##### New `ssis_logreader` database-level role in the SSIS catalog

In previous versions of the SSIS catalog, only users in the **ssis\_admin** role can access the views that contain logging output. There is now a new **ssis\_logreader** database-level role that you can use to grant permissions to access the views that contain logging output to users who aren't administrators.

There is also a new **ssis\_monitor** role. This role supports Always On and is for internal use only by the SSIS catalog.

##### New `RuntimeLineage` logging level in the SSIS catalog

The new **RuntimeLineage** logging level in the SSIS catalog collects the data required to track lineage information in the data flow. You can parse this lineage information to map the lineage relationship between tasks. ISVs and developers can build custom lineage mapping tools with this information.

##### New custom logging level in the SSIS catalog

Previous versions of the SSIS catalog let you choose from four built-in logging levels when you run a package: **None, Basic, Performance, or Verbose**. SQL Server 2017 adds the **RuntimeLineage** logging level. In addition, you can now create and save multiple customized logging levels in the SSIS catalog, and pick the logging level to use every time you run a package. For each customized logging level, select only the statistics and events you want to capture. Optionally include the event context to see variable values, connection strings, and task properties. For more info, see [Enable Logging for Package Execution on the SSIS Server](#).

#### Column names for errors in the data flow

When you redirect rows in the data flow that contain errors to an error output, the output contains a numeric identifier for the column in which the error occurred, but does not display the name of the column. There are now several ways to find or display the name of the column in which the error occurred.

- When you configure logging, select the **DiagnosticEx** event for logging. This event writes a data flow column map to the log. You can then look up the column name in this column map by using the column identifier captured by an error output. For more info, see [Error Handling in Data](#).
- In the Advanced Editor, you can see the column name for the upstream column when you view the properties of an input or output column of a data flow component.
- To see the names of the columns in which the error occurred, attach a Data Viewer to an error output. The Data Viewer now displays both the description of the error and the name of the column in which the error occurred.
- In the Script Component or a custom data flow component, call the new [GetIdentificationStringByID](#) method of the `IDTSComponentMetadata100` interface.

For more info about this improvement, see the following blog post by SSIS developer Bo Fan: [Error Column Improvements for SSIS Data Flow](#).

#### NOTE

(This support has been expanded in subsequent releases. For more info, see [Expanded support for error column names and New IDTSComponentMetaData130 interface in the API](#).)

#### Expanded support for error column names

The **DiagnosticEx** event now logs column information for all input and output columns, not just lineage columns. As a result we now call the output a pipeline column map instead of a pipeline lineage map.

The method `GetIdentificationStringByLineageID` has been renamed to [GetIdentificationStringByID](#). For more info, see [Column names for errors in the data flow](#).

For more info about this change and about the error column improvement, see the following updated blog post: [Error Column Improvements for SSIS Data Flow \(Updated for CTP3.3\)](#)

#### NOTE

(In RC0, this method has been moved to the new `IDTSComponentMetaData130` interface. For more info, see [New IDTSComponentMetaData130 interface in the API](#).)

#### Support for server-wide default logging level

In SQL Server **Server Properties**, under the **Server logging level** property, you can now select a default server-wide logging level. You can pick from one of the built-in logging levels - basic, none, verbose, performance, or runtime lineage - or you can pick an existing customized logging level. The selected logging level applies to all packages deployed to the SSIS Catalog. It also applies by default to a SQL Agent job step that runs an SSIS package.

#### New IDTSComponentMetaData130 interface in the API



The new [IDTSComponentMetaData130](#) interface adds new functionality in SQL Server 2016 to the existing [IDTSComponentMetaData100](#) interface, especially the [GetIdentificationStringByID](#) method. (The [GetIdentificationStringByID](#) method is moved to the new interface from the [IDTSComponentMetaData100](#) interface.) There are also new [IDTSInputColumn130](#) and [IDTSOutputColumn130](#) interfaces, both of which provide the [LineageIdentificationString](#) property. For more info, see [Column names for errors in the data flow](#).

## Better package management

### Improved experience for project upgrade

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.

### AutoAdjustBufferSize property automatically calculates buffer size for data flow

When you set the value of the new **AutoAdjustBufferSize** property to **true**, the data flow engine automatically calculates the buffer size for the data flow. For more info, see [Data Flow Performance Features](#).

### Reusable control flow templates

Save a commonly used control flow task or container to a standalone template file and reuse it multiple times in one or more packages in a project by using control flow templates. This reusability makes SSIS packages easier to design and maintain. For more info, see [Reuse Control Flow across Packages by Using Control Flow Package Parts](#).

### New templates renamed as parts

The new reusable control flow templates released in CTP 3.0 have been renamed as control flow parts or package parts. For more info about this feature, see [Reuse Control Flow across Packages by Using Control Flow Package Parts](#).

# Connectivity

## Expanded connectivity on premises

### Support for OData v4 data sources

The OData Source and the OData Connection Manager now support the OData v3 and v4 protocols.

- For OData V3 protocol, the component supports the ATOM and JSON data formats .
- For OData V4 protocol, the component supports the JSON data format .

For more info, see [OData Source](#).

### Explicit support for Excel 2013 data sources

The Excel Connection Manager, the Excel Source and the Excel Destination, and the SQL Server Import and Export Wizard now provide explicit support for Excel 2013 data sources.

### Support for the Hadoop file system (HDFS)

Support for HDFS contains connection managers to connect to Hadoop clusters and tasks to do common HDFS operations. For more info, see [Hadoop and HDFS Support in Integration Services \(SSIS\)](#).

### Expanded support for Hadoop and HDFS

- The Hadoop Connection Manager now supports both Basic and Kerberos authentication. For more info, see [Hadoop Connection Manager](#).
- The HDFS File Source and the HDFS File Destination now support both Text and Avro format. For more info, see [HDFS File Source](#) and [HDFS File Destination](#).
- The Hadoop File System task now supports the CopyWithinHadoop option in addition to the CopyToHadoop and the CopyFromHadoop options. For more info, see [Hadoop File System Task](#).

### HDFS File Destination now supports ORC file format

The HDFS File Destination now supports the ORC file format in addition to Text and Avro. (The HDFS File Source supports only Text and Avro.) For more info about this component, see [HDFS File Destination](#).

### **ODBC components updated for SQL Server 2016**

The ODBC Source and Destination components have been updated to provide full compatibility with SQL Server 2016. There is no new functionality and there are no changes in behavior.

### **Explicit support for Excel 2016 data sources**

The Excel Connection Manager, the Excel Source, and the Excel Destination now provide explicit support for Excel 2016 data sources.

### **Connector for SAP BW for SQL Server 2016 released**

The Microsoft® Connector for SAP BW for Microsoft SQL Server® 2016 has been released as part of the SQL Server 2016 Feature Pack. To download components of the Feature Pack, see [Microsoft® SQL Server® 2016 Feature Pack](#).

### **Connectors v4.0 for Oracle and Teradata released**

The Microsoft Connectors v4.0 for Oracle and Teradata have been released. To download the connectors, see [Microsoft Connectors v4.0 for Oracle and Teradata](#).

### **Connectors for Analytics Platform System (PDW) Appliance Update 5 released**

The destination adapters for loading data into PDW with AU5 have been released. To download the adapters, see [Analytics Platform System Appliance Update 5 Documentation and Client Tools](#).

### **Expanded connectivity to the cloud**

#### **Azure Feature Pack for SSIS released for SQL Server 2016**

The Azure Feature Pack for Integration Services has been released for SQL Server 2017. The feature pack contains connection managers to connect to Azure data sources and tasks to do common Azure operations. For more info, see [Azure Feature Pack for Integration Services \(SSIS\)](#).

#### **NOTE**

To ensure that the Azure Storage Connection Manager and the components that use it - that is, the Blob Source, the Blob Destination, the Blob Upload Task, and the Blob Download Task - can connect to both general-purpose storage accounts and to blob storage accounts, make sure you download the latest version of the Azure Feature Pack [here](#). For more info about these two types of storage accounts, see [Introduction to Microsoft Azure Storage](#).

### **Support for Microsoft Dynamics online resources released in Service Pack 1**

With SQL Server 2016 Service Pack 1 installed, the OData Source and OData Connection Manager now support connecting to the OData feeds of Microsoft Dynamics AX Online and Microsoft Dynamics CRM Online.

### **Support for Azure Data Lake Store released**

The latest version of the Azure Feature Pack includes a connection manager, source, and destination to move data to and from Azure Data Lake Store. For more info, see [Azure Feature Pack for Integration Services \(SSIS\)](#)

### **Support for Azure SQL Data Warehouse released**

The latest version of the Azure Feature Pack includes the Azure SQL DW Upload task for populating SQL Data Warehouse with data. For more info, see [Azure Feature Pack for Integration Services \(SSIS\)](#)

## **Usability and productivity**

### **Better install experience**

#### **Upgrade blocked when SSISDB belongs to an Availability Group**

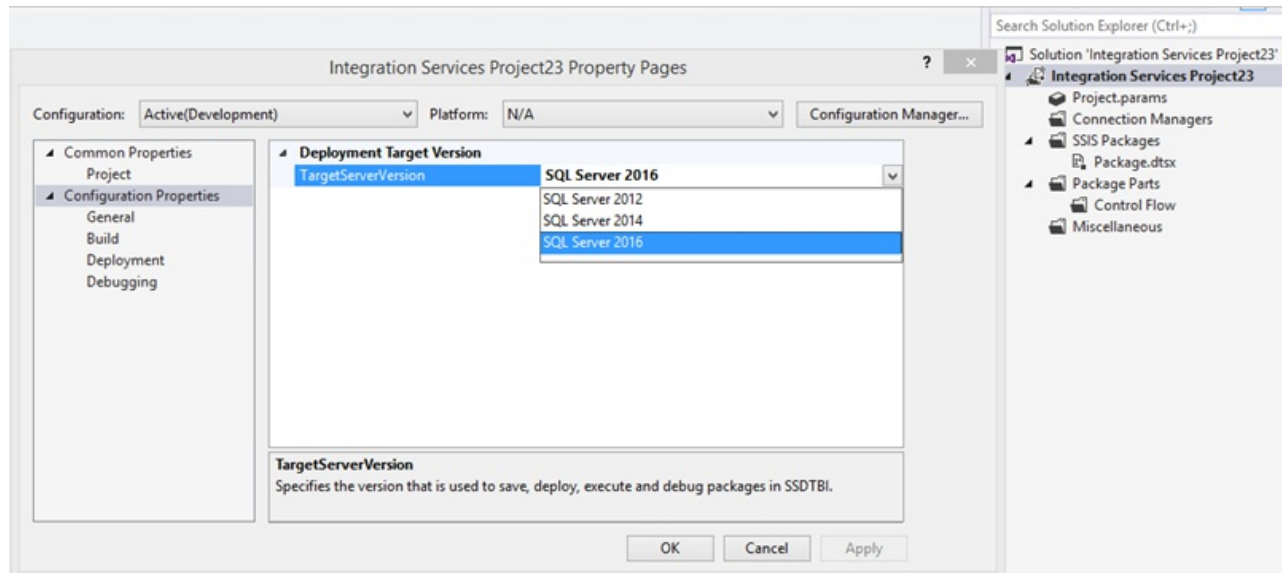
If the SSIS catalog database (SSISDB) belongs to an Always On Availability Group, you have to remove SSISDB from the availability group, upgrade SQL Server, then add SSISDB back to the availability group. For more info, see [Upgrading SSISDB in an availability group](#).

### **Better design experience**

#### **Multi-targeting and multi-version support in SSIS Designer**

You can now use SSIS Designer in SQL Server Data Tools (SSDT) for Visual Studio 2015 to create, maintain, and run packages that target SQL Server 2016, SQL Server 2014, or SQL Server 2012. To get SSDT, see [Download Latest SQL Server Data Tools](#).

In Solution Explorer, right-click on an Integration Services project and select **Properties** to open the property pages for the project. On the **General** tab of **Configuration Properties**, select the **TargetServerVersion** property, and then choose SQL Server 2016, SQL Server 2014, or SQL Server 2012.



#### IMPORTANT

If you develop custom extensions for SSIS, see [Support multi-targeting in your custom components](#) and [Getting your SSIS custom extensions to be supported by the multi-version support of SSDT 2015 for SQL Server 2016](#).

### Better management experience in SQL Server Management Studio

#### Improved performance for SSIS Catalog views

Most SSIS catalog views now perform better when they're run by a user who is not a member of the ssis\_admin role.

#### Other enhancements

##### Balanced Data Distributor transformation is now part of SSIS

The Balanced Data Distributor transformation, which required a separate download in previous versions of SQL Server, is now installed when you install Integration Services. For more info, see [Balanced Data Distributor Transformation](#).

##### Data Feed Publishing Components are now part of SSIS

The Data Feed Publishing Components, which required a separate download in previous versions of SQL Server, are now installed when you install Integration Services. For more info, see [Data Streaming Destination](#).

##### Support for Azure Blob Storage in the SQL Server Import and Export Wizard

The SQL Server Import and Export Wizard can now import data from, and save data to, Azure Blob Storage. For more info, see [Choose a Data Source \(SQL Server Import and Export Wizard\)](#) and [Choose a Destination \(SQL Server Import and Export Wizard\)](#).

## NOTE

To ensure that the Azure Storage Connection Manager and the components that use it - that is, the Blob Source, the Blob Destination, the Blob Upload Task, and the Blob Download Task - can connect to both general-purpose storage accounts and to blob storage accounts, make sure you download the latest version of the Azure Feature Pack [here](#). For more info about these two types of storage accounts, see [Introduction to Microsoft Azure Storage](#).

### Change Data Capture Designer and Service for Oracle for Microsoft SQL Server 2016 released

The Microsoft® Change Data Capture Designer and Service for Oracle by Attunity for Microsoft SQL Server® 2016 have been released as part of the SQL Server 2016 Feature Pack. These components now support Oracle 12c in classic installation. (Multitenant installation is not supported) To download components of the Feature Pack, see [Microsoft® SQL Server® 2016 Feature Pack](#).

### CDC components updated for SQL Server 2016

The CDC (Change Data Capture) Control Task, Source, and Splitter Transformation components have been updated to provide full compatibility with SQL Server 2016. There is no new functionality and there are no changes in behavior.

### Analysis Services Execute DDL Task updated

The Analysis Services Execute DDL Task has been updated to accept Tabular Model Scripting Language commands.

### Analysis Services tasks support tabular models

You can now use all the SSIS task and destinations that support SQL Server Analysis Services (SSAS) with SQL Server 2016 tabular models. The SSIS tasks have been updated to represent tabular objects instead of multidimensional objects. For example, when you select objects to process, the Analysis Services Processing Task automatically detects a Tabular model and displays a list of Tabular objects instead of showing measure groups and dimensions. The Partition Processing Destination now also shows tabular objects and supports pushing data into a partition.

The Dimension Processing Destination does not work for Tabular models with the SQL 2016 compatibility level. The Analysis Services Processing Task and the Partition Processing Destination are all you need for tabular processing.

### Support for Built-in R Services

SSIS already supports the built-in R services in SQL Server. You can use SSIS not only to extract data and load the output of analysis, but to build, run and periodically retrain R models. For more info, see the following log post. [Operationalize your machine learning project using SQL Server 2016 SSIS and R Services](#).

### Rich XML validation output in the XML Task

Validate XML documents and get rich error output by enabling the **ValidationDetails** property of the XML Task. Before the **ValidationDetails** property was available, XML validation by the XML Task returned only a true or false result, with no information about errors or their locations. Now, when you set **ValidationDetails** to true, the output file contains detailed information about every error including the line number and the position. You can use this information to understand, locate, and fix errors in XML documents. For more info, see [Validate XML with the XML Task](#).

SSIS introduced the **ValidationDetails** property in SQL Server 2012 Service Pack 2. This new property was not announced or documented at that time. The **ValidationDetails** property is also available in SQL Server 2014 and in SQL Server 2016.

## See Also

[What's New in SQL Server 2016](#)

[Editions and Supported Features for SQL Server 2016](#)



**Need help?** [MSDN Forum](#), [Stackoverflow](#), [Connect](#)

# What's New in Integration Services in SQL Server 2017

5/17/2017 • 2 min to read • [Edit Online](#)

This topic describes the features that have been added or updated in SQL Server 2017 Integration Services.

## NOTE

SQL Server 2017 also includes the features of SQL Server 2016 and the features added in SQL Server 2016 updates. For info about the new SSIS features in SQL Server 2016, see [What's New in Integration Services in SQL Server 2016](#).

## New in SSIS in SQL Server 2017 CTP 2.1

### New features in Scale Out for SSIS

1. You can now use the **Use32BitRuntime** parameter when you trigger execution in Scale Out.
2. The performance of logging to SSISDB for package executions in Scale Out has been improved. The Event Message and Message Context logs are now written to SSISDB in batch mode instead of one by one. Here are some additional notes about this improvement:
  - Some reports in the current version of SQL Server Management Studio (SSMS) don't currently display these logs for executions in Scale Out. We anticipate that they will be supported in the next release of SSMS. The affected reports include the *All Connections* report, the *Error Context* report, and the *Connection Information* section in the Integration Service Dashboard.
  - A new column **event\_message\_guid** has been added. Use this column to join the [catalog].[event\_message\_context] view and the [catalog].[event\_messages] view instead of using **event\_message\_id** when you query these logs of executions in Scale Out.

## New in SSIS in SQL Server 2017 CTP 2.0

There are no new SSIS features in SQL Server 2017 CTP 2.0.

## New in SSIS in SQL Server 2017 CTP 1.4

There are no new SSIS features in SQL Server 2017 CTP 1.4.

## New in SSIS in SQL Server 2017 CTP 1.3

There are no new SSIS features in SQL Server 2017 CTP 1.3.

## New in SSIS in SQL Server 2017 CTP 1.2

There are no new SSIS features in SQL Server 2017 CTP 1.2.

## New in SSIS in SQL Server 2017 CTP 1.1

There are no new SSIS features in SQL Server 2017 CTP 1.1.

## New in SSIS in SQL Server 2017 CTP 1.0

## **Scale Out for SSIS**

The Scale Out feature makes it much easier to run SSIS on multiple machines.

After installing the Scale Out Master and Workers, the package can be distributed to execute on different Workers automatically. If the execution is terminated unexpectedly, the execution is retried automatically. Also, all the executions and Workers can be centrally managed using the Master.

For more information, see [Integration Services Scale Out](#).

## **Support for Microsoft Dynamics Online Resources**

The OData Source and OData Connection Manager now support connecting to the OData feeds of Microsoft Dynamics AX Online and Microsoft Dynamics CRM Online.

# Integration Services Features Supported by the Editions of SQL Server

5/24/2017 • 1 min to read • [Edit Online](#)

This topic provides details about the features of SQL Server Integration Services (SSIS) supported by the different editions of SQL Server.

For features supported by Evaluation and Developer editions, see features listed for Enterprise Edition in the tables below.

For the latest release notes and what's new information, see the following:

- [SQL Server 2016 release notes](#)
- [What's New in Integration Services in SQL Server 2016](#)
- [What's New in Integration Services in SQL Server 2017](#)

## Try SQL Server 2016!

The SQL Server Evaluation edition is available for a 180-day trial period.



[Download SQL Server 2016 from the Evaluation Center](#)

## New Integration Services Features in SQL Server 2017

FEATURE	ENTERPRISE	STANDARD	WEB	EXPRESS WITH ADVANCED SERVICES	EXPRESS
Scale Out	Yes	TBD	TBD	TBD	TBD
Support for Microsoft Dynamics AX and Microsoft Dynamics CRM in OData components <sup>1</sup>	Yes	Yes			

<sup>1</sup> This feature is also supported in SQL Server 2016 with Service Pack 1. C

## Integration Services

FEATURE	ENTERPRISE	STANDARD	WEB	EXPRESS WITH ADVANCED SERVICES	EXPRESS
Built-in data source connectors	Yes	Yes	Yes	Yes	Yes

FEATURE	ENTERPRISE	STANDARD	WEB	EXPRESS WITH ADVANCED SERVICES	EXPRESS
Azure data source connectors and tasks	Yes	Yes	Yes	Yes	Yes
SQL Server Import and Export Wizard	Yes	Yes	Yes	Yes	Yes
Hadoop / HDFS connectors and tasks	Yes	Yes	Yes		
SSIS Designer and runtime	Yes	Yes			
Built in tasks and transformations	Yes	Yes			
Basic data profiling tools	Yes	Yes			
Change Data Capture Service for Oracle by Attunity	Yes				
Change Data Capture Designer for Oracle by Attunity	Yes				

## Integration Services - Advanced Adapters

FEATURE	ENTERPRISE	STANDARD	WEB	EXPRESS WITH ADVANCED SERVICES	EXPRESS
High performance Oracle destination	Yes				
High performance Teradata Destination	Yes				
SAP BW source and destination	Yes				



FEATURE	ENTERPRISE	STANDARD	WEB	EXPRESS WITH ADVANCED SERVICES	EXPRESS
Data mining model training destination adapter	Yes				
Dimension processing destination adapter	Yes				
Partition processing destination adapter	Yes				
Change Data Capture components by Attunity	Yes				
Connector for Open Database Connectivity (ODBC) by Attunity	Yes				

## Integration Services - Advanced Transforms

FEATURE	ENTERPRISE	STANDARD	WEB	EXPRESS WITH ADVANCED SERVICES	EXPRESS
Persistent (high performance) lookups	Yes				
Data mining query transformation	Yes				
Fuzzy grouping and lookup transformations	Yes				
Term extractions and lookup transformations	Yes				

# Integration Services Backward Compatibility

3/24/2017 • 1 min to read • [Edit Online](#)

This topic describes changes between versions of SQL Server Integration Services. It covers features that are no longer available or are scheduled to be removed in a future release. It also describes changes to the product that are known to break, or to change the behavior of, an existing application that includes Integration Services functionality.

## Deprecated Integration Services Features in SQL Server 2016

This section describes deprecated Integration Services features that are still available in the current release of SQL Server Integration Services. These features are scheduled to be removed in a future release of SQL Server. Do not use deprecated features in new applications.

There are no deprecated Integration Services features in SQL Server 2017.

## Discontinued Integration Services Functionality in SQL Server 2016

This section describes Integration Services features that are no longer available in the current release of SQL Server Integration Services.

There are no discontinued Integration Services features in SQL Server 2017.

## Breaking Changes to Integration Services Features in SQL Server 2016

This section describes breaking changes in Integration Services. These changes may break applications, scripts, or other items that are based on earlier versions of SQL Server. You may encounter these issues after you upgrade.

There are no breaking changes to Integration Services features in SQL Server 2017.

## Behavior Changes to Integration Services Features in SQL Server 2016

This section describes behavior changes in Integration Services. Behavior changes affect how features work or interact in the current release of SQL Server Integration Services when compared to earlier versions of SQL Server.

There are no behavior changes for Integration Services features in SQL Server 2017.

# Upgrade Integration Services

4/14/2017 • 9 min to read • [Edit Online](#)

If SQL Server 2008 Integration Services (SSIS) or later is currently installed on your computer, you can upgrade to SQL Server 2017 Integration Services (SSIS).

When you upgrade to SQL Server 2017 Integration Services (SSIS) on a machine that has one of these earlier versions of Integration Services installed, SQL Server 2017 Integration Services (SSIS) is installed side-by-side with the earlier version.

With this side-by-side install, multiple versions of dtexec utility are installed. To ensure that you run the correct version of the utility, at the command prompt run the utility by entering the full path (<drive>:\Program Files\Microsoft SQL Server\DTS\Binn). For more information about dtexec, see [dtexec Utility](#).

## NOTE

In previous versions of SQL Server, by default when you installed SQL Server all users in the Users group had access to the Integration Services service. When you install SQL Server 2017, users do not have access to the Integration Services service. The service is secure by default. After SQL Server 2017 is installed, the SQL Server administrator must run the DCOM Configuration tool (Dcomcnfg.exe) to grant specific users access to the Integration Services service. For more information, see [Integration Services Service \(SSIS Service\)](#).

## Before Upgrading Integration Services

We recommend that you run Upgrade Advisor before you upgrade to SQL Server 2017. Upgrade Advisor reports issues that you might encounter if you migrate existing Integration Services packages to the new package format that SQL Server 2017 uses.

## NOTE

Support for migrating or running Data Transformation Services (DTS) packages has been discontinued in the current release of SQL Server Integration Services. The following DTS functionality has been discontinued.

- DTS runtime
  - DTS API
  - Package Migration Wizard for migrating DTS packages to the next version of Integration Services
  - Support for DTS package maintenance in SQL Server Management Studio
  - Execute DTS 2000 Package task
  - Upgrade Advisor scan of DTS packages.

For information about other discontinued features, see [Discontinued Integration Services Functionality in SQL Server 2016](#).

## Upgrading Integration Services

You can upgrade by using one of the following methods:

- Run SQL Server 2017 Setup and select the option to **Upgrade from SQL Server 2008, SQL Server 2008 R2, SQL Server 2012, or SQL Server 2014**.
- Run **setup.exe** at the command prompt and specify the **/ACTION=upgrade** option. For more information,

see the section, "Installation Scripts for Integration Services," in [Install SQL Server 2016 from the Command Prompt](#).

You cannot use upgrade to perform the following actions:

- Reconfigure an existing installation of Integration Services.
- Move from a 32-bit to a 64-bit version of SQL Server or from a 64-bit version to a 32-bit version.
- Move from one localized version of SQL Server to another localized version.

When you upgrade, you can upgrade both Integration Services and the Database Engine, or just upgrade the Database Engine, or just upgrade Integration Services. If you upgrade only the Database Engine, SQL Server 2008 Integration Services (SSIS) or later remains functional, but you do not have the functionality of SQL Server 2017 Integration Services (SSIS). If you upgrade only Integration Services, SQL Server 2017 Integration Services (SSIS) is fully functional, but can only store packages in the file system, unless an instance of the SQL Server 2017 Database Engine is available on another computer.

## Upgrading Both Integration Services and the Database Engine to SQL Server 2017

This section describes the effects of performing an upgrade that has the following criteria:

- You upgrade both Integration Services and an instance of the Database Engine to SQL Server 2017.
- Both Integration Services and the instance of the Database Engine are on the same computer.

### What the Upgrade Process Does

The upgrade process does the following tasks:

- Installs the SQL Server 2017 Integration Services (SSIS) files, service, and tools ( Management Studio and SQL Server Data Tools). When there are multiple instances of SQL Server 2008, SQL Server 2008 R2, SQL Server 2012, or SQL Server 2014 on the same computer, the first time you upgrade any of the instances to SQL Server 2017, the SQL Server 2017 Integration Services (SSIS) files, service, and tools are installed.
- Upgrades the instance of the SQL Server 2008, SQL Server 2008 R2, SQL Server 2012, or SQL Server 2014 Database Engine to the SQL Server 2017 version.
- Moves data from the SQL Server 2008 Integration Services (SSIS) or later system tables to the SQL Server 2017 Integration Services (SSIS) system tables, as follows:
  - Moves packages without change from the msdb.dbo.sysdtspackages90 system table to the msdb.dbo.sysssispackages system table.

#### NOTE

Although the data moves to a different system table, the upgrade process does not migrate packages to the new format.

- Moves folder metadata from the msdb.sysdtsfolders90 system table to the msdb.sysssisfolders system table.
  - Moves log data from the msdb.sysdtslog90 system table to the msdb.sysssislog system table.
- Removes the msdb.sysdts90 system tables and the stored procedures that are used to access them after moving the data to the new msdb.sysssis\ tables. However, upgrade replaces the sysdtslog90 table with a view that is also named sysdtslog90. This new sysdtslog90 view exposes the new msdb.sysssislog system table. This ensures that reports based on the log table continue to run without interruption.

- To control access to packages, creates three new fixed database-level roles: db\_ssisadmin, db\_ssisltduser, and db\_ssisoperator. The SQL Server 2005 Integration Services roles of db\_dtsadmin, db\_dtsltduser, and db\_dtsoperator are not removed, but are made members of the corresponding new roles.
- If the SSIS package store (that is, the file system location managed by the Integration Services service) is the default location under **\SQL Server\90**, **\SQL Server\100**, **\SQL Server\110**, or **\SQL Server\120** moves those packages to the new default location under **\SQL Server\130**.
- Updates the Integration Services service configuration file to point to the upgraded instance of the Database Engine.

### What the Upgrade Process Does Not Do

The upgrade process does not do the following tasks:

- **Does not** remove the SQL Server 2008 Integration Services (SSIS) or later service.
- Does not migrate existing Integration Services packages to the new package format that SQL Server 2017 uses. For information about how to migrate packages, see [Upgrade Integration Services Packages](#).
- Does not move packages from file system locations, other than the default location, that have been added to the service configuration file. If you have previously edited the service configuration file to add more file system folders, packages that are stored in those folders will not be moved to a new location.
- In SQL Server Agent job steps that call the **dtexec** utility (dtexec.exe) directly, does not update the file system path for the **dtexec** utility. You have to edit these job steps manually to update the file system path to specify the SQL Server 2017 location for the **dtexec** utility.

### What You Can Do After Upgrading

After the upgrade process finishes, you can do the following tasks:

- Run SQL Server Agent jobs that run packages.
- Use Management Studio to manage Integration Services packages that are stored in an instance of SQL Server 2008, SQL Server 2008 R2, SQL Server 2012, or SQL Server 2014. You need to modify the service configuration file to add the instance of SQL Server 2008, SQL Server 2008 R2, SQL Server 2012, or SQL Server 2014 to the list of locations managed by the service.

#### NOTE

Early versions of Management Studio cannot connect to SQL Server 2017 Integration Services (SSIS) Service.

- Identify the version of packages in the msdb.dbo.sysssispackages system table by checking the value in the packageformat column. The table has a packageformat column that identifies the version of each package. A value of 3 indicates a SQL Server 2008 Integration Services (SSIS) package. Until you migrate packages to the new package format, the value in the packageformat column does not change.
- You cannot use the SQL Server 2008, SQL Server 2008 R2, SQL Server 2012, or SQL Server 2014 tools to design, run, or manage Integration Services packages. The SQL Server 2008, SQL Server 2008 R2, SQL Server 2012, or SQL Server 2014 tools include the respective versions of SQL Server Data Tools (SSDT), the SQL Server Import and Export Wizard, and the Package Execution Utility (dtexecui.exe). The upgrade process does not remove the SQL Server 2008, SQL Server 2008 R2, SQL Server 2012, or SQL Server 2014 tools. However, you will not be able to use these tools to continue to work with SQL Server 2008 Integration Services (SSIS) or later packages on a server that has been upgraded.
- By default, in an upgrade installation, Integration Services is configured to log events that are related to the running of packages to the Application event log. This setting might generate too many event log entries when you use the Data Collector feature of SQL Server 2017. The events that are logged include EventID

12288, "Package started," and EventID 12289, "Package finished successfully." To stop logging these two events to the Application event log, open the registry for editing. Then in the registry, locate the HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Microsoft SQL Server\130\SSIS node, and change the DWORD value of the LogPackageExecutionToEventLog setting from 1 to 0.

## Upgrading only the Database Engine to SQL Server 2017

This section describes the effects of performing an upgrade that has the following criteria:

- You upgrade only an instance of the Database Engine. That is, the instance of the Database Engine is now an instance of SQL Server 2017, but the instance of Integration Services and the client tools are from SQL Server 2008, SQL Server 2008 R2, SQL Server 2012, or SQL Server 2014.
- The instance of the Database Engine is on one computer, and Integration Services and the client tools are on another computer.

### What You Can Do After Upgrading

The system tables that store packages in the upgraded instance of the Database Engine are not the same as those used in SQL Server 2008. Therefore, the SQL Server 2008 versions of Management Studio and SQL Server Data Tools cannot discover the packages in the system tables on the upgraded instance of the Database Engine. Because these packages cannot be discovered, there are limitations on what you can do with those packages:

- You cannot use the SQL Server 2008 tools, Management Studio and SQL Server Data Tools, on other computers to load or manage packages from the upgraded instance of the Database Engine.

#### NOTE

Although the packages in the upgraded instance of the Database Engine have not yet been migrated to the new package format, they are not discoverable by the SQL Server 2008 tools. Therefore, the packages cannot be used by the SQL Server 2008 tools.

- You cannot use SQL Server 2008 Integration Services (SSIS) on other computers to run packages that are stored in msdb on the upgraded instance of the Database Engine.
- You cannot use SQL Server Agent jobs on SQL Server 2008 computers to run SQL Server 2008 Integration Services (SSIS) packages that are stored in the upgraded instance of the Database Engine.

## External Resources

Blog entry, [Making your Existing Custom SSIS Extensions and Applications Work in Denali](#), on blogs.msdn.com.

# Integration Services (SSIS) Development and Management Tools

3/24/2017 • 1 min to read • [Edit Online](#)

Integration Services includes two studios for working with Integration Services:

- SQL Server Data Tools (SSDT) for developing the Integration Services packages that a business solution requires. SQL Server Data Tools (SSDT) provides the Integration Services project in which you create packages.
- SQL Server Management Studio for managing packages in a production environment.

## SQL Server Data Tools

Working in SQL Server Data Tools (SSDT), 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 SQL Server Data Tools (SSDT).
- 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.

For more information about SQL Server Data Tools (SSDT), see [SQL Server Data Tools](#).

## SQL Server Management Studio

SQL Server Management Studio provides the Integration Services service that you use to manage packages, monitor running packages, and determine impact and data lineage for Integration Services and SQL Server objects.

Working in SQL Server Management Studio, you can perform the following tasks:

- Create folders to organize packages in a way that is meaningful to your organization.
- Run packages that are stored on the local computer by using the Execute Package utility.
- Run the Execute Package utility to generate a command line to use when you run the **dtexec** command prompt utility (dtexec.exe).
- Import and export packages to and from the SQL Server msdb database, the SSIS Package Store, and the file system.

# Integration Services (SSIS) Projects and Solutions

3/24/2017 • 8 min to read • [Edit Online](#)

SQL Server provides SQL Server Data Tools (SSDT) for the development of Integration Services packages.

Integration Services packages reside in projects. To create and work with Integration Services projects, you must install the SQL Server Data Tools (SSDT) environment. For more information, see [Install Integration Services](#).

When you create a new Integration Services project in SQL Server Data Tools (SSDT), the **New Project** dialog box includes an **Integration Services Project** template. This project template creates a new project that contains a single package.

## Projects and solutions

Projects are stored in solutions. You can create a solution first and then add an Integration Services project to the solution. If no solution exists, SQL Server Data Tools (SSDT) automatically creates one for you when you first create the project. A solution can contain multiple projects of different types.

### TIP

By default, when you create a new project in SQL Server Data Tools, the solution is not shown in **Solution Explorer** pane. To change this default behavior, on the **Tools** menus, click **Options**. In the **Options** dialog box, expand **Projects and Solutions**, and then click **General**. On the **General** page, select **Always show solution**.

## Solutions contain projects

A solution is a container that groups and manages the projects that you use when you develop end-to-end business solutions. A solution lets you handle multiple projects as one unit and to bring together one or more related projects that contribute to a business solution.

Solutions can include different types of projects. If you want to use SSIS Designer to create an Integration Services package, you work in an Integration Services project in a solution provided by SQL Server Data Tools (SSDT).

When you create a new solution, SQL Server Data Tools (SSDT) adds a Solution folder to Solution Explorer, and creates files that have the extensions, .sln and .suo:

- The \*.sln file contains information about the solution configuration and lists the projects in the solution.
- The \*.suo file contains information about your preferences for working with the solution.

While SQL Server Data Tools (SSDT) automatically creates a solution when you create a new project, you can also create a blank solution, and then add projects later.

## Integration Services projects contain packages

A project is a container in which you develop Integration Services packages.

In SQL Server Data Tools (SSDT), an Integration Services project stores and groups the files that are related to the package. For example, a project includes the files that are required to create a specific extract, transfer, and load (ETL) solution.

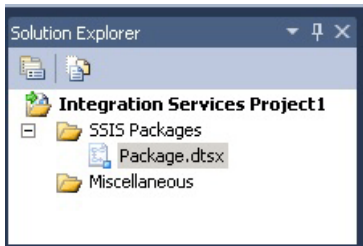
Before you create an Integration Services project, you should become familiar with the basic contents of this kind of project. After you understand what a project contains, you can begin creating and working with an Integration



Services project.

## Folders in Integration Services projects

The following diagram shows the folders in an Integration Services project in SQL Server Data Tools (SSDT).



The following table describes the folders that appear in an Integration Services project.

FOLDER	DESCRIPTION
SSIS Packages	Contains packages. For more information, see <a href="#">Integration Services (SSIS) Packages</a> .
Miscellaneous	Contains files other than package files.

## Files in Integration Services projects

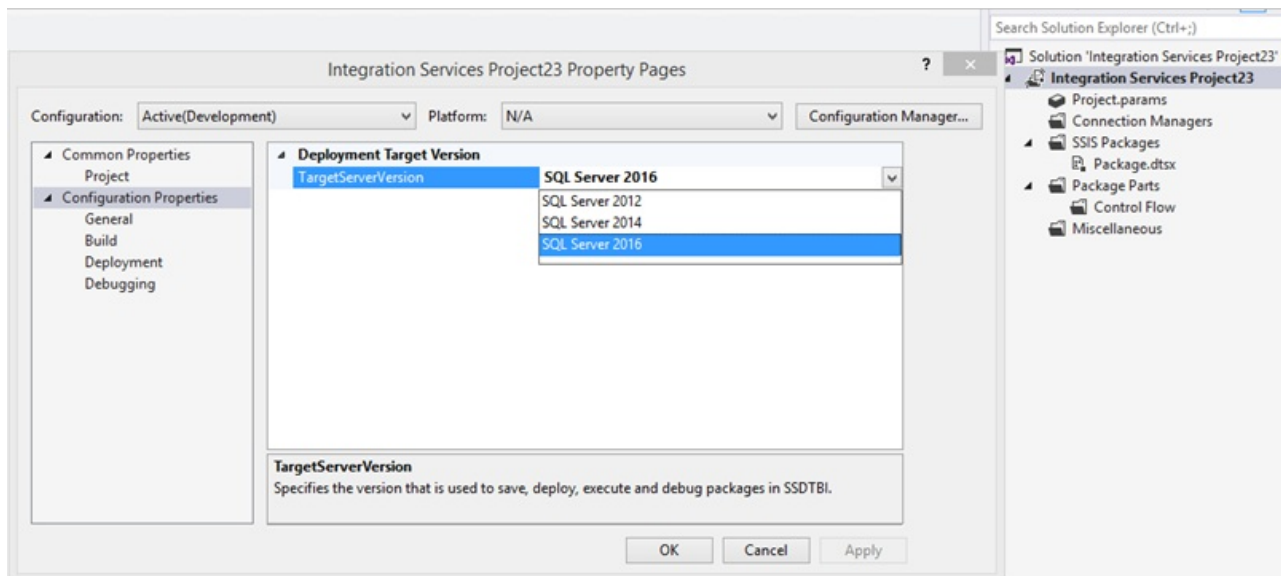
When you add a new or an existing Integration Services project to a solution, SQL Server Data Tools (SSDT) creates project files that have the extensions \*.dtproj and \*.dtproj.user and \*.database.

- The \*.dtproj file contains information about project configurations and items such as packages.
- The \*.dtproj.user file contains information about your preferences for working with the project.
- The \*.database file contains information that SQL Server Data Tools (SSDT) requires to open the Integration Services project.

## Version targeting in Integration Services projects

In SQL Server Data Tools (SSDT), you can create, maintain, and run packages that target SQL Server 2016, SQL Server 2014, or SQL Server 2012.

In Solution Explorer, right-click on an Integration Services project and select **Properties** to open the property pages for the project. On the **General** tab of **Configuration Properties**, select the **TargetServerVersion** property, and then choose SQL Server 2016, SQL Server 2014, or SQL Server 2012.



## Create a new Integration Services project

1. Open SQL Server Data Tools (SSDT).
2. On the **File** menu, point to **New**, and then click **Project**.
3. In the **New Project** dialog box, in the **Templates** pane, select the **Integration Services Project** template.

The **Integration Services Project** template creates an Integration Services project that contains a single, empty package.

4. (Optional) Edit the project name and the location.

The solution name is automatically updated to match the project name.

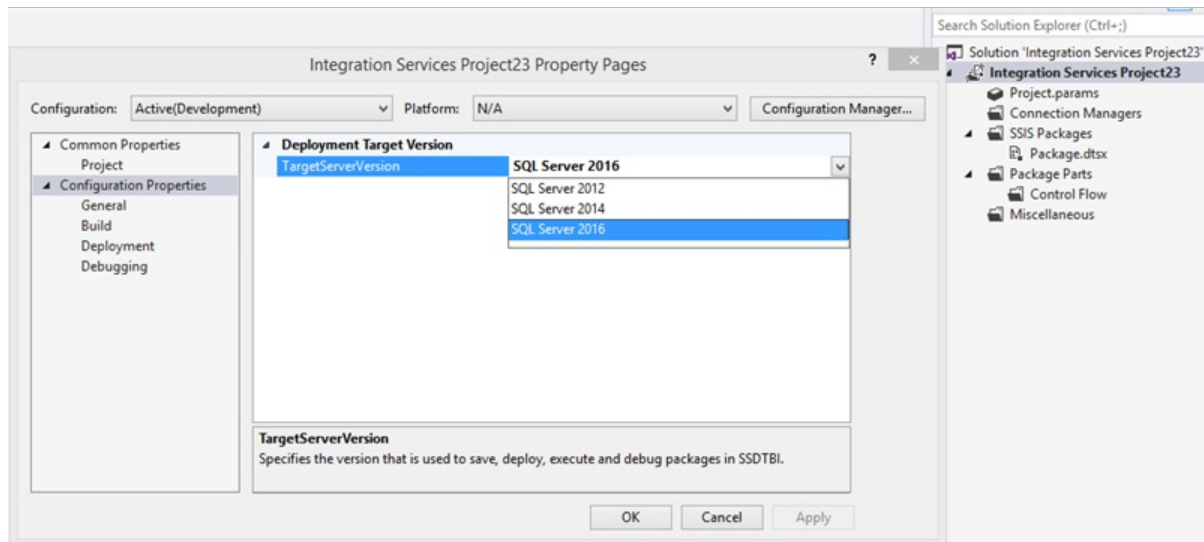
5. To create a separate folder for the solution file, select **Create directory for solution**. This is the default option.
6. If source control software is installed on the computer, select **Add to source control** to associate the project with source control.
7. If the source control software is Microsoft Visual SourceSafe, the **Visual SourceSafe Login** dialog box opens. In **Visual SourceSafe Login**, provide a user name, a password, and the name of the Microsoft Visual SourceSafe database. Click **Browse** to locate the database.

**NOTE:** To view and change the selected source control plug-in and to configure the source control environment, click **Options** on the **Tools** menu, and then expand the **Source Control** node.

8. Click **OK** to add the solution to **Solution Explorer** and add the project to the solution.

## Choose the target version of a project and its packages

1. In Solution Explorer, right-click on an Integration Services project and select **Properties** to open the property pages for the project.
2. On the **General** tab of **Configuration Properties**, select the **TargetServerVersion** property, and then choose SQL Server 2016, SQL Server 2014, or SQL Server 2012.



You can create, maintain, and run packages that target SQL Server 2016, SQL Server 2014, or SQL Server 2012.

## Import an existing project with the Import Project Wizard

1. In Visual Studio, click **New** > **Project** on the **File** menu.
2. In the **Installed Templates** area of the **New Project** window, expand **Business Intelligence**, and click **Integration Services**.
3. Select **Integration Services Import Project Wizard** from the project types list.
4. Type a name for the new project to be created in the **Name** text box.
5. Type the path or location for the project in the **Location** text box, or click **Browse** to select one.
6. Type a name for the solution in the **Solution name** text box.
7. Click **OK** to launch the **Integration Services Import Project Wizard** dialog box.
8. Click **Next** to switch to the **Select Source** page.
9. If you are importing from an **.ispac** file, type the path including file name in the **Path** text box. Click **Browse** to navigate to the folder where you want the solution to be stored and type file name in the **File name** text box, and click **Open**.

If you are importing from an **Integration Services Catalog**, type the database instance name in the **Server name** text box or click **Browse** and select the database instance that contains the catalog.

Click **Browse** next to **Path** text box, expand folder in the catalog, select the project you want to import, and click **OK**.

Click **Next** to switch to the **Review** page.

10. Review the information and click **Import** to create a project based on the existing project you selected.
11. Optional: click **Save Report** to save the results to a file
12. Click **Close** to close the **Integration Services Import Project Wizard** dialog box.

## Add a project to a solution

When you add a project, you can have Integration Services create a new, blank project, or you can add a project

that you have already created for a different solution. You can only add a project to an existing solution when the solution is visible in SQL Server Data Tools (SSDT).

### Add a new project to a solution

1. In SQL Server Data Tools (SSDT), open the solution to which you want to add a new Integration Services project, and do one of the following:
  - Right-click the solution, click **Add**, and then click **New Project**.
  - On the **File** menu, point to **Add**, and then click **New Project**.
2. In the **Add New Project** dialog box, click **Integration Services Project** in the **Templates** pane.
3. Optionally, edit the project name and location.
4. Click **OK**.

### Add an existing project to a solution

1. In SQL Server Data Tools (SSDT), open the solution to which you want to add an existing Integration Services project, and do one of the following:
  - Right-click the solution, point to **Add**, and then click **Existing Project**.
  - On the **File** menu, click **Add**, and then click **Existing Project**.
2. In the **Add Existing Project** dialog box, browse to locate the project you want to add, and then click **Open**.
3. The project is added to the solution folder in **Solution Explorer**.

## Remove a project from a solution

You can only remove a project from a solution when the solution is visible in SQL Server Data Tools (SSDT). After the solution is visible, you can remove all except one project. As soon as only one project remains, SQL Server Data Tools (SSDT) no longer displays the solution folder and you cannot remove the last project.

1. In SQL Server Data Tools (SSDT), open the solution from which you want to remove an Integration Services project.
2. In Solution Explorer, right-click the project, and then click **Unload Project**.
3. Click **OK** to confirm the removal.

## Add an item to a project

1. In SQL Server Data Tools (SSDT), open the solution that contains the Integration Services project to which you want to add an item.
2. In Solution Explorer, right-click the project, point to **Add**, and do one of the following:
  - Click **New Item**, and then select a template from the **Templates** pane in the **Add New Item** dialog box.
  - Click **Existing Item**, browse in the **Add Existing Item** dialog box to locate the item you want to add to the project, and then click **Add**.
3. The new item appears in the appropriate folder in Solution Explorer.

## Copy project items

You can copy objects within an Integration Services project or between Integration Services projects. You can also

copy objects between the other types of SQL Server Data Tools (SSDT) projects, Reporting Services and Analysis Services. To copy between projects, the project must be part of the same SQL Server Data Tools (SSDT) solution.

1. In SQL Server Data Tools (SSDT), open the Integration Services project or solution that you want to work with.
2. Expand the project and item folder to copy from.
3. Right-click the item and click **Copy**.
4. Right-click the Integration Services project to copy to and click **Paste**.

The items are automatically copied to the correct folder. If you copy items to the Integration Services project that are not packages, the items are copied to the **Miscellaneous** folder.

# Integration Services User Interface

3/24/2017 • 5 min to read • [Edit Online](#)

In addition to the design surfaces on the SSIS Designer tabs, the user interface provides access to the following windows and dialog boxes for adding features to packages and configuring the properties of package objects:

- The dialog boxes and windows that you use to add functionality such as logging and configurations to packages.
- The custom editors for configuring the properties of package objects. Almost every type of container, task, and data flow component has its own custom editor.
- The **Advanced Editor** dialog box, a generic editor that provides more detailed configuration options for many data flow components.

SQL Server Data Tools (SSDT) also provides windows and dialog boxes for configuring the environment and working with packages.

## Dialog Boxes and Windows

After you open a package or create a new package in SSIS Designer, the following dialog boxes and windows are available.

This table lists the dialog boxes that are available from the **SSIS** menu and the design surfaces of SSIS Designer.

DIALOG BOX	PURPOSE	ACCESS
<b>Getting Started</b>	Access samples, tutorials, and videos.	On the design surface of the <b>Control Flow</b> tab or the <b>Data Flow</b> tab, right-click and then click <b>Getting Started</b> .  To automatically display the <b>Getting Started</b> window when you create a new Integration Services project, select <b>Always show in new project</b> at the bottom of the window.
<b>Configure SSIS Logs</b>	Configure logging for a package and its tasks by adding logs and setting logging details.	On the <b>SSIS</b> menu, click <b>Logging</b> .  -or-  Right-click anywhere on the design surface of the <b>Control Flow</b> tab, and then click <b>Logging</b> .
<b>Package Configuration Organizer</b>	Add and edit package configurations. You run the Package Configuration Wizard from this dialog box.	On the <b>SSIS</b> menu, click <b>Package Configurations</b> .  -or-  Right-click anywhere on the design surface of the <b>Control Flow</b> tab, and then click <b>Package Configurations</b> .

DIALOG BOX	PURPOSE	ACCESS
<b>Digital Signing</b>	Sign a package or remove the signature from the package.	On the <b>SSIS</b> menu, click <b>Digital Signing</b> .  -or-  Right-click anywhere on the design surface of the <b>Control Flow</b> tab, and then click <b>Digital Signing</b> .
<b>Set Breakpoints</b>	Enable breakpoints on tasks and set breakpoint properties.	On the design surface of the <b>Control Flow</b> tab, right-click a task or container, and then click <b>Edit Breakpoints</b> . To set a breakpoint on the package, right-click anywhere on the design surface of the <b>Control Flow</b> tab, and then click <b>Edit Breakpoints</b> .

The **Getting Started** window provides links to samples, tutorials, and videos. To add links to additional content, modify the SamplesSites.xml file that is included with the current release of SQL Server Integration Services. It is recommended that you not modify the <GettingStartedSamples> element value that specifies the RSS feed URL. The file is located in the <drive>:\Program Files\Microsoft SQL Server\110\DTS\Binn folder. On a 64-bit computer, the file is located in the <drive>:\Program Files(x86)\Microsoft SQL Server\110\DTS\Binn folder

If the SamplesSites.xml file does become corrupted, replace the xml in the file with the following default xml.

```
<?xml version="1.0" ?>

- <SamplesSites>

<GettingStartedSamples>http://go.microsoft.com/fwlink/?LinkID=203147</GettingStartedSamples>

- <ToolboxSamples>

<Site>http://go.microsoft.com/fwlink/?LinkID=203286&query=SSIS%20{0}</Site>

</ToolboxSamples>

</SamplesSites>
```

This table lists the windows that are available from the **SSIS** and **View** menus and the design surfaces of SSIS Designer.

WINDOW	PURPOSE	ACCESS
<b>Variables</b>	Add and manage custom variables.	On the <b>SSIS</b> menu, click <b>Variables</b> .  -or-  Right-click anywhere in the design surface of the <b>Control Flow</b> and <b>Data Flow</b> tabs, and then click <b>Variables</b> .  -or-  On the <b>View</b> menu, point to <b>Other Windows</b> , and then click <b>Variables</b> .

WINDOW	PURPOSE	ACCESS
<b>Log Events</b>	View log entries at run time.	<p>On the <b>SSIS</b> menu, click <b>Log Events</b>.</p> <p>-or-</p> <p>Right-click anywhere in the design surface of the <b>Control Flow</b> and <b>Data Flow</b> tabs, and then click <b>Log Events</b>.</p> <p>-or-</p> <p>On the <b>View</b> menu, point to <b>Other Windows</b>, and then click <b>Log Events</b>.</p>

## Custom Editors

Integration Services provides a custom dialog box for most containers, tasks, sources, transformations, and destinations.

The following table describes how to access custom dialog boxes.

EDITOR TYPE	ACCESS
Container. For more information, see <a href="#">Integration Services Containers</a> .	On the design surface of the <b>Control Flow</b> tab, double-click the container.
Task. For more information, see <a href="#">Integration Services Tasks</a> .	On the design surface of the <b>Control Flow</b> tab, double-click the task.
Source.	On the design surface of the <b>Data Flow</b> tab, double-click the source.
Transformation. For more information, see <a href="#">Integration Services Transformations</a> .	On the design surface of the <b>Data Flow</b> tab, double-click the transformation.
Destination.	On the design surface of the <b>Data Flow</b> tab, double-click the destination.

## Advanced Editor

The **Advanced Editor** dialog box is a user interface for configuring data flow components. It reflects the properties of the component using a generic layout. The **Advanced Editor** dialog box is not available to Integration Services transformations that have multiple inputs.

To open this editor, click **ShowAdvanced Editor** in the **Properties** window or right-click a data flow component, and then click **ShowAdvanced Editor**.

If you create a custom source, transformation, or destination but do not want to write a custom user interface, you can use the **Advanced Editor** instead.

## SQL Server Data Tools Features

SQL Server Data Tools (SSDT) provides windows, dialog boxes, and menu options for working with Integration Services packages.

The following is a summary of the available windows and menus:



- The **Solution Explorer** window lists projects, including the Integration Services project in which you develop Integration Services packages, and project files.

To sort by name the packages contained in a project, right-click the **SSIS Packages** node and then click **Sort by name**.

- The **Toolbox** window lists the control flow and data flow items for building control flows and data flows.
- The **Properties** window lists object properties.
- The **Format** menu provides options for sizing and aligning controls in a package.
- The **Edit** menu provides copy and paste functionality for copying objects on the design surfaces.
- The **View** menu provides options for modifying the graphical representation of objects in SSIS Designer

For more information about additional windows and menus, see the Visual Studio documentation.

## Related Tasks

For information about how to create packages in SQL Server Data Tools (SSDT), see [Create Packages in SQL Server Data Tools](#)

## See Also

[SSIS Designer](#)

# SSIS Designer

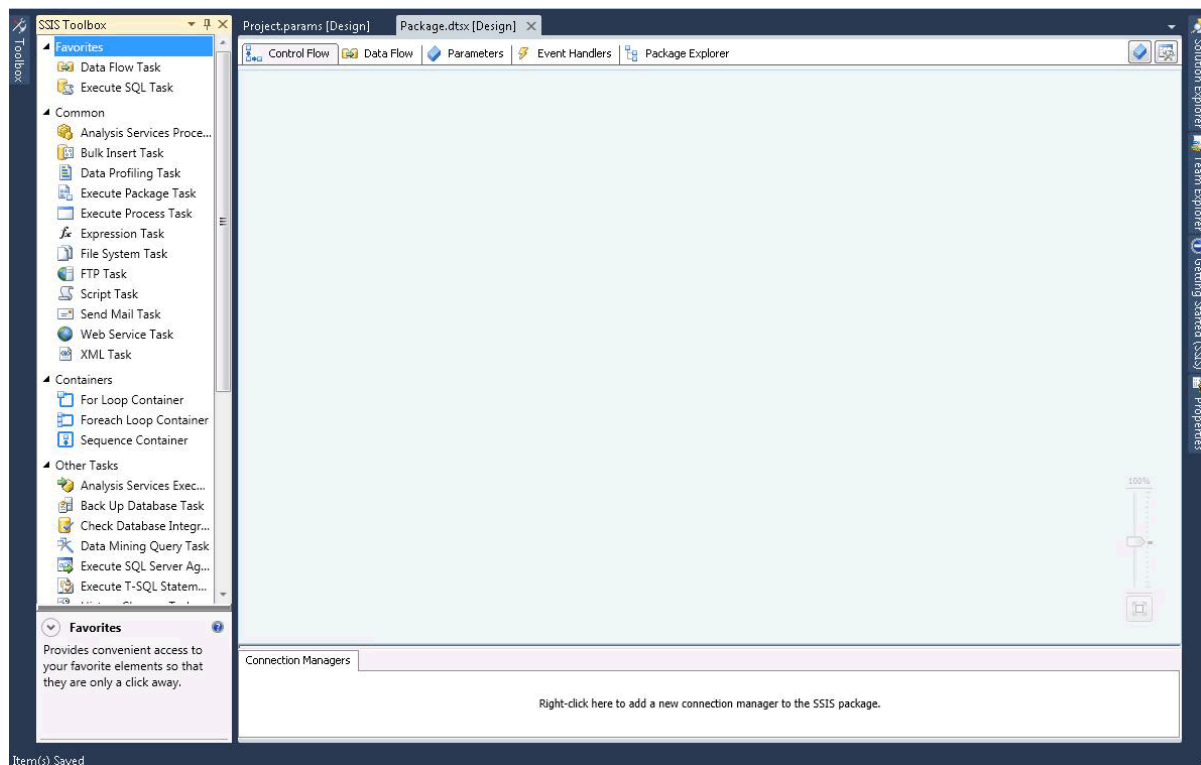
3/24/2017 • 8 min to read • [Edit Online](#)

SSIS Designer is a graphical tool that you can use to create and maintain Integration Services packages. SSIS Designer is available in SQL Server Data Tools (SSDT) as part of an Integration Services project.

You can use SSIS Designer to perform the following tasks:

- Constructing the control flow in a package.
- Constructing the data flows in a package.
- Adding event handlers to the package and package objects.
- Viewing the package content.
- At run time, viewing the execution progress of the package.

The following diagram shows SSIS Designer and the **Toolbox** window.



Integration Services includes additional dialog boxes and windows for adding functionality to packages, and SQL Server Data Tools (SSDT) provides windows and dialog boxes for configuring the development environment and working with packages. For more information, see [Integration Services User Interface](#).

SSIS Designer has no dependency on the Integration Services service, the service that manages and monitors packages, and it is not required that the service be running to create or modify packages in SSIS Designer. However, if you stop the service while SSIS Designer is open, you can no longer open the dialog boxes that SSIS Designer provides and you may receive the error message "RPC server is unavailable." To reset SSIS Designer and continue working with the package, you must close the designer, exit SQL Server Data Tools (SSDT), and then reopen SQL Server Data Tools (SSDT), the Integration Services project, and the package.

# Undo and Redo

You can undo and redo up to 20 actions in the SSIS Designer. For a package, undo /redo is available in the **Control Flow**, **Data Flow**, **Event Handlers**, and **Parameters** tabs, and in the **Variables** window. For a project, undo/redo is available in the **Project Parameters** window.

You can't undo/redo changes to the new **SSIS Toolbox**.

When you make changes to a component using the component editor, you undo and redo the changes as a set rather than undoing and redoing individual changes. The set of changes appears as a single action in the undo and redo drop-down list.

To undo an action, click the undo toolbar button, **Edit/Undo** menu item, or press CTRL+Z. To redo an action, click the redo toolbar button, **Edit/Redo** menu item or press CTRL + Y. You can undo and redo multiple actions, by clicking the arrow next to the toolbar button, highlighting multiple actions in the drop-down list, and then clicking in the list.

## Parts of the SSIS Designer

SSIS Designer has five permanent tabs: one each for building package control flow, data flows, parameters, and event handlers, and one tab for viewing the contents of a package. At run time a sixth tab appears that shows the execution progress of a package while it is running and the execution results after it finishes.

In addition, SSIS Designer includes the Connection Managers area for adding and configuring the connection managers that a package uses to connect to data.

### Control Flow Tab

You construct the control flow in a package on the design surface of the **Control Flow** tab. Drag items from **Toolbox** to the design surface and connect them into a control flow by clicking the icon for the item, and then dragging the arrow from one item to another.

For more information, see [Control Flow](#).

### Data Flow Tab

If a package contains a Data flow task, you can add data flows to the package. You construct the data flows in a package on the design surface of the **Data Flow** tab. Drag items from **Toolbox** to the design surface and connect them into a data flow by clicking the icon for the item, and then dragging the arrow from one item to another.

For more information, see [Data Flow](#).

### Parameters Tab

Integration Services (SSIS) parameters allow you to assign values to properties within packages at the time of package execution. You can create project parameters at the project level and package parameters at the package level. Project parameters are used to supply any external input the project receives to one or more packages in the project. Package parameters allow you to modify package execution without having to edit and redeploy the package. This tab allows you to manage package parameters.

For more information about parameters, see [Integration Services \(SSIS\) Parameters](#).

**IMPORTANT!!** Parameters are available only to projects developed for the project deployment model. Therefore, you will see the Parameters tab only for packages that are part of a project configured to use the project deployment model.

### Event Handlers Tab

You construct the events in a package on the design surface of the **Event Handlers** tab. On the **Event Handlers** tab, you select the package or package object that you want to create an event handler for, and then select the

event to associate with the event handler. An event handler has a control flow and optional data flows.

For more information, see [Add an Event Handler to a Package](#).

### Package Explorer Tab

Packages can be complex, including many tasks, connection managers, variables, and other elements. The explorer view of the package lets you see a complete list of package elements.

For more information, see [View Package Objects](#).

### Progress/Execution Result Tab

While a package is running, the **Progress** tab shows the execution progress of the package. After the package has finished running, the execution results remain available on the **Execution Result** tab.

**NOTE:** To enable or disable the display of messages on the **Progress** tab, toggle the **Debug Progress Reporting** option on the **SSIS** menu.

### Connection Managers Area

You add and modify the connection managers that a package uses in the **Connection Managers** area. Integration Services includes connection managers to connect to a variety of data sources, such as text files, OLE DB databases, and .NET providers.

For more information, see [Integration Services \(SSIS\) Connections](#) and [Create Connection Managers](#).

## Control Flow tab

Use the **Control Flow** tab of SSIS Designer to build the control flow in a Integration Services package.

Create the control flow by dragging graphical objects that represent SSIS tasks and containers from the **Toolbox** to the design surface of the **Control Flow** tab, and then connecting the objects by dragging the connector on an object to another object. Each connecting line represents a precedence constraint that specifies the order in which the tasks and containers run

Additionally, you can use SSIS Designer to add the following functionality from the **Control Flow** tab:

- Implement logging
- Create package configurations
- Sign the package with a certificate
- Manage variables
- Add annotations
- Configure breakpoints

To add these functions to individual tasks or containers in SSIS Designer, right-click the object on the design surface, and then select the option.

## Data Flow tab

Use the **Data Flow** tab of SSIS Designer to create data flows in a Integration Services package.

Create the data flows by dragging graphical objects that represent sources, transformations, and destinations from the **Toolbox** to the design surface of the **Data Flow** tab, and then connecting the objects to create paths that determine the sequence in which the transformations run.

Right-click a path, and then click **Data Viewers**, to add data viewers to view the data before and after each data

flow object.

You can also use SSIS Designer to add the following functionality from the **Data Flow** tab:

- Manage variables
- Add annotations

To add these functions in SSIS Designer, right-click the design surface, and then select the option you want.

## Event Handlers tab

Use the **Event Handlers** tab of SSIS Designer to build a control flow in an Integration Services package. An event handler runs in response to an event raised by the package or by a task or container in the package.

## Options

### Executable

Select the executable for which you want to build an event handler. The executable can be the package, or a task or containers in the package.

### Event handler

Select a type of event handler. Create the event handler by dragging items from the **Toolbox**.

### Delete

Select an event handler, and remove it from the package by clicking **Delete**.

**Click here to create an <event handler name> for the executable <executable name>**

Click to create the event handler.

Create the control flow by dragging graphical objects that represent SSIS tasks and containers from the **Toolbox** to the design surface of the **Event Handlers** tab, and then connecting the objects by using precedence constraints to define the sequence in which they run.

Additionally, to add annotations, right-click the design surface, and then on the menu, click **Add Annotation**.

## Package Explorer tab

Use the **Package Explorer** tab of SSIS Designer to see a hierarchical view of all of the elements in a package: configurations, connections, event handlers, executable objects such as tasks and containers, log providers, precedence constraints, and variables. If a package contains a Data Flow task, the **Package Explorer** tab includes a node that contains a hierarchical view of the data flow components.

Right-click a package element, and then click **Properties** to show the properties of the element in the **Properties** window, or click **Delete** to delete the element.

## Progress tab

Use the **Progress** tab of SSIS Designer to view the progress of execution of an Integration Services package when you run the package in SQL Server Data Tools (SSDT). The **Progress** tab lists the start time, the finish time, and the elapsed time for validation and execution of the package and its executables; any information or warnings for the package; progress notifications; the success or failure of the package; and any error messages that are generated during package execution.

To enable or disable the display of messages on the **Progress** tab, toggle the **Debug Progress Reporting** option on the **SSIS** menu. Disabling progress reporting can help improve performance while running a complex package in SQL Server Data Tools.

After the package stops running, the **Progress** tab becomes the **Execution Results** tab.

## Connection Managers area

Packages use connection managers to connect to data sources such as files, relational databases, and servers.

Use the **Connections Managers** area of SSIS Designer to add, delete, modify, rename, and copy and paste the connection managers.

Right-click in this area, and then on the menu, click the option for the task you want to perform.

## Related Tasks

- [Create Packages in SQL Server Data Tools](#)

## See Also

[Integration Services User Interface](#)

# Advanced Editor

3/24/2017 • 1 min to read • [Edit Online](#)

Use the **Advanced Editor** dialog box to configure to configure properties for the selected Integration Services object.

The **Advanced Editor** is available for most Integration Services objects that have configurable properties. It is the only editor available for those objects that do not expose a custom user interface.

Integration Services data flow objects have properties that can be set at the component level, the input and output level, and the input and output column level. The **Advanced Editor** enumerates all the common and custom properties of the selected object and displays them on up to four of the following five tabs as applicable:

- **Connection Managers** -- use this tab to set connection properties
- **Component Properties** -- use this tab to set component-level properties
- **Column Mappings** -- use this tab to map available columns as output columns
- **Input Columns** -- use this tab to select input columns
- **Input and Output Properties** -- use this tab to set input and output properties; and to add and remove outputs, select or remove columns for inputs and outputs, and set properties for inputs and outputs

The properties displayed vary by component. For more information on the properties that may be displayed in the **Advanced Editor**, see the following topics:

- [Common Properties](#)
- [Transformation Custom Properties](#)
- [Path Properties](#)

For more information about the specific component that you are editing, see the description of the component in the Data Flow Elements section of the Integration Services Objects and Concepts documentation:

- [Integration Services Transformations](#)

## See Also

[Integration Services Error and Message Reference](#)

# Group or Ungroup Components

3/24/2017 • 1 min to read • [Edit Online](#)

The **Control Flow**, **Data Flow**, and **Event Handlers** tabs in SSIS Designer supports collapsible grouping. If a package has many components, the tabs can become crowded, making it difficult to view all the components at one time and to locate the item with which you want to work. The collapsible grouping feature can conserve space on the work surface and make it easier to work with large packages.

You select the components that you want to group, group them, and then expand or collapse the groups to suit your work. Expanding a group provides access to the properties of the components in the group. The precedence constraints that connect tasks and containers are automatically included in the group.

The following are considerations for grouping components.

- To group components, the control flow, data flow, or event handler must contain more than one component.
- Groups can also be nested, making it possible to create groups within groups. The design surface can implement multiple un-nested groups, but a component can belong to only one group, unless the groups are nested.
- When a package is saved, SSIS Designer saves the grouping, but the grouping has no effect on package execution. The ability to group components is a design-time feature; it does not affect the run-time behavior of the package.

## To group components

1. In SQL Server Data Tools (SSDT), open the Integration Services project that contains the package you want.
2. In Solution Explorer, double-click the package to open it.
3. Click the **Control Flow**, **Data Flow**, or **Event Handlers** tab.
4. On the design surface of the tab, select the components you want to group, right-click a selected component, and then click **Group**.
5. To save the updated package, click **Save Selected Items** on the **File** menu.

## To ungroup components

1. In SQL Server Data Tools (SSDT), open the Integration Services project that contains the package you want.
2. In Solution Explorer, double-click the package to open it.
3. Click the **Control Flow**, **Data Flow**, or **Event Handlers** tab.
4. On the design surface of the tab, select the group that contains the component you want to ungroup, right-click, and then click **Ungroup**.
5. To save the updated package, click **Save Selected Items** on the **File** menu.

## See Also

[Add or Delete a Task or a Container in a Control Flow](#)

[Connect Tasks and Containers by Using a Default Precedence Constraint](#)



# Use Annotations in Packages

3/24/2017 • 1 min to read • [Edit Online](#)

The SSIS Designer provides annotations, which you can use to make packages self-documenting and easier to understand and maintain. You can add annotations to the control flow, data flow, and event handler design surfaces of SSIS Designer. The annotations can contain any type of text, and they are useful for adding labels, comments, and other descriptive information to a package. Annotations are a design-time feature only. For example, they are not written to logs.

When you press ENTER, the text wraps to the next line. The annotation box automatically increases in size as you add additional lines of text. Package annotations are persisted as clear text in the CDATA section of the package file.

For more information about changes to the format of the package file, see [SSIS Package Format](#).

When you save the package, SSIS Designer saves the annotations in the package.

## Add an annotation to a package

1. In SQL Server Data Tools (SSDT), open the Integration Services project that contains the package to which you want to add an annotation.
2. In Solution Explorer, double-click the package to open it.
3. In SSIS Designer, right-click anywhere on the design surface of the **Control Flow**, **Data Flow**, or **Event Handler** tab, and then click **Add Annotation**. A text block appears on the design surface of the tab.
4. Add text.

### NOTE

If you add no text, the text block is removed when you click outside the block.

5. To change the size or format of the text in the annotation, right-click the annotation and then click **Set Text Annotation Font**.
6. To add an additional line of text, press ENTER.  
The annotation box automatically increases in size as you add additional lines of text.
7. To add an annotation to a group, right-click the annotation and then click **Group**.
8. To save the updated package, on the **File** menu, click **Save All**.

# SSIS Toolbox

3/24/2017 • 1 min to read • [Edit Online](#)

All components installed on the local machine automatically appear in the **SSIS Toolbox**. When you install additional components, right-click inside the toolbox and then click **Refresh Toolbox** to add the components.

When you create a new SSIS project or open an existing project, the **SSIS Toolbox** displays automatically. You can also open the toolbox by clicking the toolbox button that is located in the top-right corner of the package design surface, or by clicking VIEW -> Other Windows -> SSIS Toolbox.

## NOTE

If you can't see the toolbox, go to VIEW -> Other Windows -> SSIS Toolbox.

Get more information about a component in the toolbox by clicking the component to view its description at the bottom of the toolbox. For some components you can also access samples that demonstrate how to configure and use the components. The samples are available on [MSDN](#). To access the samples from the **SSIS Toolbox**, click the **Find Samples** link that appears below the description.

## NOTE

You can't *remove* installed components from the toolbox.

## Toolbox categories

In the **SSIS Toolbox**, control flow and data flow components are organized into categories. You can expand and collapse categories, and rearrange components. Restore the default organization by right-clicking inside the toolbox and then click **Restore Toolbox Defaults**.

The **Favorites** and **Common** categories appear in the toolbox when you select the **Control Flow**, **Data Flow**, and **Event Handlers** tabs. The **Other Tasks** category appears in the toolbox when you select the **Control Flow** tab or the **Event Handlers** tab. The **Other Transforms**, **Other Sources**, and **Other Destinations** categories appear in the toolbox when you select the **Data Flow** tab.

## Add Azure components to the Toolbox

The Azure Feature Pack for Integration Services contains connection managers to connect to Azure data sources and tasks to do common Azure operations. Install the Feature Pack to add these items to the Toolbox. For more info, see [Azure Feature Pack for Integration Services \(SSIS\)](#).

## Move a Toolbox item to another category

1. Right-click an item in the SSIS Toolbox, and then click one of the following:

- **Move to Favorites**
- **Move to Common**
- **Move to Other Sources**
- **Move to Other Destinations**

- **Move to Other Transforms**

- **Move to Other Tasks**

## Refresh the SSIS Toolbox

1. Right-click in the SSIS Toolbox, and then click **Refresh Toolbox**.

# General Page of Integration Services Designers Options

4/25/2017 • 1 min to read • [Edit Online](#)

Use the **General** page of the **Integration Services Designers** page in the **Options** dialog box to specify the options for loading, displaying, and upgrading packages.

To open the **General** page, in SQL Server Data Tools (SSDT), on the **Tools** menu, click **Options**, expand **Business Intelligence Designers**, and select **Integration Services Designers**.

## Options

### Check digital signature when loading a package

Select to have Integration Services check the digital signature when loading a package. Integration Services will only check whether the digital signature is present, is valid, and is from a trusted source. Integration Services will not check whether the package has been changed since the package was signed.

If you set the **BlockedSignatureStates** registry value, this registry value overrides the **Check digital signature when loading a package** option. For more information, see [Implement a Signing Policy by Setting a Registry Value](#).

For more information about digital certificates and packages, see [Identify the Source of Packages with Digital Signatures](#).

### Show warning if package is unsigned

Select to display a warning when loading a package that is not signed.

### Show precedence constraint labels

Select which label—Success, Failure, or Completion—to display on precedence constraints when viewing packages in SQL Server Data Tools (SSDT).

### Scripting language

Select the default scripting language for new Script tasks and Script components.

### Update connection strings to use new provider names

When opening or upgrading SQL Server 2005 Integration Services (SSIS) packages, update connection strings to use the names for the following providers, for the current release of SQL Server Integration Services:

- Analysis Services OLE DB provider
- SQL Server Native Client

The SSIS Package Upgrade Wizard updates only connection strings that are stored in connection managers. The wizard does not update connection strings that are constructed dynamically by using the Integration Services expression language, or by using code in a Script task.

### Create new package ID

When upgrading SQL Server 2005 Integration Services (SSIS) packages, create new package IDs for the upgraded versions of the packages.

## See Also

[Security Overview \(Integration Services\)](#)



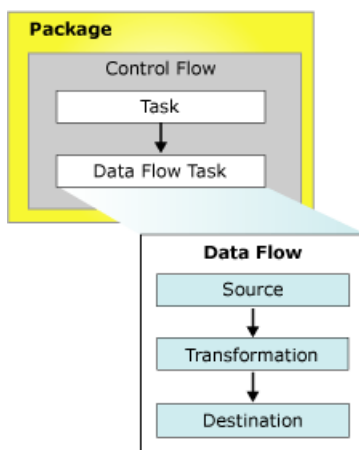
# Integration Services (SSIS) Packages

4/25/2017 • 8 min to read • [Edit Online](#)

A package is an organized collection of connections, control flow elements, data flow elements, event handlers, variables, parameters, and configurations, that you assemble using either the graphical design tools that SQL Server Integration Services provides, or build programmatically. You then save the completed package to SQL Server, the SSIS Package Store, or the file system, or you can deploy the ssISnoversion project to the SSIS server. The package is the unit of work that is retrieved, executed, and saved.

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

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



After you have created the basic package, you can add advanced features such as logging and variables to extend package functionality. For more information, see the section about Objects that Extend Package Functionality.

The completed package can then be configured by setting package-level properties that implement security, enable restarting of packages from checkpoints, or incorporate transactions in package workflow. For more information, see the section about Properties that Support Extended Features.

## Contents of a package

**Tasks and containers (control flow).** A control flow consists of one or more tasks and containers that execute when the package runs. To control order or define the conditions for running the next task or container in the package control flow, you use precedence constraints to connect the tasks and containers in a package. A subset of tasks and containers can also be grouped and run repeatedly as a unit within the package control flow. For more information, see [Control Flow](#).

**Data sources and destinations (data flow).** A data flow consists of the sources and destinations that extract and load data, the transformations that modify and extend data, and the paths that link sources, transformations, and destinations. Before you can add a data flow to a package, the package control flow must include a Data Flow task. The Data Flow task is the executable within the SSIS package that creates, orders, and runs the data flow. A separate instance of the data flow engine is opened for each Data Flow task in a package. For more information, see [Data Flow Task](#) and [Data Flow](#).

**Connection managers (connections).** A package typically includes at least one connection manager. A connection manager is a link between a package and a data source that defines the connection string for accessing

the data that the tasks, transformations, and event handlers in the package use. Integration Services includes connection types for data sources such as text and XML files, relational databases, and Analysis Services databases and projects. For more information, see [Integration Services \(SSIS\) Connections](#).

## Objects that extend package functionality

Packages can include additional objects that provide advanced features or extend existing functionality, such as event handlers, configurations, logging, and variables.

### Event Handlers

An event handler is a workflow that runs in response to the events raised by a package, task, or container. For example, you could use an event handler to check disk space when a pre-execution event occurs or if an error occurs, and send an e-mail message that reports the available space or error information to an administrator. An event handler is constructed like a package, with a control flow and optional data flows. Event handlers can be added to individual tasks or containers in the package. For more information, see [Integration Services \(SSIS\) Event Handlers](#).

### Configurations

A configuration is a set of property-value pairs that defines the properties of the package and its tasks, containers, variables, connections, and event handlers when the package runs. Using configurations makes it possible to update properties without modifying the package. When the package is run, the configuration information is loaded, updating the values of properties. For example, a configuration can update the connection string of connection.

The configuration is saved and then deployed with the package when the package is installed on a different computer. The values in the configuration can be updated when the package is installed to support the package in a different environment. For more information, see [Create Package Configurations](#).

### Logging and Log Providers

A log is a collection of information about the package that is collected when the package runs. For example, a log can provide the start and finish time for a package run. A log provider defines the destination type and the format that the package and its containers and tasks can use to log run-time information. The logs are associated with a package, but the tasks and containers in the package can log information to any package log. Integration Services includes a variety of built-in log providers for logging. For example, Integration Services includes log providers for SQL Server and text files. You can also create custom log providers and use them for logging. For more information, see [Integration Services \(SSIS\) Logging](#).

### Variables

Integration Services supports system variables and user-defined variables. The system variables provide useful information about package objects at run time, and user-defined variables support custom scenarios in packages. Both types of variables can be used in expressions, scripts, and configurations.

The package-level variables include the pre-defined system variables available to a package and the user-defined variables with package scope. For more information, see [Integration Services \(SSIS\) Variables](#).

### Parameters

Integration Services parameters allow you to assign values to properties within packages at the time of package execution. You can create *project parameters* at the project level and *package parameters* at the package level. Project parameters are used to supply any external input the project receives to one or more packages in the project. Package parameters allow you to modify package execution without having to edit and redeploy the package. For more information, see [Integration Services \(SSIS\) Parameters](#).

## Package properties that support extended features

The package object can be configured to support features such as restarting the package at checkpoints, signing the package with a digital certificate, setting the package protection level, and ensuring data integrity by using transactions.

### Restarting Packages

The package includes checkpoint properties that you can use to restart the package when one or more of its tasks fail. For example, if a package has two Data Flow tasks that update two different tables and the second task fails, the package can be rerun without repeating the first Data Flow task. Restarting a package can save time for long-running packages. Restarting means you can start the package from the failed task instead of having to rerun the whole package. For more information, see [Restart Packages by Using Checkpoints](#).

### Securing Packages

A package can be signed with a digital signature and encrypted by using a password or a user key. A digital signature authenticates the source of the package. However, you must also configure Integration Services to check the digital signature when the package loads. For more information, see [Identify the Source of Packages with Digital Signatures](#) and [Access Control for Sensitive Data in Packages](#).

### Supporting Transactions

Setting a transaction attribute on the package enables tasks, containers, and connections in the package to join the transaction. Transaction attributes ensure that the package and its elements succeed or fail as a unit. Packages can also run other packages and enroll other packages in transactions, so that you can run multiple packages as a single unit of work. For more information, see [Integration Services Transactions](#).

## Custom log entries available on the package

The following table lists the custom log entries for packages. For more information, see [Integration Services \(SSIS\) Logging](#).

LOG ENTRY	DESCRIPTION
<b>PackageStart</b>	Indicates that the package began to run.  Note: This log entry is automatically written to the log. You cannot exclude it.
<b>PackageEnd</b>	Indicates that the package completed.  Note: This log entry is automatically written to the log. You cannot exclude it.
<b>Diagnostic</b>	Provides information about the system configuration that affects package execution such as the number executables that can be run concurrently.

## Set the properties of a package

You can set properties in the **Properties** window of SQL Server Data Tools (SSDT) or programmatically.

For information about how to set these properties using SQL Server Data Tools (SSDT), see [Set Package Properties](#).

For information about programmatically setting these properties, see [Package](#).

## Reuse an existing package as a template

Packages are frequently used as templates from which to build packages that share basic functionality. You build the basic package and then copy it, or you can designate the package is a template. For example, a package that



downloads and copies files and then extracts the data may include the FTP and File System tasks in a Foreach Loop that enumerates files in a folder. It may also include Flat File connection managers to access the data, and Flat File sources to extract the data. The destination of the data varies, and the destination is added to each new package after it is copied from the basic package. You can also create packages and then use them as templates for the new packages that you add to an Integration Services project. For more information, see [Create Packages in SQL Server Data Tools](#).

When a package is first created, either programmatically or by using SSIS Designer, a GUID is added to its **ID** property and a name to its **Name** property. If you create a new package by copying an existing package or by using a template package, the name and the GUID are copied as well. This can be a problem if you use logging, because the GUID and the name of the package are written to the logs to identify the package to which the logged information belongs. Therefore, you should update the name and the GUID of the new packages to help differentiate them from the package from which they were copied and from each other in the log data.

To change the package GUID, you regenerate a GUID in the **ID** property in the Properties window in SQL Server Data Tools (SSDT). To change the package name, you can update the value of the **Name** property in the Properties window. You can also use the **dtutil** command prompt, or update the GUID and name programmatically. For more information, see [Set Package Properties](#) and [dtutil Utility](#).

## Related Tasks

Integration Services includes two graphical tools, SSIS Designer and SQL Server Import and Export Wizard, in addition to the SSIS object model for creating packages. See the following topics for details.

- [Import and Export Data with the SQL Server Import and Export Wizard](#)
- [Create Packages in SQL Server Data Tools](#)
- See [Building Packages Programmatically](#) in the Developer Guide.

# Create Packages in SQL Server Data Tools

3/24/2017 • 3 min to read • [Edit Online](#)

In SQL Server Data Tools (SSDT), you can create a new package by using one of the following methods:

- Use the package template that Integration Services includes.
- Use a custom template

To use custom packages as templates for creating new packages, you simply copy them to the DataTransformationItems folder. By default, this folder is in C:\Program Files\Microsoft Visual Studio 10.0\Common7\IDE\PrivateAssemblies\ProjectItems\DataTransformationProject.

- Copy an existing package.

If existing packages include functionality that you want to reuse, you can build the control flow and data flows in the new package more quickly by copying and pasting objects from other packages. For more information about using copy and paste in Integration Services projects, see [Reuse of Package Objects](#).

If you create a new package by copying an existing package or by using a custom package as a template, the name and the GUID of the existing package are copied as well. You should update the name and the GUID of the new package to help differentiate it from the package from which it was copied. For example, if packages have the same GUID, it is more difficult to identify the package to which log data belongs. You can regenerate the GUID in the **ID** property and update the value of the **Name** property by using the Properties window in SQL Server Data Tools (SSDT). For more information, see [Set Package Properties](#) and [dtutil Utility](#).

- Use a custom package that you have designated as a template.
- Run the SQL Server Import and Export Wizard

The SQL Server Import and Export Wizard creates a complete package for a simple import or export. This wizard configures the connections, source, and destination, and adds any data transformations that are required to let you run the import or export immediately. You can optionally save the package to run it again later, or to refine and enhance the package in SQL Server Data Tools. However, if you save the package, you must add the package to an existing Integration Services project before you can change the package or run the package in SQL Server Data Tools.

The packages that you create in SQL Server Data Tools (SSDT) using SSIS Designer are saved to the file system. To save a package to SQL Server or to the package store, you need to save a copy of the package. For more information, see [Save a Copy of a Package](#).

For a video that demonstrates how to create a basic package using the default package template, see [Creating a Basic Package \(SQL Server Video\)](#).

## Get SQL Server Data Tools

To install SQL Server Data Tools (SSDT), see [Download SQL Server Data Tools \(SSDT\)](#).

## Create a package in SQL Server Data Tools using the Package Template

1. In SQL Server Data Tools (SSDT), open the Integration Services project in which you want to create a

package.

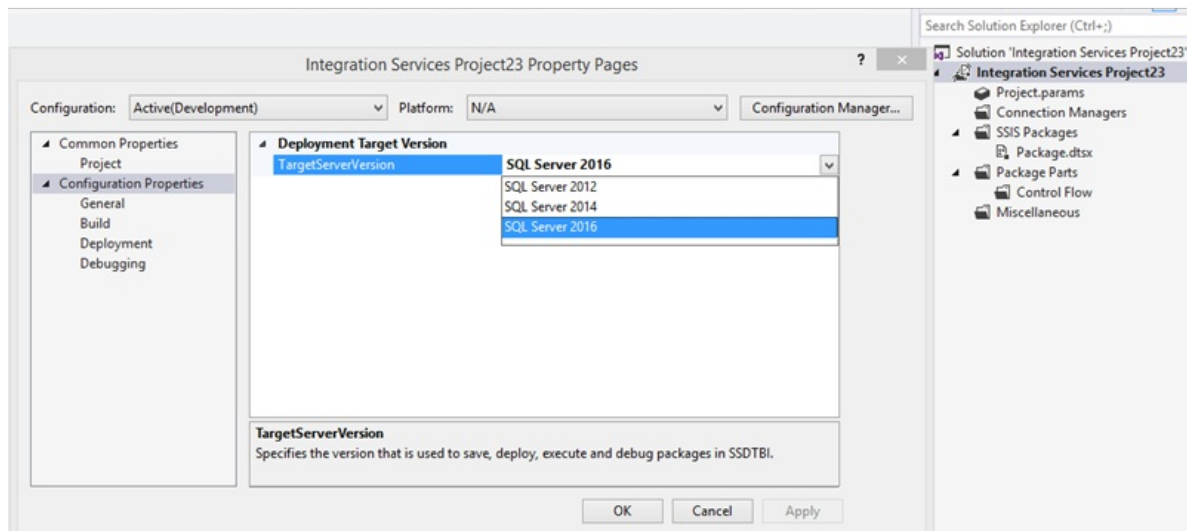
2. In Solution Explorer, right-click the **SSIS Packages** folder, and then click **New SSIS Package**.
3. Optionally, add control flow, data flow tasks, and event handlers to the package. For more information, see [Control Flow](#), [Data Flow](#), and [Integration Services \(SSIS\) Event Handlers](#).
4. On the **File** menu, click **Save Selected Items** to save the new package.

#### NOTE

You can save an empty package.

## Choose the target version of a project and its packages

1. In Solution Explorer, right-click on an Integration Services project and select **Properties** to open the property pages for the project.
2. On the **General** tab of **Configuration Properties**, select the **TargetServerVersion** property, and then choose SQL Server 2016, SQL Server 2014, or SQL Server 2012.



You can create, maintain, and run packages that target SQL Server 2016, SQL Server 2014, or SQL Server 2012.

# Add Copy of Existing Package

4/10/2017 • 1 min to read • [Edit Online](#)

Use the **Add Copy of Existing Package** dialog box to add a copy of a package stored in SQL Server, the file system, or the SSIS Package Store to an Integration Services project.

## Options

### Package location

Select the type of storage location from which to copy the package.

### Server

If copying from SQL Server or the SSIS Package Store, type a server name or select a server from the list.

### Authentication type

If copying from SQL Server, select an authentication type.

### User name

If using SQL Server Authentication, provide a user name.

### Password

If using SQL Server Authentication, provide a password.

### Package path

Type the package path, or click the browse button (...) and locate the package to copy.

## See Also

[Save Copy of Package](#)

[Save Packages](#)

[Integration Services Service \(SSIS Service\)](#)

# Set Package Properties

4/19/2017 • 7 min to read • [Edit Online](#)

When you create a package in SQL Server Data Tools (SSDT) by using the graphical interface that Integration Services provides, you set the properties of the package object in the Properties window.

The **Properties** window provides a categorized and alphabetical list of properties. To arrange the **Properties** window by category, click the Categorized icon.

When arranged by category, the **Properties** window groups properties in the following categories:

- [Checkpoints](#)
- [Execution](#)
- [Forced Execution Value](#)
- [Identification](#)
- [Misc](#)
- [Security](#)
- [Transactions](#)
- [Version](#)

For information about additional package properties that you cannot set in the **Properties** window, see [Package](#).

## To set package properties in the Properties window

- [Set the Properties of a Package](#)

## Properties by Category

The following tables list the package properties by category.

### Checkpoints

You can use the properties in this category to restart the package from a point of failure in the package control flow, instead of rerunning the package from the beginning of its control flow. For more information, see [Restart Packages by Using Checkpoints](#).

PROPERTY	DESCRIPTION
<b>CheckpointFileName</b>	The name of the file that captures the checkpoint information that enables a package to restart. When the package finishes successfully, this file is deleted.
<b>CheckpointUsage</b>	Specifies when a package can be restarted. The values are <b>Never</b> , <b>IfExists</b> , and <b>Always</b> . The default value of this property is <b>Never</b> , which indicates that the package cannot be restarted. For more information, see <a href="#">DTSCheckpointUsage</a> .

PROPERTY	DESCRIPTION
<b>SaveCheckpoints</b>	Specifies whether the checkpoints are written to the checkpoint file when the package runs. The default value of this property is <b>False</b> .

#### NOTE

The **/CheckPointing on** option of dtexec is equivalent to setting the **SaveCheckpoints** property of the package to True, and the **CheckpointUsage** property to Always. For more information, see [dtexec Utility](#).

## Execution

The properties in this category configure the run-time behavior of the package object.

PROPERTY	DESCRIPTION
<b>DelayValidation</b>	Indicates whether package validation is delayed until the package runs. The default value for this property is <b>False</b> .
<b>Disable</b>	Indicates whether the package is disabled. The default value of this property is <b>False</b> .
<b>DisableEventHandlers</b>	Specifies whether the package event handlers run. The default value of this property is <b>False</b> .
<b>FailPackageOnFailure</b>	Specifies whether the package fails if an error occurs in a package component. The only valid value of this property is <b>False</b> .
<b>FailParentOnError</b>	Specifies whether the parent container fails if an error occurs in a child container. The default value of this property is <b>False</b> .
<b>MaxConcurrentExecutables</b>	The number of executable files that the package can run concurrently. The default value of this property is <b>-1</b> , which indicates that there is no limit.
<b>MaximumErrorCount</b>	The maximum number of errors that can occur before a package stops running. The default value of this property is <b>1</b> .
<b>PackagePriorityClass</b>	The Win32 thread priority class of the package thread. The values are <b>Default</b> , <b>AboveNormal</b> , <b>Normal</b> , <b>BelowNormal</b> , <b>Idle</b> . The default value of this property is <b>Default</b> . For more information, see <a href="#">DTSPriorityClass</a> .

## Forced Execution Value

The properties in this category configure an optional execution value for the package.

PROPERTY	DESCRIPTION
<b>ForcedExecutionValue</b>	If ForceExecutionValue is set to <b>True</b> , a value that specifies the optional execution value that the package returns. The default value of this property is <b>0</b> .

PROPERTY	DESCRIPTION
<b>ForcedExecutionValueType</b>	The data type of ForcedExecutionValue. The default value of this property is <b>Int32</b> .
<b>ForceExecutionValue</b>	A Boolean value that specifies whether the optional execution value of the container should be forced to contain a particular value. The default value of this property is <b>False</b> .

## Identification

The properties in this category provide information such as the unique identifier and name of the package.

PROPERTY	DESCRIPTION
<b>CreationDate</b>	The date that the package was created.
<b>CreatorComputerName</b>	The name of the computer on which the package was created.
<b>CreatorName</b>	The name of the person who created the package.
<b>Description</b>	A description of package functionality.
<b>ID</b>	The package GUID, which is assigned when the package is created. This property is read-only. To generate a new random value for the <b>ID</b> property, select <b>&lt;Generate New ID&gt;</b> in the drop-down list.
<b>Name</b>	The name of the package.
<b>PackageType</b>	The package type. The values are <b>Default</b> , <b>DTSDesigner</b> , <b>DTSDesigner100</b> , <b>DTSWizard</b> , <b>SQLDBMaint</b> , and <b>SQLReplication</b> . The default value of this property is <b>Default</b> . For more information, see <a href="#">DTSPackageType</a> .

## Misc

The properties in this category are used to access the configurations and expressions that a package uses and to provide information about the locale and logging mode of the package. For more information, see [Use Property Expressions in Packages](#).

PROPERTY	DESCRIPTION
<b>Configurations</b>	The collection of configurations that the package uses. Click the browse button (...) to view and configure package configurations.

PROPERTY	DESCRIPTION
<b>Expressions</b>	<p>Click the browse button (...) to create expressions for package properties.</p> <p>Note that you can create property expressions for all the package properties that object model includes, not just the properties listed in the Properties window.</p> <p>For more information, see <a href="#">Use Property Expressions in Packages</a>.</p> <p>To view existing property expressions, expand <b>Expressions</b>. Click the browse button (...) in an expression text box to modify and evaluate an expression.</p>
<b>ForceExecutionResult</b>	The execution result of the package. The values are <b>None</b> , <b>Success</b> , <b>Failure</b> , and <b>Completion</b> . The default value of this property is <b>None</b> . For more information, see T:Microsoft.SqlServer.Dts.Runtime.DTSForcedExecResult.
<b>LocaleId</b>	A Microsoft Win32 locale. The default value of this property is the locale of the operating system on the local computer.
<b>LoggingMode</b>	A value that specifies the logging behavior of the package. The values are <b>Disabled</b> , <b>Enabled</b> , and <b>UseParentSetting</b> . The default value of this property is <b>UseParentSetting</b> . For more information, see <a href="#">DTSLoggingMode</a> .
<b>OfflineMode</b>	Indicates whether the package is in offline mode. This property is read-only. The property is set at the project level. Normally, SSIS Designer tries to connect to each data source used by your package to validate the metadata associated with sources and destinations. You can enable <b>Work Offline</b> from the <b>SSIS</b> menu, even before you open a package, to prevent these connection attempts and the resulting validation errors when the data sources are not available. You can also enable <b>Work Offline</b> to speed up operations in the designer, and disable it only when you want your package to be validated.
<b>SuppressConfigurationWarnings</b>	Indicates whether the warnings generated by configurations are suppressed. The default value of this property is <b>False</b> .
<b>UpdateObjects</b>	Indicates whether the package is updated to use newer versions of the objects it contains, if newer versions are available. For example, if this property is set to <b>True</b> , a package that includes a Bulk Insert task is updated to use the newer version of the Bulk Insert task that Integration Services provides. The default value of this property is <b>False</b> .

## Security

The properties in this category are used to set the protection level of the package. For more information, see [Access Control for Sensitive Data in Packages](#).

PROPERTY	DESCRIPTION
----------	-------------



PROPERTY	DESCRIPTION
<b>PackagePassword</b>	The password for package protection levels ( <b>EncryptSensitiveWithPassword</b> and <b>EncryptAllWithPassword</b> ) that require passwords.
<b>ProtectionLevel</b>	The protection level of the package. The values are <b>DontSaveSensitive</b> , <b>EncryptSensitiveWithUserKey</b> , <b>EncryptSensitiveWithPassword</b> , <b>EncryptAllWithPassword</b> , and <b>ServerStorage</b> . The default value of this property is <b>EncryptSensitiveWithUserKey</b> . For more information, see <a href="#">DTSProtectionLevel</a> .

## Transactions

The properties in this category configure the isolation level and the transaction option of the package. For more information, see [Integration Services Transactions](#).

PROPERTY	DESCRIPTION
<b>IsolationLevel</b>	<p>The isolation level of the package transaction. The values are <b>Unspecified</b>, <b>Chaos</b>, <b>ReadUncommitted</b>, <b>ReadCommitted</b>, <b>RepeatableRead</b>, <b>Serializable</b>, and <b>Snapshot</b>. The default value of this property is <b>Serializable</b>.</p> <p>Note: The <b>Snapshot</b> value of the <b>IsolationLevel</b> property is incompatible with package transactions. Therefore, you cannot use the <b>IsolationLevel</b> property to set the isolation level of package transactions to <b>Snapshot</b>. Instead, use an SQL query to set package transactions to <b>Snapshot</b>. For more information, see <a href="#">SET TRANSACTION ISOLATION LEVEL (Transact-SQL)</a>.</p> <p>The system applies the <b>IsolationLevel</b> property to package transactions only when the value of the <b>TransactionOption</b> property is <b>Required</b>.</p> <p>The value of the <b>IsolationLevel</b> property requested by a child container is ignored when the following conditions are true:  The value of the child container's <b>TransactionOption</b> property is <b>Supported</b>.  The child container joins the transaction of a parent container.</p> <p>The value of the <b>IsolationLevel</b> property requested by the container is respected only when the container initiates a new transaction. A container initiates a new transaction when the following conditions are true:  The value of the container's <b>TransactionOption</b> property is <b>Required</b>.  The parent has not already started a transaction.</p> <p>For more information, see <a href="#">IsolationLevel</a>.</p>
<b>TransactionOption</b>	The transactional participation of the package. The values are <b>NotSupported</b> , <b>Supported</b> , <b>Required</b> . The default value of this property is <b>Supported</b> . For more information, see <a href="#">DTSTransactionOption</a> .

## Version

The properties in this category provide information about the version of the package object.

PROPERTY	DESCRIPTION
<b>VersionBuild</b>	The version number of the build of the package.
<b>VersionComments</b>	Comments about the version of the package.
<b>VersionGUID</b>	The GUID of the version of the package. This property is read-only.
<b>VersionMajor</b>	The latest major version of the package.
<b>VersionMinor</b>	The latest minor version of the package.

## Set package properties in the Properties window

1. In SQL Server Data Tools (SSDT), open the Integration Services project that contains the package you want to configure.
2. In **Solution Explorer**, double-click the package to open it in SSIS Designer, or right-click and select **View Designer**.
3. Click the **Control Flow** tab and then do one of the following:
  - Right-click anywhere in the background of the control flow design surface, and then click **Properties**.
  - On the **View** menu, click **Properties Window**.
4. Edit the package properties in the **Properties** window.
5. On the **File** menu, click **Save Selected Items** to save the updated package.

# View Package Objects

3/24/2017 • 1 min to read • [Edit Online](#)

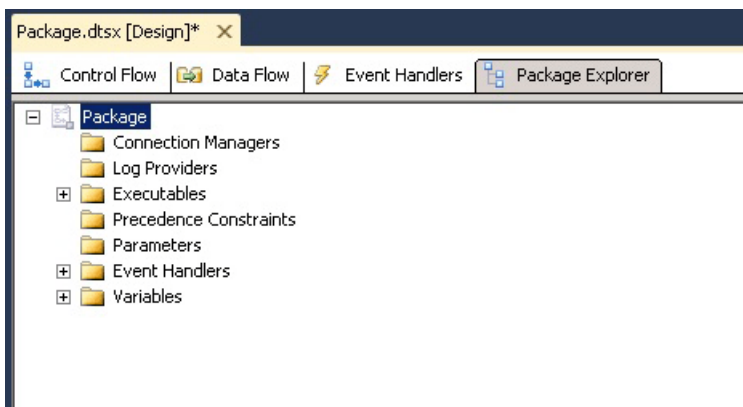
In SSIS Designer, the **Package Explorer** tab provides an explorer view of the package. The view reflects the container hierarchy of the Integration Services architecture. The package container is at the top of the hierarchy, and you expand the package to view the connections, executables, event handlers, log providers, precedence constraints, and variables in the package.

The executables, which are the containers and tasks in the package, can include event handlers, precedence constraints, and variables. Integration Services supports a nested hierarchy of containers, and the For Loop, Foreach Loop, and Sequence containers can include other executables.

If a package includes a data flow, the **Package Explorer** lists the Data Flow task and includes a **Components** folder that lists the data flow components.

From the **Package Explorer** tab, you can delete objects in a package and access the **Properties** window to view object properties.

The following diagram shows a tree view of a simple package.



## View the package structure and content

1. In SQL Server Data Tools (SSDT), open the Integration Services project that contains the package you want to view in **Package Explorer**.
2. Click the **Package Explorer** tab.
3. To view the contents of the **Variables**, **Precedence Constraints**, **Event Handlers**, **Connection Managers**, **Log Providers**, or **Executables** folders, expand each folder.
4. Depending on the package structure, expand any next-level folders.

## View the properties of a package object

- Right-click an object and then click **Properties** to open the **Properties** window.

## Delete an object in a package

- Right-click an object and then click **Delete**.

## See Also

[Integration Services Tasks](#)

[Integration Services Containers](#)

[Precedence Constraints](#)

[Integration Services \(SSIS\) Variables](#)

[Integration Services \(SSIS\) Event Handlers](#)

[Integration Services \(SSIS\) Logging](#)

# Copy a Package in SQL Server Data Tools

3/24/2017 • 1 min to read • [Edit Online](#)

This topic describes how to create a new Integration Services package by copying an existing package, and how to update the **Name** and **GUID** properties of the new package.

## To copy a package

1. In SQL Server Data Tools (SSDT), open the Integration Services project that contains the package that you want to copy.
2. In Solution Explorer, double-click the package.
3. Verify either the package to copy is selected in Solution Explorer or the tab in SSIS Designer that contains the package is the active tab
4. On the **File** menu, click **Save <package name> As**.

### NOTE

The package must be opened in SSIS Designer before the **Save As** option appears on the **File** menu.

5. Optionally, browse to a different folder.
6. Update the name of the package file. Make sure that you retain the .dtsx file extension.
7. Click **Save**.
8. At the prompt, choose whether to update the name of the package object to match the file name. If you click **Yes**, the **Name** property of the package is updated. The new package is added to the Integration Services project and opened in SSIS Designer.
9. Optionally, click in the background of the **Control Flow** tab, and then click **Properties**.
10. In the Properties window, click the value of the ID property, and then in the drop-down list click **<Generate New ID>**.
11. On the **File** menu, click **Save Selected Items** to save the new package.

## See Also

[Save a Copy of a Package](#)

[Create Packages in SQL Server Data Tools](#)

[Integration Services \(SSIS\) Packages](#)

# Copy Package Objects

3/24/2017 • 1 min to read • [Edit Online](#)

This topic describes how to copy control flow items, data flow items, and connection managers within a package or between packages.

## To copy control and data flow items

1. In SQL Server Data Tools (SSDT), open the Integration Services project that contains the packages that you want work with.
2. In Solution Explorer, double-click the packages that you want to copy between.
3. In SSIS Designer, click the tab for the package that contains the items to copy and click the **Control Flow**, **Data Flow**, or **Event Handlers** tab.
4. Select the control flow or data flow items to copy. You can either select items one at a time by pressing the Shift key and clicking the item or select items as a group by dragging the pointer across the items you want to select.

### IMPORTANT

The precedence constraints and paths that connect items are not selected automatically when you select the two items that they connect. To copy an ordered workflow—a segment of control flow or data flow—make sure to also copy the precedence constraints and the paths.

5. Right-click a selected item and click **Copy**.
6. If copying items to a different package, click the package that you want to copy to, and then click the appropriate tab for the item type.

### IMPORTANT

You cannot copy a data flow to a package unless the package contains at least one Data Flow task.

7. Right-click and click **Paste**.

## To copy connection managers

1. In SQL Server Data Tools (SSDT), open the Integration Services project that contains the package that you want to work with.
2. In Solution Explorer, double-click the package.
3. In SSIS Designer, click the **Control Flow**, **Data Flow**, or **Event Handler** tab.
4. In the **Connection Managers** area, right-click the connection manager, and then click **Copy**. You can copy only one connection manager at a time.
5. If you are copying items to a different package, click the package that you want to copy to and then click the **Control Flow**, **Data Flow**, or **Event Handler** tab.
6. Right-click in the **Connection Managers** area and click **Paste**.

## See Also

[Control Flow](#)

[Data Flow](#)

[Integration Services \(SSIS\) Connections](#)

[Copy Project Items](#)

# Save Packages

4/10/2017 • 5 min to read • [Edit Online](#)

In SQL Server Data Tools (SSDT) you build packages by using SSIS Designer and save the packages to the file system as XML files (.dtsx files). You can also save copies of the package XML file to the msdb database in SQL Server or to the package store. The package store represents the folders in the file system location that the Integration Services service manages.

If you save a package to the file system, you can later use the Integration Services service to import the package to SQL Server or to the package store. For more information, see [Integration Services Service \(SSIS Service\)](#).

You can also use a command prompt utility, **dtutil**, to copy a package between the file system and msdb. For more information, see [dtutil Utility](#).

## Save a package to the file system

1. In SQL Server Data Tools (SSDT), open the Integration Services project that contains the package you want to save to a file.
2. In Solution Explorer, click the package you want to save.
3. On the **File** menu, click **Save Selected Items**.

### NOTE

You can verify the path and file name where the package was saved in the Properties window.

## Save a copy of a package

This section describes how to save a copy of a package to the file system, to the package store, or to the **msdb** database in Microsoft SQL Server. When you specify a location to save the package copy, you can also update the name of the package.

The package store can include both the **msdb** database and the folders in the file system, only **msdb**, or only folders in the file system. In **msdb**, packages are saved to the **sysssispackages** table. This table includes a **folderid** column that identifies the logical folder to which the package belongs. The logical folders provide a useful way to group packages saved to **msdb** in the same way that folders in the file system provide a way to group packages saved to the file system. Rows in the **sysssispackagefolders** table in **msdb** define the folders.

If **msdb** is not defined as part of the package store, you can continue to associate packages with existing logical folders when you select SQL Server in the **Package Path** option.

### NOTE

The package must be opened in SSIS Designer before you can save a copy of the package.

### To save a copy of a package

1. In Solution Explorer, double-click the package of which you want to save a copy.
2. On the **File** menu, click **Save Copy of <package file> As**.
3. In the **Save Copy of Package** dialog box, select a package location in the **Package location** list. The



following options are available:

- SQL Server
- File System
- SSIS Package Store

4. If the location is **SQL Server** or **SSIS Package Store**, provide a server name.
5. If saving to SQL Server, specify the authentication type and, if using SQL Server Authentication, provide a user name and password.
6. To specify the package path, either type the path or click the browse button (...) to specify the location of the package. The default name of the package is Package. Optionally, update the package name to one that suits your needs.

If you select **SQL Server** as the **Package Path** option, the package path consists of logical folders in **msdb** and the package name. For example, if the package DownloadMonthlyData is associated with the Finance folder within the MSDB folder (the default name of the root logical folder in **msdb**), the package path for the package named DownloadMonthlyData is MSDB/Finance/DownloadMonthlyData

If you select **SSIS Package Store** as the **Package Path** option, the package path consists of the folder that the Integration Services service manages. For example, if the package UpdateDeductions is located in the HumanResources folder within the file system folder that the service manages, the package path is /File System/HumanResources/UpdateDeductions; likewise, if the package PostResumes is associated with the HumanResources folder within the MSDB folder, the package path is MSDB/HumanResources/PostResumes.

If you select **File System** as the **Package Path** option, the package path is the location in the file system and the file name. For example, if the package name is UpdateDemographics the package path is C:\HumanResources\Quarterly\UpdateDemographics.dtsx.

7. Review the package protection level.
8. Optionally, click the browse button (...) by the **Protection level** box to change the protection level.
  - In the **Package Protection Level** dialog box, select a different protection level.
  - Click **OK**.
9. Click **OK**.

## Save a package as a package template

This section describes how to designate and use custom packages as templates when you create new Integration Services packages in SQL Server Data Tools (SSDT). By default, Integration Services uses a package template that creates an empty package when you add a new package to an Integration Services project. You cannot replace this default template, but you can add new templates.

You can designate multiple packages to use as templates. Before you can implement custom packages as templates, you must create the packages.

When you create package using custom packages as templates, the new packages have the same name and GUID as the template. To differentiate among packages you should update the value of the **Name** property and generate a new GUID for the **ID** property. For more information, see [Create Packages in SQL Server Data Tools](#) and [Set Package Properties](#).

### To designate a custom package as a package template

1. In the file system, locate the package that you want to use as template.

2. Copy the package to the DataTransformationItems folder. By default this folder is in C:\Program Files\Microsoft Visual Studio 9.0\Common7\IDE\PrivateAssemblies\ProjectItems\DataTransformationProject.
3. Repeat steps 1 and 2 for each package that you want to use as a template.

#### **To use a custom package as a package template**

1. In SQL Server Data Tools (SSDT), open the Integration Services project in which you want to create a package.
2. In Solution Explorer, right-click the project, point to **Add**, and then click **New Item**.
3. In the **Add New Item -<project name>** dialog box, click the package that you want to use as a template.

The list of templates includes the default package template named New SSIS Package. The package icon identifies templates that can be used as package templates.

4. Click **Add**.

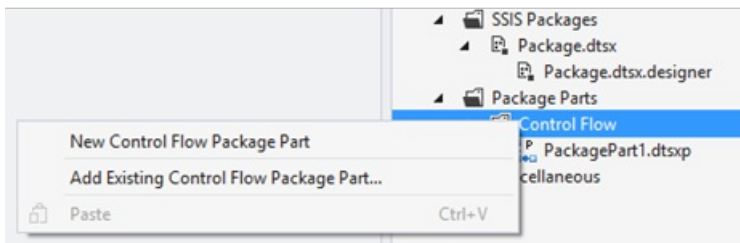
# Reuse Control Flow across Packages by Using Control Flow Package Parts

3/24/2017 • 5 min to read • [Edit Online](#)

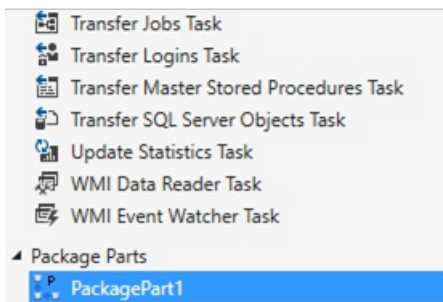
Save a commonly used control flow task or container to a standalone part file - a ".dtsx" file - and reuse it multiple times in one or more packages by using control flow package parts. This reusability makes SSIS packages easier to design and maintain.

## Create a new control flow package part

To create a new control flow package part, in Solution Explorer, expand the **Package Parts** folder. Right-click on **Control Flow** and select **New Control Flow Package Part**.

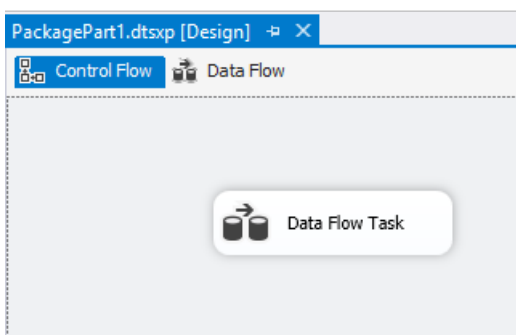


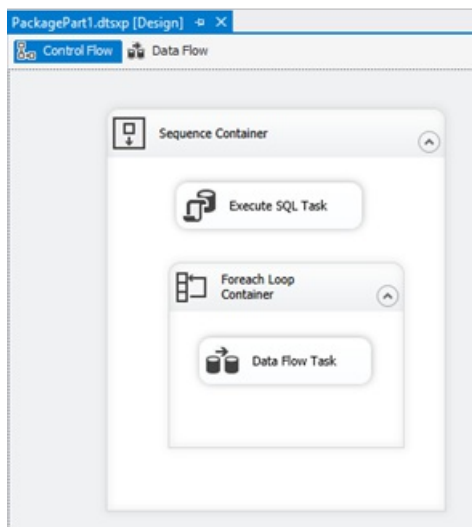
A new part file with the ".dtsx" extension is created under the **Package Parts | Control Flow** folder. At the same time, a new item with the same name is also added to the SSIS toolbox. (The toolbox item is only visible while you have a project that contains the part open in Visual Studio.)



## Design a control flow package part

To open the package part editor, double-click on the part file in Solution Explorer. You can design the part just like you design a package.





Control flow package parts have the following limitations.

- A part can have only one top-level task or container. If you want to include multiple tasks or containers, put them all in a single sequence container.
- You can't run or debug a part directly in the designer.

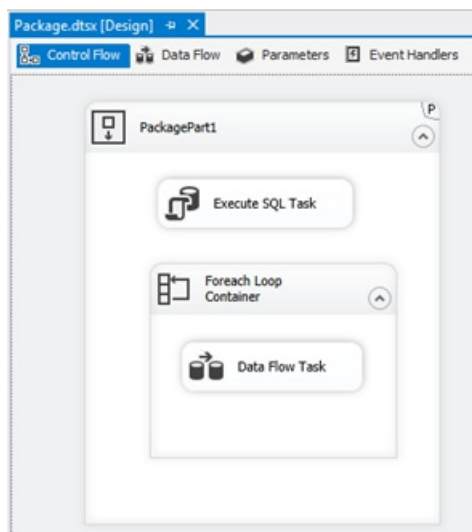
## Add an existing control flow package part to a package

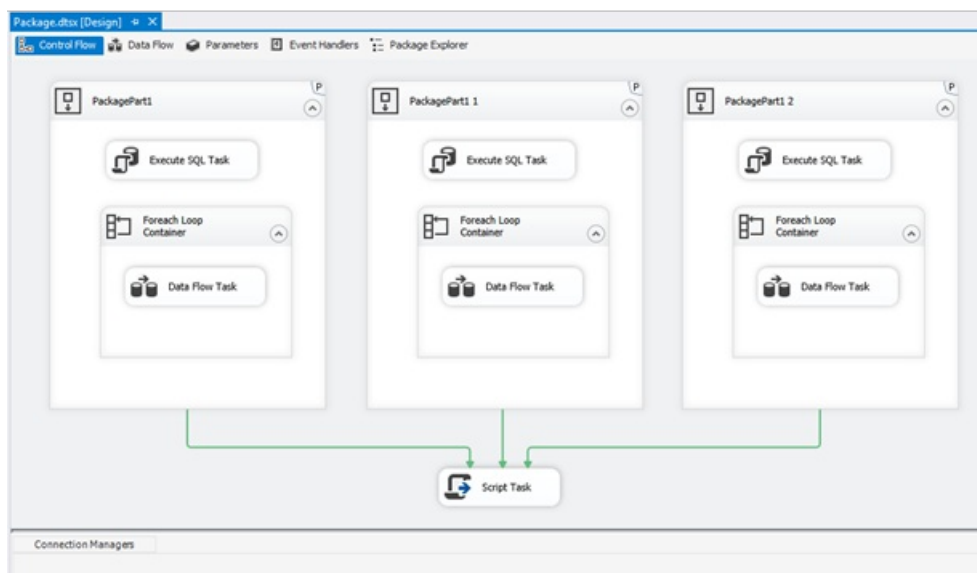
You can reuse parts that are saved in the current Integration Services project or in a different project.

- To reuse a part that's part of the current project, drag and drop the part from the toolbox.
- To reuse a part that's part of a different project, use the **Add Existing Control Flow Package Part** command.

### Drag and drop a control flow package part

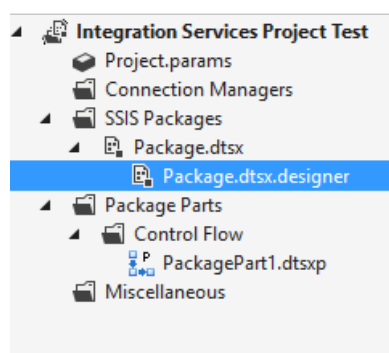
To reuse a part in a project, simply drag and drop the part item from the toolbox just like any other task or container. You can drag and drop the part into a package multiple times to reuse the logic in multiple locations in the package. Use this method to reuse a part that is part of the current project.





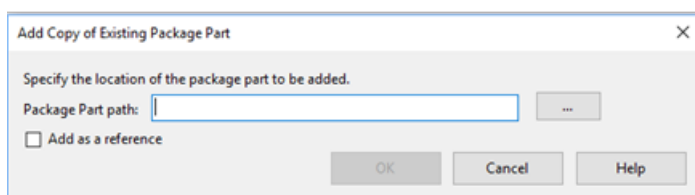
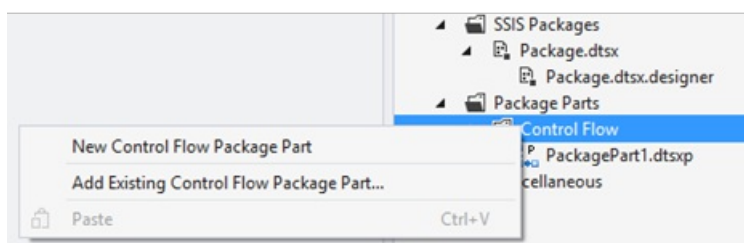
When you save the package, SSIS designer checks whether there are any part instances in the package.

- If the package contains part instances, the designer generates a new .dtsx.designer file which contains all part-related information.
- If the package does not use parts, the designer deletes any previously created .dtsx.designer file for the package (that is, any .dtsx.designer file that has the same name as the package).



### Add a copy of an existing control flow package part or a reference to an existing part

To add a copy of an existing part in the file system to a package, in Solution Explorer, expand the **Package Parts** folder. Right-click on **Control Flow** and select **Add Existing Control Flow Package Part**.



## Options

### Package Part path

Type the path to the part file, or click the browse button (...) and locate the part file to copy or to reference.

### Add as a reference

- If selected, the part is added to the Integration Services project as a reference. Select this option when you want to reference a single copy of a part file in multiple Integration Services projects.
- If cleared, a copy of the part file is added to the project.

## Configure a control flow package part

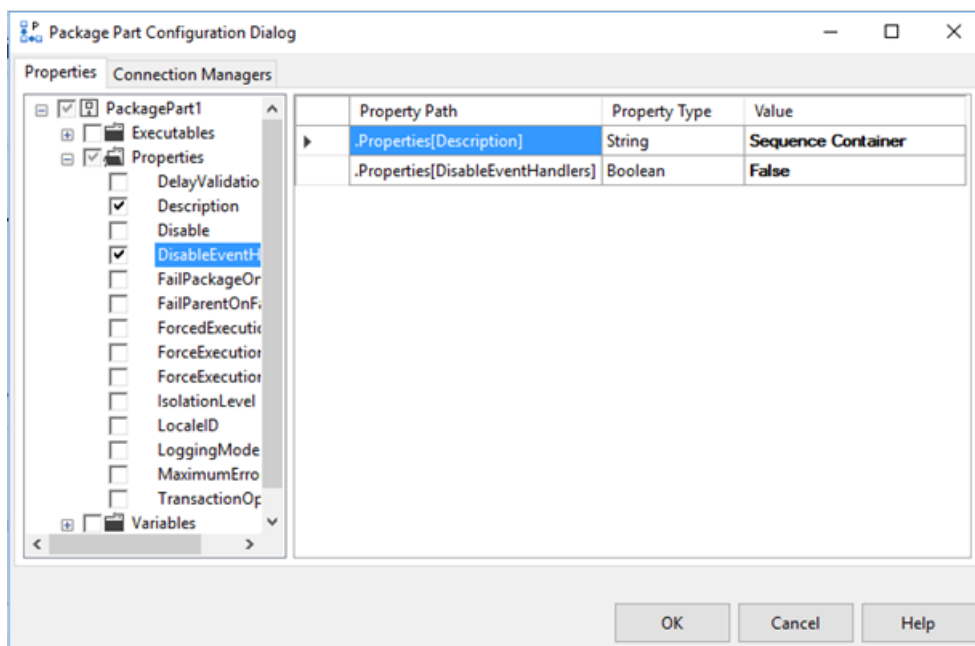
To configure control flow package parts after you've added them to the control flow of a package, use the **Package Part Configuration** dialog box.

### To open the Package Part Configuration dialog box

1. To configure a part instance, double-click the part instance in the control flow. Or right-click on the part instance and select **Edit**. The **Package Part Configuration** dialog box opens.
2. Configure the properties and connection managers for the part instance.

### Properties tab

Use the **Properties** tab of the **Package Part Configuration** dialog box to specify the properties of the part.



The tree view hierarchy in the left pane lists all configurable properties of the part instance.

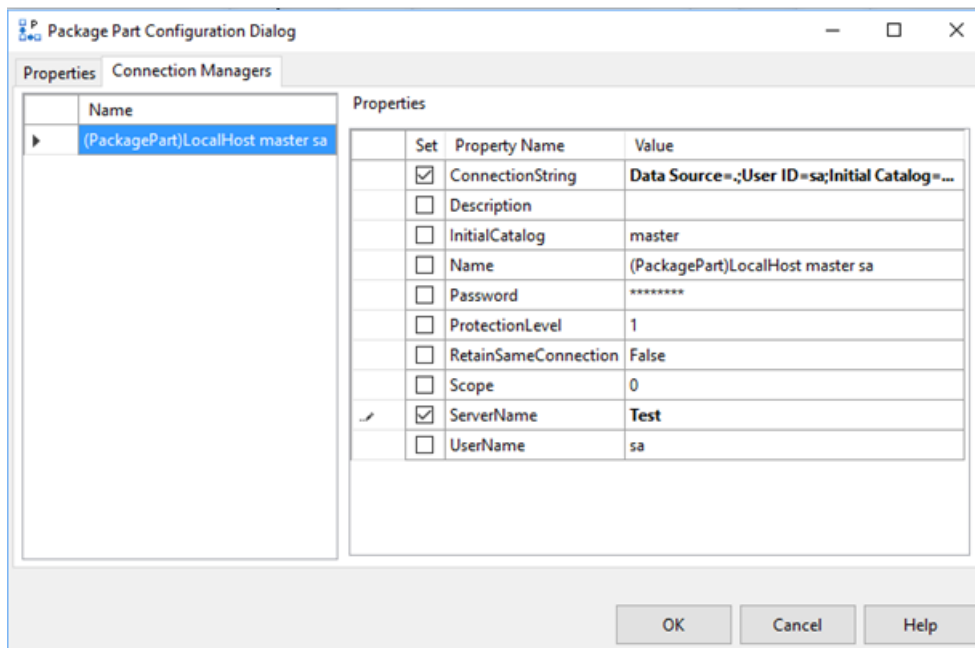
- If cleared, the property is not configured in the part instance. The part instance uses the default value for the property, which is defined in the control flow package part.
- If selected, the value that you enter or select overrides the default value.

The table in the right pane lists the properties to configure.

- **Property Path.** The property path of the property.
- **Property Type.** The data type of the property.
- **Value.** The configured value. This value overrides the default value.

## Connection Managers tab

Use the **Connection Managers** tab of the **Package Part Configuration** dialog box to specify the properties of connection managers for the part instance.



The table in the left pane lists all the connection managers defined in the control flow part. Choose the connection manager that you want to configure.

The list in the right pane lists the properties of the selected connection manager.

- **Set.** Checked if the property is configured for the part instance.
- **Property Name.** The name of the property.
- **Value.** The configured value. This value overrides the default value.

## Delete a control flow part

To delete a part, in Solution Explorer, right-click the part, and then select **Delete**. Select **OK** to confirm the deletion or select **Cancel** to keep the part.

If you delete a part from a project, it is deleted permanently from the file system and it cannot be restored.

### NOTE

If you want to remove a part from an Integration Services project, but continue to use it in other projects, use the **Exclude from Project** option instead of the **Delete** option.

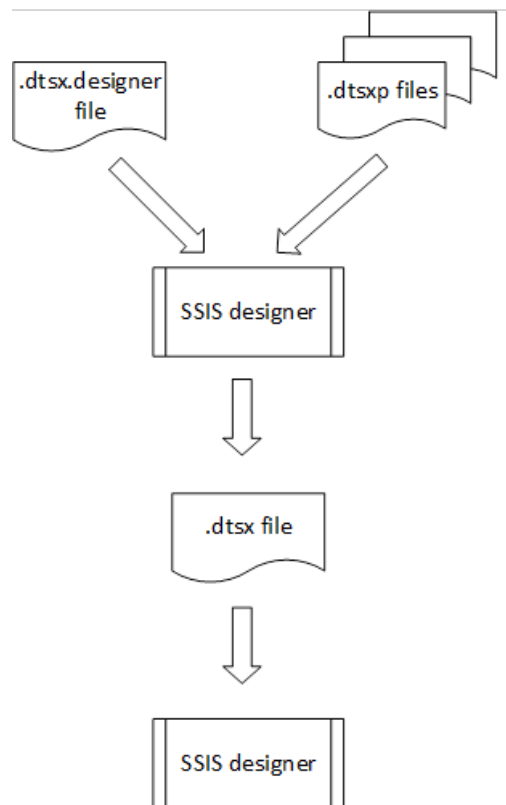
## Package parts are a design-time feature only

Package parts are purely a design-time feature. SSIS designer creates, opens, saves, and updates parts, and adds, configures, or deletes part instances in a package. However, the SSIS runtime is not aware of the parts. Here is how the designer achieves this separation.

- The designer saves package part instances with their configured properties to a ".dtsx.designer" file.
- When the designer saves the ".dtsx.designer" file, it also extracts the content from the parts referenced by this file and replaces the part instances in the package with the content of the parts.

- Finally all the content, which no longer includes part information, is saved back to the “.dtsx” package file. This is the file that the SSIS runtime runs.

The diagram below demonstrates the relationship among parts (“.dtsx” files), SSIS designer, and the SSIS runtime.





# Reuse of Package Objects

3/24/2017 • 1 min to read • [Edit Online](#)

Frequently packages functionality that you want to reuse. For example, if you created a set of tasks, you might want to reuse the items together as a group, or you might want to reuse a single item such as a connection manager that you created in a different Integration Services project.

## Copy and Paste

SQL Server Data Tools (SSDT) and SSIS Designer support copying and pasting package objects, which can include control flow items, data flow items, and connection managers. You can copy and paste between projects and between packages. If the solution contains multiple projects you can copy between projects, and the projects can be of different types.

If a solution contains multiple packages, you can copy and paste between them. The packages can be in the same or different Integration Services projects. However, package objects may have dependencies on other objects, without which they are not valid. For example, an Execute SQL task uses a connection manager, which you must copy as well to make the task work. Also, some package objects require that the package already contain a certain object, and without this object you cannot successfully paste the copied objects into a package. For example, you cannot paste a data flow into a package that does not have at least one Data Flow task.

You may find that you copy the same packages repeatedly. To avoid the copy process, you can designate these packages as templates and use them when you create new packages.

When you copy a package object, Integration Services automatically assigns a new GUID to the **ID** property of the new object and updates the **Name** property.

You cannot copy variables. If an object such as a task uses variables, then you must re-create the variables in the destination package. In contrast, if you copy the entire package, then the variables in the package are also copied.

## Related Tasks

- [Copy Package Objects](#)
- [Copy Project Items](#)
- [Save a Package as a Package Template](#)

# Delete Packages

3/24/2017 • 1 min to read • [Edit Online](#)

In SQL Server Data Tools (SSDT), you can delete packages saved to the file system. If you delete a package, it is deleted permanently and it cannot be restored to an Integration Services project.

## NOTE

If you want to remove packages from an Integration Services project, but use them in other projects, then you should use the **Exclude From Project** option instead of the **Delete** option.

### To delete a package in Business Intelligence

1. In SQL Server Data Tools (SSDT), open the Integration Services project that contains the package you want to delete.
2. In Solution Explorer, right-click the package, and then click **Delete**.
3. Click **OK** to confirm the deletion or click **Cancel** to keep the package.

# dtutil Utility

4/25/2017 • 17 min to read • [Edit Online](#)

The **dtutil** command prompt utility is used to manage SQL Server Integration Services packages. The utility can copy, move, delete, or verify the existence of a package. These actions can be performed on any SSIS package that is stored in one of three locations: a Microsoft SQL Server database, the SSIS Package Store, and the file system. If the utility accesses a package that is stored in **msdb**, the command prompt may require a user name and a password. If the instance of SQL Server uses SQL Server Authentication, the command prompt requires both a user name and a password. If the user name is missing, **dtutil** tries to log on to SQL Server using Windows Authentication. The storage type of the package is identified by the **/SQL**, **/FILE**, and **/DTS** options.

The **dtutil** command prompt utility does not support the use of command files or redirection.

The **dtutil** command prompt utility includes the following features:

- Remarks in the command prompt, which makes the command prompt action self-documenting and easier to understand.
- Overwrite protection, to prompt for a confirmation before overwriting an existing package when you are copying or moving packages.
- Console help, to provide information about the command options for **dtutil**.

## NOTE

Many of the operations that are performed by **dtutil** can also be performed visually in SQL Server Management Studio when you are connected to an instance of Integration Services. For more information, see [Package Management \(SSIS Service\)](#).

The options can be typed in any order. The pipe ("|") character is the **OR** operator and is used to show possible values. You must use one of the options that are delimited by the **OR** pipe.

All options must start with a slash (/) or a minus sign (-). However, do not include a space between the slash or minus sign and the text for the option; otherwise, the command will fail.

Arguments must be strings that are either enclosed in quotation marks or contain no white space.

Double quotation marks within strings that are enclosed in quotation marks represent escaped single quotation marks.

Options and arguments, except for passwords, are not case sensitive.

## Installation Considerations on 64-bit Computers

On a 64-bit computer, Integration Services installs a 64-bit version of the **dtexec** utility (**dtexec.exe**) and the **dtutil** utility (**dtutil.exe**). To install 32-bit versions of these Integration Services tools, you must select either Client Tools or SQL Server Data Tools (SSDT) during setup.

By default, a 64-bit computer that has both the 64-bit and 32-bit versions of an Integration Services command prompt utility installed will run the 32-bit version at the command prompt. The 32-bit version runs because the directory path for the 32-bit version appears in the PATH environment variable before the directory path for the 64-bit version. (Typically, the 32-bit directory path is <drive>:\Program Files(x86)\Microsoft SQL Server\130\DTS\Binn, while the 64-bit directory path is <drive>:\Program Files\Microsoft SQL Server\130\DTS\Binn.)

## NOTE

If you use SQL Server Agent to run the utility, SQL Server Agent automatically uses the 64-bit version of the utility. SQL Server Agent uses the registry, not the PATH environment variable, to locate the correct executable for the utility.

To ensure that you run the 64-bit version of the utility at the command prompt, you can take one of the following actions:

- Open a Command Prompt window, change to the directory that contains the 64-bit version of the utility (<drive>:\Program Files\Microsoft SQL Server\130\DTS\Binn), and then run the utility from that location.
- At the command prompt, run the utility by entering the full path (<drive>:\Program Files\Microsoft SQL Server\130\DTS\Binn) to the 64-bit version of the utility.
- Permanently change the order of the paths in the PATH environment variable by placing the 64-bit path (<drive>:\Program Files\Microsoft SQL Server\130\DTS\Binn) before the 32-bit path (<drive>:\Program Files(x86)\Microsoft SQL Server\130\DTS\Binn) in the variable.

## Syntax

```
dtutil /option [value] [/option [value]]...
```

### Parameters

OPTION	DESCRIPTION
/?	Displays the command prompt options.
/C[opy] <i>location;destinationPathandPackageName</i>	<p>Specifies a copy action on an SSIS package. Use of this parameter requires that you first specify the location of the package using the <b>/FI</b>, <b>/SQ</b>, or <b>/DT</b> option. Next, specify the destination location destination package name. The <i>destinationPathandPackageName</i> argument specifies where the SSIS package is copied to. If the destination <i>location</i> is <b>SQL</b>, the <i>DestUser</i>, <i>DestPassword</i> and <i>DestServer</i> arguments must also be specified in the command.</p> <p>When the <b>Copy</b> action encounters an existing package at the destination, <b>dtutil</b> prompts the user to confirm package deletion. The <b>Y</b> reply overwrites the package and the <b>N</b> reply ends the program. When the command includes the <i>Quiet</i> argument, no prompt appears and any existing package is overwritten.</p>
/Dec[rypt] <i>password</i>	(Optional). Sets the decryption password that is used when you load a package with password encryption.
/Del[ete]	Deletes the package specified by the <b>SQL</b> , <b>DTS</b> or <b>FILE</b> option. If <b>dtutil</b> cannot delete the package, the program ends.

OPTION	DESCRIPTION
<i>/DestP[assword] password</i>	<p>Specifies the password that is used with the SQL option to connect to a destination SQL Server instance using SQL Server Authentication. An error is generated if <i>DESTPASSWORD</i> is specified in a command line that does not include the <i>DTSUSER</i> option.</p> <p>Note: When possible, use Windows Authentication..</p>
<i>/DestS[erver] server_instance</i>	<p>Specifies the server name that is used with any action that causes a destination to be saved to SQL Server. It is used to identify a non-local or non-default server when saving an SSIS package. It is an error to specify <i>DESTSERVER</i> in a command line that does not have an action associated with SQL Server. Actions such as <i>SIGN SQL</i>, <i>COPY SQL</i>, or <i>MOVE SQL</i> options would be appropriate commands to combine with this option.</p> <p>A SQL Server instance name can be specified by adding a backslash and the instance name to the server name.</p>
<i>/DestU[ser] username</i>	<p>Specifies the user name that is used with the <i>SIGN SQL</i>, <i>COPY SQL</i>, and <i>MOVE SQL</i> options to connect to a SQL Server instance that uses SQL Server Authentication. It is an error to specify <i>DESTUSER</i> in a command line that does not include the <i>SIGN SQL</i>, <i>COPY SQL</i>, or <i>MOVE SQL</i> option.</p>
<i>/Dump process ID</i>	<p>(Optional) Causes the specified process, either the <b>dtexec</b> utility or the <b>dtsDebugHost.exe</b> process, to pause and create the debug dump files, .mdmp and .tmp.</p> <p>Note: To use the <b>/Dump</b> option, you must be assigned the Debug Programs user right (SeDebugPrivilege).</p> <p>To find the <i>process ID</i> for the process that you want to pause, use Windows Task Manager.</p> <p>By default, Integration Services stores the debug dump files in the folder, &lt;drive&gt;:\Program Files\Microsoft SQL Server\130\Shared&gt;ErrorDumps.</p> <p>For more information about the <b>dtexec</b> utility and the <b>dtsDebugHost.exe</b> process, see <a href="#">dtexec Utility</a> and <a href="#">Building, Deploying, and Debugging Custom Objects</a>.</p> <p>For more information about debug dump files, see <a href="#">Generating Dump Files for Package Execution</a>.</p> <p>Note: Debug dump files may contain sensitive information. Use an access control list (ACL) to restrict access to the files, or copy the files to a folder with restricted access.</p>

OPTION	DESCRIPTION
/DT[S] <i>filespec</i>	<p>Specifies that the SSIS package to be operated on is located in the SSIS Package Store. The <i>filespec</i> argument must include the folder path, starting with the root of the SSIS Package Store. By default, the names of the root folders in the configuration file are "MSDB" and "File System." Paths that contain a space must be delimited by using double quotation marks.</p> <p>If the DT[S] option is specified on the same command line as any of the following options, a DTEXEC_DTEXECERROR is returned:</p> <p><b>FILE</b></p> <p><b>SQL</b></p> <p><b>SOURCEUSER</b></p> <p><b>SOURCEPASSWORD</b></p> <p><b>SOURCESERVER</b></p>
/En[rypt] {SQL   FILE}; Path;ProtectionLevel[;password]	<p>(Optional). Encrypts the loaded package with the specified protection level and password, and saves it to the location specified in <i>Path</i>. The <i>ProtectionLevel</i> determines whether a password is required.</p> <p><i>SQL</i> - Path is the destination package name.</p> <p><i>FILE</i> - Path is the fully-qualified path and file name for the package.</p> <p><i>DTS</i> - This option is not supported currently.</p> <p><i>ProtectionLevel</i> options:</p> <p>Level 0: Strips sensitive information.</p> <p>Level 1: Sensitive information is encrypted by using local user credentials.</p> <p>Level 2: Sensitive information is encrypted by using the required password.</p> <p>Level 3: Package is encrypted by using the required password.</p> <p>Level 4: Package is encrypted by using local user credentials.</p> <p>Level 5 Package uses SQL Server storage encryption.</p>
/Ex[ists]	<p>(Optional). Used to determine whether a package exists. <b>dtutil</b> tries to locate the package specified by either the <i>SQL</i>, <i>DTS</i> or <i>FILE</i> options. If <b>dtutil</b> cannot locate the package specified, a DTEXEC_DTEXECERROR is returned.</p>
/FC[reate] {SQL   DTS};ParentFolderPath;NewFolderName	<p>(Optional). Create a new folder that has the name that you specified in <i>NewFolderName</i>. The location of the new folder is indicated by the <i>ParentFolderPath</i>.</p>

OPTION	DESCRIPTION
<code>/FDe[lete] {SQL   DTS};ParentFolderPath;FolderName]</code>	(Optional). Deletes from SQL Server or SSIS the folder that was specified by the name in <i>FolderName</i> . The location of the folder to delete is indicated by the <i>ParentFolderPath</i> .
<code>/FDi[rectory] {SQL   DTS};FolderPath[:S]</code>	(Optional). Lists the contents, both folders and packages, in a folder on SSIS or SQL Server. The optional <i>FolderPath</i> parameter specifies the folder that you want to view the content of. The optional <i>S</i> parameter specifies that you want to view a listing of the contents of the subfolders for the folder specified in <i>FolderPath</i> .
<code>/FE[xists ] {SQL   DTS};FolderPath</code>	(Optional). Verifies if the specified folder exists on SSIS or SQL Server. The <i>FolderPath</i> parameter is the path and name of the folder to verify.
<code>/Fi[le] filespec</code>	<p>This option specifies that the SSIS package to be operated on is located in the file system. The <i>filespec</i> value can be provided as either a Universal Naming Convention (UNC) path or local path.</p> <p>If the <i>File</i> option is specified on the same command line as any of the following options, a DTEXEC_DTEXECERROR is returned:</p> <p><b>DTS</b></p> <p><b>SQL</b></p> <p><b>SOURCEUSER</b></p> <p><b>SOURCEPASSWORD</b></p> <p><b>SOURCESERVER</b></p>
<code>/FR[ename] {SQL   DTS} [;ParentFolderPath;OldFolderName;NewFolderName]</code>	(Optional). Renames a folder on the SSIS or SQL Server. The <i>ParentFolderPath</i> is the location of the folder to rename. The <i>OldFolderName</i> is the current name of the folder, and <i>NewFolderName</i> is the new name to give the folder.
<code>/H[elp] option</code>	<p>Displays text extensive help that shows the <b>dtutil</b> options and describes their use. The option argument is optional. If the argument is included, the Help text includes detailed information about the specified option. The following example displays help for all options:</p> <pre>dtutil /H</pre> <p>The following two examples show how to use the <i>/H</i> option to display extended help for a specific option, the <i>/Q [uiet]</i> option, in this example:</p> <pre>dtutil /Help Quiet</pre> <pre>dtutil /H Q</pre>

OPTION	DESCRIPTION
/I[DRegenerate]	<p>Creates a new GUID for the package and updates the package ID property. When a package is copied, the package ID remains the same; therefore, the log files contain the same GUID for both packages. This action creates a new GUID for the newly-copied package to distinguish it from the original.</p>
/M[ove] {SQL   File   DTS}; pathandname	<p>Specifies a move action on an SSIS package. To use this parameter, first specify the location of the package using the <b>/FI</b>, <b>/SQ</b>, or <b>/DT</b> option. Next, specify the <b>Move</b> action. This action requires two arguments, which are separated by a semicolon:</p> <p>The destination argument can specify <i>SQL</i>, <i>FILE</i>, or <i>DTS</i>. A <i>SQL</i> destination can include the <i>DESTUSER</i>, <i>DETPASSWORD</i>, and <i>DESTSERVER</i> options.</p> <p>The <i>pathandname</i> argument specifies the package location: <i>SQL</i> uses the package path and package name, <i>FILE</i> uses a UNC or local path, and <i>DTS</i> uses a location that is relative to the root of the SSIS Package Store. When the destination is <i>FILE</i> or <i>DTS</i>, the path argument does not include the file name. Instead, it uses the package name at the specified location as the file name.</p> <p>When the <b>MOVE</b> action encounters an existing package at the destination, <b>dtutil</b> prompts you to confirm that you want to overwrite the package. The <b>Y</b> reply overwrites the package and the <b>N</b> reply ends the program. When the command includes the <i>QUIET</i> option, no prompt appears and any existing package is overwritten.</p>
/Q[uiet]	<p>Stops the confirmation prompts that can appear when a command including the <b>COPY</b>, <b>MOVE</b>, or <b>SIGN</b> option is executed. These prompts appear if a package with the same name as the specified package already exists at the destination computer or if the specified package is already signed.</p>
/R[emark] text	<p>Adds a comment to the command line. The comment argument is optional. If the comment text includes spaces, the text must be enclosed in quotation marks. You can include multiple <b>REM</b> options in a command line.</p>



OPTION	DESCRIPTION
<p><i>/Si[gn] {SQL   File   DTS}; path; hash</i></p>	<p>Signs an SSIS package. This action uses three required arguments, which are separated by semicolons; destination, path, and hash:</p> <p>The destination argument can specify <i>SQL</i>, <i>FILE</i>, or <i>DTS</i>. A SQL destination can include the <i>DESTUSER</i>, <i>DESTPASSWORD</i> and <i>DESTSERVER</i> options.</p> <p>The path argument specifies the location of the package to take action on.</p> <p>The hash argument specifies a certificate identifier expressed as a hexadecimal string of varying length.</p> <p>For more information, see <a href="#">Identify the Source of Packages with Digital Signatures</a>.</p> <p><b>** Important *</b> When configured to check the signature of the package, Integration Services only checks whether the digital signature is present, is valid, and is from a trusted source. Integration Services does <i>not</i> check whether the package has been changed.</p>
<p><i>/SourceP[assword] password</i></p>	<p>Specifies the password that is used with the <i>SQL</i> and <i>SOURCEUSER</i> options to enable the retrieval of an SSIS package that is stored in a database on a SQL Server instance that uses SQL Server Authentication. It is an error to specify <i>SOURCEPASSWORD</i> in a command line that does not include the <b>SOURCEUSER</b> option.</p> <p>Note: When possible, use Windows Authentication.</p>
<p><i>/SourceS[erver] server_instance</i></p>	<p>Specifies the server name that is used with the <b>SQL</b> option to enable the retrieval of an SSIS package that is stored in SQL Server. It is an error to specify <i>SOURCESERVER</i> in a command line that does not include the <i>SIGN SQL</i>, <i>COPY SQL</i>, or <i>MOVE SQL</i> option.</p> <p>A SQL Server instance name can be specified by adding a backslash and the instance name to the server name.</p>
<p><i>/SourceU[ser] username</i></p>	<p>Specifies the user name that is used with the <i>SOURCESERVER</i> option to enable the retrieval of an SSIS package stored in SQL Server using SQL Server Authentication. It is an error to specify <i>SOURCEUSER</i> in a command line that does not include the <i>SIGN SQL</i>, <i>COPY SQL</i>, or <i>MOVE SQL</i> option.</p> <p>Note: When possible, use Windows Authentication.</p>

OPTION	DESCRIPTION
<i>/SQ[L] package_path</i>	<p>Specifies the location of an SSIS package. This option indicates that the package is stored in the <b>msdb</b> database. The <i>package_path</i> argument specifies the path and name of the SSIS package. Folder names are terminated with back slashes.</p> <p>If the <i>SQL</i> option is specified on the same command line as any of the following options, a DTEXEC_DTEXECERROR is returned:</p> <p><i>DTS</i></p> <p><i>FILE</i></p> <p>The <i>SQL</i> option may be accompanied by zero or one instance of the following options:</p> <p><i>SOURCEUSER</i></p> <p><i>SOURCEPASSWORD</i></p> <p><i>SOURCESERVER</i></p> <p>If <i>SOURCEUSERNAME</i> is not included, Windows Authentication is used to access the package. <i>SOURCEPASSWORD</i> is allowed only if <i>SOURCEUSER</i> is present. If <i>SOURCEPASSWORD</i> is not included, a blank password is used.</p> <p><b>** Important *</b> Do not use a blank password. Use a strong password.</p>

## dtutil Exit Codes

**dtutil** sets an exit code that alerts you when syntax errors are detected, incorrect arguments are used, or invalid combinations of options are specified. Otherwise, the utility reports "The operation completed successfully".The following table lists the values that the **dtutil** utility can set when exiting.

VALUE	DESCRIPTION
0	The utility executed successfully.
1	The utility failed.
4	The utility cannot locate the requested package.
5	The utility cannot load the requested package
6	The utility cannot resolve the command line because it contains either syntactic or semantic errors.

## Remarks

You cannot use command files or redirection with **dtutil**.

The order of the options within the command line is not significant.

## Examples

The following examples detail typical command line usage scenarios.

### Copy Examples

To copy a package that is stored in the **msdb** database on a local instance of SQL Server using Windows Authentication to the SSIS Package Store, use the following syntax:

```
dtutil /SQL srcPackage /COPY DTS;destFolder\destPackage
```

To copy a package from a location on the File system to another location and give the copy a different name, use the following syntax:

```
dtutil /FILE c:\myPackages\mypackage.dtsx /COPY FILE;c:\myTestPackages\mynewpackage.dtsx
```

To copy a package on the local file system to an instance of SQL Server hosted on another computer, use the following syntax:

```
dtutil /FILE c:\sourcepkg.dtsx /DestServer <servername> /COPY SQL;destpkgname
```

Because the */DestU[ser]* and */DestP[assword]* options were not used, Windows Authentication is assumed.

To create a new ID for a package after it is copied, use the following syntax:

```
dtutil /I /FILE copiedpkg.dtsx
```

To create a new ID for all the packages in a specific folder, use the following syntax:

```
for %%f in (C:\test\SSISPackages\*.dtsx) do dtutil.exe /I /FILE %%f
```

Use a single percent sign (%) when typing the command at the command prompt. Use a double percent sign (%%) if the command is used inside a batch file.

### Delete Examples

To delete a package that is stored in the **msdb** database on an instance of SQL Server that uses Windows Authentication, use the following syntax:

```
dtutil /SQL delPackage /DELETE
```

To delete a package that is stored in the **msdb** database on an instance of SQL Server that uses SQL Server Authentication, use the following syntax:

```
dtutil /SQL delPackage /SOURCEUSER srcUserName /SOURCEPASSWORD #8nGs*w7F /DELETE
```

#### NOTE

To delete a package from a named server, include the **SOURCESERVER** option and its argument. You can only specify a server by using the *SQL* option.

To delete a package that is stored in the SSIS Package Store, use the following syntax:

```
dtutil /DTS delPackage.dtsx /DELETE
```

To delete a package that is stored in the file system, use the following syntax:

```
dtutil /FILE c:\delPackage.dtsx /DELETE
```

#### Exists Examples

To determine whether a package exists in the **msdb** database on a local instance of SQL Server that uses Windows Authentication, use the following syntax:

```
dtutil /SQL srcPackage /EXISTS
```

To determine whether a package exists in the **msdb** database on a local instance of SQL Server that uses SQL Server Authentication, use the following syntax:

```
dtutil SQL srcPackage /SOURCEUSER srcUserName /SOURCEPASSWORD *hY$d56b /EXISTS
```

#### NOTE

To determine whether a package exists on a named server, include the **SOURCESERVER** option and its argument. You can only specify a server by using the *SQL* option.

To determine whether a package exists in the local package store, use the following syntax:

```
dtutil /DTS srcPackage.dtsx /EXISTS
```

To determine whether a package exists in the local file system, use the following syntax:

```
dtutil /FILE c:\srcPackage.dtsx /EXISTS
```

#### Move Examples

To move a package that is stored in the SSIS Package Store to the **msdb** database on a local instance of SQL Server that uses Windows Authentication, use the following syntax:

```
dtutil /DTS srcPackage.dtsx /MOVE SQL;destPackage
```

To move a package that is stored in the **msdb** database on a local instance of SQL Server that uses SQL Server Authentication to the **msdb** database on another local instance of SQL Server that uses SQL Server Authentication, use the following syntax:

```
dtutil /SQL srcPackage /SOURCEUSER srcUserName /SOURCEPASSWORD $Hj45jhd@X /MOVE SQL;destPackage /DESTUSER  
destUserName /DETPASSWORD !38dsFH@v
```

#### NOTE

To move a package from one named server to another, include the **SOURCES** and the **DESTS** option and their arguments. You can only specify servers by using the *SQL* option.

To move a package that is stored in the SSIS Package Store, use the following syntax:

```
dtutil /DTS srcPackage.dtsx /MOVE DTS;destPackage.dtsx
```

To move a package that is stored in the file system, use the following syntax:

```
dtutil /FILE c:\srcPackage.dtsx /MOVE FILE;c:\destPackage.dtsx
```

### Sign Examples

To sign a package that is stored in a SQL Server database on a local instance of SQL Server that uses Windows Authentication, use the following syntax:

```
dtutil /FILE srcPackage.dtsx /SIGN FILE;destpkg.dtsx;1767832648918a9d989fdac9819873a91f919
```

To locate information about your certificate, use **CertMgr**. The hash code can be viewed in the **CertMgr** utility by selecting the certificate, and then clicking **View** to view the properties. The **Details** tab provides more information about the certificate. The **Thumbprint** property is used as the hash value, with spaces removed.

#### NOTE

The hash used in this example is not a real hash.

For more information, see the CertMgr section in [Signing and Checking Code with Authenticode](#).

### Encrypt Examples

The following sample encrypts the file-based PackageToEncrypt.dtsx to the file-based EncryptedPackage.dts using full package encryption, with a password. The password that is used for the encryption is *EncPswd*.

```
dtutil /FILE PackageToEncrypt.dtsx /ENCRYPT file;EncryptedPackage.dtsx;3;EncPswd
```

## See Also

[Run Integration Services \(SSIS\) Packages](#)

# SSIS Package Upgrade Wizard F1 Help

3/24/2017 • 8 min to read • [Edit Online](#)

Use the SSIS Package Upgrade Wizard to upgrade packages created by earlier versions of SQL Server to the package format for the current release of SQL Server Integration Services.

## To run the SSIS Package Upgrade Wizard

- [Upgrade Integration Services Packages Using the SSIS Package Upgrade Wizard](#)

## SSIS Upgrade Wizard

### Options

#### Do not show this page again.

Skip the Welcome page the next time that you open the wizard.

## Select Source Location page

Use the **Select Source Location** page to specify the source from which to upgrade packages.

### NOTE

This page is only available when you run the SSIS Package Upgrade Wizard from SQL Server Management Studio or at the command prompt.

### Static Options

#### Package source

Select the storage location that contains the packages to be upgraded. This option has the values listed in the following table.

VALUE	DESCRIPTION
<b>File System</b>	<p>Indicates that the packages to be upgraded are in a folder on the local computer.</p> <p>To have the wizard back up the original packages before upgrading those packages, the original packages must be stored in the file system. For more information, see <a href="#">How To Topic</a>.</p>
<b>SSIS Package Store</b>	<p>Indicates that the packages to be upgraded are in the package store. The package store consists of the set of file system folders that the Integration Services service manages. For more information, see <a href="#">Package Management (SSIS Service)</a>.</p> <p>Selecting this value displays the corresponding <b>Package source</b> dynamic options.</p>

VALUE	DESCRIPTION
<b>Microsoft SQL Server</b>	<p>Indicates the packages to be upgraded are from an existing instance of SQL Server.</p> <p>Selecting this value displays the corresponding <b>Package source</b> dynamic options.</p>

### Folder

Type the name of a folder that contains the packages you want to upgrade or click **Browse** and locate the folder.

### Browse

Browse to locate the folder that contains the packages you want to upgrade.

### Package Source Dynamic Options

**Package source** = SSIS Package Store

### Server

Type the name of the server that has the packages to be upgraded, or select this server in the list.

**Package source** = Microsoft SQL Server

### Server

Type the name of the server that has the packages to be upgraded, or select this server from the list.

### Use Windows authentication

Select to use Windows Authentication to connect to the server.

### Use SQL Server authentication

Select to use SQL Server Authentication to connect to the server. If you use SQL Server Authentication, you must provide a user name and password.

### User name

Type the user name that SQL Server Authentication will use to connect to the server.

### Password

Type the password that SQL Server Authentication will use to connect to the server.

## Select Destination Location page

Use the **Select Destination Location** page to specify the destination to which to save the upgraded packages.

### NOTE

This page is only available when you run the SSIS Package Upgrade Wizard from SQL Server Management Studio or at the command prompt.

### Static Options

#### Save to source location

Save the upgraded packages to the same location as specified on the **Select Source Location** page of the wizard.

If the original packages are stored in the file system and you want the wizard to back up those packages, select the **Save to source location** option. For more information, see [Upgrade Integration Services Packages Using the SSIS Package Upgrade Wizard](#).

#### Select new destination location

Save the upgraded packages to the destination location that is specified on this page.

### Package source

Specify where the upgrade packages are to be stored. This option has the values listed in the following table.

VALUE	DESCRIPTION
<b>File System</b>	Indicates that the upgraded packages are to be saved to a folder on the local computer.
<b>SSIS Package Store</b>	<p>Indicates that the upgraded packages are to be saved to the Integration Services package store. The package store consists of the set of file system folders that the Integration Services service manages. For more information, see <a href="#">Package Management (SSIS Service)</a>.</p> <p>Selecting this value displays the corresponding <b>Package source</b> dynamics options.</p>
<b>Microsoft SQL Server</b>	<p>Indicates that the upgraded packages are to be saved to an existing instance of SQL Server.</p> <p>Selecting this value displays the corresponding dynamic <b>Package source</b> dynamic options.</p>

### Folder

Type the name of a folder to which the upgraded packages will be saved, or click **Browse** and locate the folder.

### Browse

Browse to locate the folder to which the upgraded packages will be saved.

### Package Source Dynamic Options

**Package source = SSIS Package Store**

#### Server

Type the name of the server to which the upgrade packages will be saved, or select a server in the list.

**Package source = Microsoft SQL Server**

#### Server

Type the name of the server to which the upgrade packages will be saved, or select this server in the list.

### Use Windows authentication

Select to use Windows Authentication to connect to the server.

### Use SQL Server authentication

Select to use SQL Server Authentication to connect to the server. If you use SQL Server Authentication, you must provide a user name and password.

#### User name

Type the user name to be used when using SQL Server Authentication to connect to the server.

#### Password

Type the password to be used when using SQL Server Authentication to connect to the server.

## Select Package Management Options page

Use the **Select Package Management Options** page to specify options for upgrading packages.

### To run the SSIS Package Upgrade Wizard

- [Upgrade Integration Services Packages Using the SSIS Package Upgrade Wizard](#)

### Options



## Update connection strings to use new provider names

Update the connection strings to use the names for the following providers for the current release of Integration Services:

- OLE DB Provider for Analysis Services
- SQL Server Native Client

The SSIS Package Upgrade Wizard updates only connection strings that are stored in connection managers. The wizard does not update connection strings that are constructed dynamically by using the Integration Services expression language, or by using code in a Script task.

## Validate upgrade packages

Validate the upgrade packages and save only those upgrade packages that pass validation.

If you do not select this option, the wizard will not validate upgrade packages. Therefore, the wizard will save all upgrade packages, regardless of whether the packages are valid or not. The wizard saves upgrade packages to the destination that is specified on the **SelectDestination Location** page of the wizard.

Validation adds time to the upgrade process. We recommend that you do not select this option for large packages that are likely to be upgraded successfully.

## Create new package IDs

Create new package IDs for the upgrade packages.

## Continue upgrade process when a package upgrade fails

Specify that when a package cannot be upgraded, the SSIS Package Upgrade Wizard continues to upgrade the remaining packages.

## Package name conflicts

Specify how the wizard should handle packages that have the same name. This option has the values listed in the following table.

## Overwrite existing package files

Replaces the existing package with the upgrade package of the same name.

## Add numeric suffixes to upgrade package names

Adds a numeric suffix to the name of the upgrade package.

## Do not upgrade packages

Stops the packages from being upgraded and displays an error when you complete the wizard.

These options are not available when you select the **Save to source location** option on the **Select Destination Location** page of the wizard.

## Ignore Configurations

Does not load package configurations during the package upgrade. Selecting this option reduces the time required to upgrade the package.

## Backup original packages

Have the wizard back up the original packages to an **SSISBackupFolder** folder. The wizard creates the **SSISBackupFolder** folder as a subfolder to the folder that contains the original packages and the upgraded packages.

### NOTE

This option is available only when you specify that the original packages and the upgraded packages are stored in the file system and in the same folder.

## Select Packages page

Use the **Select Packages** page to select the packages to upgrade. This page lists the packages that are stored in the location that was specified on the **Select Source Location** page of the wizard.

### Options

#### Existing package name

Select one or more packages to upgrade.

#### Upgrade package name

Provide the destination package name, or use the default name that the wizard provides.

#### NOTE

You can also change the destination package name after upgrading the package. In SQL Server Data Tools (SSDT) or SQL Server Management Studio, open the upgraded package and change the package name.

### Password

Specify the password that is used to decrypt the selected upgrade packages.

### Apply to selection

Apply the specified password to decrypt the selected upgrade packages.

## Complete the Wizard page

Use the **Complete the Wizard** page to review and confirm the package upgrade options that you have selected. This is the last wizard page from which you can go back and change options for this session of the wizard.

### Options

#### Summary of options

Review the upgrade options that you have selected in the wizard. To change any options, click **Back** to return to previous wizard pages.

## Upgrading the Packages page

Use the **Upgrading the Packages** page to view the progress of package upgrade and to interrupt the upgrade process. The SSIS Package Upgrade Wizard upgrades the selected packages one by one.

### Options

#### Message pane

Displays progress messages and summary information during the upgrade process.

#### Action

View the actions in the upgrade.

#### Status

View the result of each action.

#### Message

View the error messages that each action generates.

#### Stop

Stop the package upgrade.

#### Report

Select what you want to do with the report that contains the results of the package upgrade:

- View the report online.
- Save the report to a file.
- Copy the report to the Clipboard
- Send the report as an e-mail message.

## View upgraded packages

### **View upgraded packages that were saved to a SQL Server database or to the package store**

In Management Studio, in Object Explorer, connect to the local instance of Integration Services, and then expand the **Stored Packages** node to see the packages that were upgraded.

### **View upgraded packages that were upgraded from SQL Server Data Tools**

In SQL Server Data Tools (SSDT), in Solution Explorer, open the Integration Services project, and then expand the **SSIS Packages** node to see the upgraded packages.

## See Also

[Upgrade Integration Services Packages](#)

# Integration Services (SSIS) Package and Project Parameters

4/19/2017 • 11 min to read • [Edit Online](#)

Integration Services (SSIS) parameters allow you to assign values to properties within packages at the time of package execution. You can create *project parameters* at the project level and *package parameters* at the package level. Project parameters are used to supply any external input the project receives to one or more packages in the project. Package parameters allow you to modify package execution without having to edit and redeploy the package.

In SQL Server Data Tools you create, modify, or delete project parameters by using the **Project.params** window. You create, modify, and delete package parameters by using the **Parameters** tab in the SSIS Designer. You associate a new or an existing parameter with a task property by using the **Parameterize** dialog box. For more about using the **Project.params** window and the **Parameters** tab, see [Create Parameters](#). For more information about the **Parameterize** dialog box, see [Parameterize Dialog Box](#).

## Parameters and Package Deployment Model

In general, if you are deploying a package using the package deployment model, you should use configurations instead of parameters.

When you deploy a package that contains parameters using the package deployment model and then execute the package, the parameters are not called during execution. If the package contains package parameters and expressions within the package use the parameters, the resulting values are applied at runtime. If the package contains project parameters, the package execution may fail.

## Parameters and Project Deployment Model

When you deploy a project to the Integration Services (SSIS) server, you use views, stored procedures, and the SQL Server Management Studio UI to manage project and package parameters. For more information, see the following topics.

- [Views \(Integration Services Catalog\)](#)
- [Stored Procedures \(Integration Services Catalog\)](#)
- [Configure Dialog Box](#)
- [Execute Package Dialog Box](#)

### Parameter Values

You can assign up to three different types of values to a parameter. When a package execution is started, a single value is used for the parameter, and the parameter is resolved to its final literal value.

The following table lists the types of values.

VALUE NAME	DESCRIPTION	TYPE OF VALUE
------------	-------------	---------------

VALUE NAME	DESCRIPTION	TYPE OF VALUE
Execution Value	The value that is assigned to a specific instance of package execution. This assignment overrides all other values, but applies to only a single instance of package execution.	Literal
Server Value	The value assigned to the parameter within the scope of the project, after the project is deployed to the Integration Services server. This value overrides the design default.	Literal or Environment Variable Reference
Design Value	The value assigned to the parameter when the project is created or edited in SQL Server Data Tools. This value persists with the project.	Literal

You can use a single parameter to assign a value to multiple package properties. A single package property can be assigned a value only from a single parameter.

### Executions and Parameter Values

The *execution* is an object that represents a single instance of package execution. When you create an execution, you specify all of the details necessary to run a package such as execution parameter values. You can also modify the parameters values for existing executions.

When you explicitly set an execution parameter value, the value is applicable only to that particular instance of execution. The execution value is used instead of a server value or a design value. If you do not explicitly set an execution value, and a server value has been specified, the server value is used.

When a parameter is marked as required, a server value or execution value must be specified for that parameter. Otherwise, the corresponding package does not execute. Although the parameter has a default value at design time, it will never be used once the project is deployed.

### Environment Variables

If a parameter references an environment variable, the literal value from that variable is resolved through the specified environment reference and applied to the parameter. The final literal parameter value that is used for package execution is referred to as the execution parameter value. You specify the environment reference for an execution by using the **Execute** dialog box

If a project parameter references an environment variable and the literal value from the variable cannot be resolved at execution, the design value is used. The server value is not used.

To view the environment variables that are assigned to parameter values, query the `catalog.object_parameters` view. For more information, see [catalog.object\\_parameters \(SSISDB Database\)](#).

### Determining Execution Parameter Values

The following Transact-SQL views and stored procedure can be used to display and set parameter values.

[catalog.execution\\_parameter\\_values \(SSISDB Database\)](#)(view)

Shows the actual parameter values that will be used by a specific execution

[catalog.get\\_parameter\\_values \(SSISDB Database\)](#) (stored procedure)

Resolves and shows the actual values for the specified package and environment reference

[catalog.object\\_parameters \(SSISDB Database\)](#) (view)

Displays the parameters and properties for all packages and projects in the Integration Services catalog, including

the design default and server default values.

#### [catalog.set\\_execution\\_parameter\\_value \(SSISDB Database\)](#)

Sets the value of a parameter for an instance of execution in the Integration Services catalog.

You can also use the **Execute Package** dialog box in SQL Server Data Tools (SSDT) to modify the parameter value. For more information, see [Execute Package Dialog Box](#).

You can also use the dtexec **/Parameter** option to modify a parameter value. For more information, see [dtexec Utility](#).

### Parameter Validation

If parameter values cannot be resolved, the corresponding package execution will fail. To help avoid failures, you can validate projects and packages by using the **Validate** dialog box in SQL Server Data Tools (SSDT). Validation allows you to confirm that all parameters have the necessary values or can resolve the necessary values with specific environment references. Validation also checks for other common package issues.

For more information, see [Validate Dialog Box](#).

### Parameter Example

This example describes a parameter named **pkgOptions** that is used to specify options for the package in which it resides.

During design time, when the parameter was created in SQL Server Data Tools, a default value of 1 was assigned to the parameter. This default value is referred to as the design default. If the project was deployed to the SSISDB catalog and no other values were assigned to this parameter, the package property corresponding to the **pkgOptions** parameter would be assigned the value of 1 during package execution. The design default persists with the project throughout its life cycle.

While preparing a specific instance of package execution, a value of 5 is assigned to the **pkgOptions** parameter. This value is referred to as the execution value because it applies to the parameter only for that particular instance of execution. When execution starts, the package property corresponding to the **pkgOptions** parameter is assigned the value of 5.

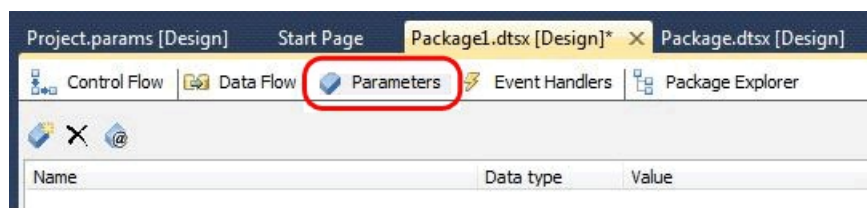
## Create parameters

You use SQL Server Data Tools (SSDT) to create project parameters and package parameters. The following procedures provide step-by-step instructions for creating package/project parameters.

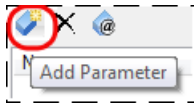
**NOTE:** If you are converting a project that you created using an earlier version of Integration Services to the project deployment model, you can use the **Integration Services Project Conversion Wizard** to create parameters based on configurations. For more information, see [Deploy Integration Services \(SSIS\) Projects and Packages](#).

### Create package parameters

1. Open the package in SQL Server Data Tools, and then click the **Parameters** tab in the SSIS Designer.



2. Click the **Add Parameter** button on the toolbar.



- Enter values for the **Name**, **Data Type**, **Value**, **Sensitive**, and **Required** properties in the list itself or in the **Properties** window. The following table describes these properties.

PROPERTY	DESCRIPTION
Name	The name of the parameter.
Data type	The data type of the parameter.
Default value	The default value for the parameter assigned at design time. This is also known as the design default.
Sensitive	Sensitive parameter values are encrypted in the catalog and appear as a NULL value when viewed with Transact-SQL or SQL Server Management Studio.
Required	Requires that a value, other than the design default, is specified before the package can execute.
Description	For maintainability, the description of the parameter. In SQL Server Data Tools (SSDT), set the parameter description in the Visual Studio Properties window when the parameter is selected in the applicable parameters window.

**NOTE:** When you deploy a project to the catalog, several more properties become associated with the project. To see all properties for all parameters in the catalog, use the [catalog.object\\_parameters \(SSISDB Database\)](#) view.

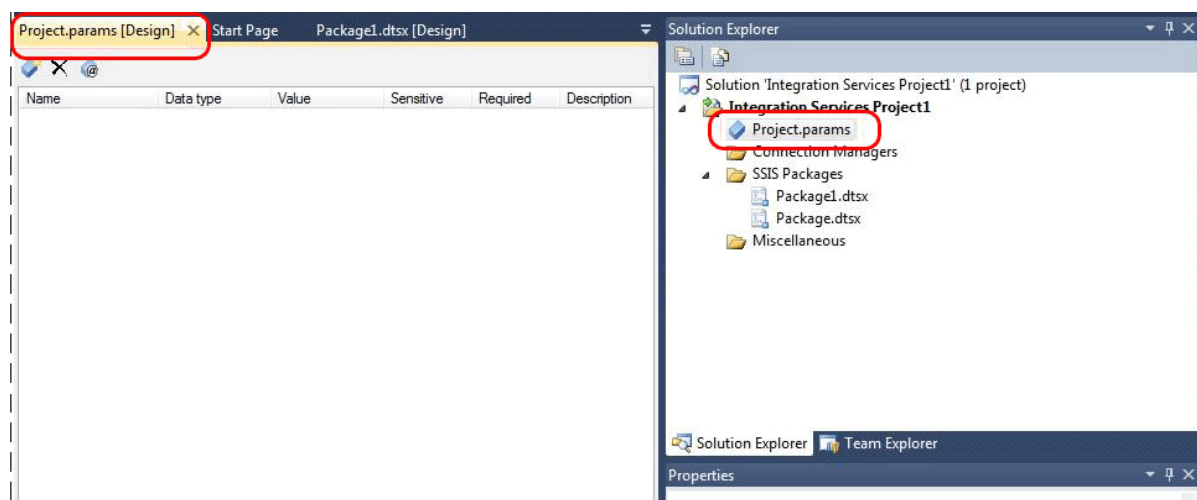
- Save the project to save changes to parameters. Parameter values are stored in the project file.

**WARNING!!** You can in-place edit in the list or use the **Properties** window to modify the values of parameter properties. You can delete a parameter by using the **Delete (X)** toolbar button. Using the last toolbar button, you can specify a value for a parameter that is used only when you execute the package in SQL Server Data Tools.

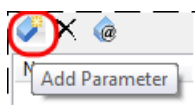
**NOTE:** If you re-open the package file without opening the project in SQL Server Data Tools, the **Parameters** tab will be empty and disabled.

### Create project parameters

- Open the project in SQL Server Data Tools.
- Right-click **Project.params** in Solution Explorer, and then click **Open** (OR) double-click **Project.params** to open it.



- Click the **Add Parameter** button on the toolbar.



- Enter values for the **Name**, **Data Type**, **Value**, **Sensitive**, and **Required** properties.

PROPERTY	DESCRIPTION
Name	The name of the parameter.
Data type	The data type of the parameter.
Default value	The default value for the parameter assigned at design time. This is also known as the design default.
Sensitive	Sensitive parameter values are encrypted in the catalog and appear as a NULL value when viewed with Transact-SQL or SQL Server Management Studio.
Required	Requires that a value, other than the design default, is specified before the package can execute.
Description	For maintainability, the description of the parameter. In SQL Server Data Tools, set the parameter description in the Visual Studio Properties window when the parameter is selected in the applicable parameters window.

- Save the project to save changes to parameters. Parameter values are stored in configurations in the project file. Save the project file to commit to disk any changes in the parameter values.

**WARNING!!!** You can in-place edit in the list or use the **Properties** window to modify the values of parameter properties. You can delete a parameter by using the **Delete (X)** toolbar button. Using the last toolbar button to open the **Manage Parameter Values** dialog box, you can specify a value for a parameter that is used only when you execute the package in SQL Server Data Tools.

## Parameterize Dialog Box

The **Parameterize** dialog box lets you associate a new or an existing parameter with a property of a task. You open the dialog box by right-clicking a task or the Control Flow tab in SSIS Designer and then by clicking **Parameterize**. The following list describes UI elements in the dialog box. For more information about parameters, see [Integration](#)



## Options

### Property

Select the property of the task that you want to associate with a parameter. This list is populated with all the properties that can be parameterized.

### Use existing parameter

Select this option to associate the property of task with an existing parameter and then select the parameter from drop-down list.

### Do not use parameter

Select this option to remove a reference to a parameter. The parameter is not deleted.

### Create new parameter

Select this option to create a new parameter that you want to associate with the property of the task.

### Name

Specify the name of the parameter you want to create.

### Description

Specify the description for parameter.

### Value

Specify the default value for the parameter. This is also known as the design default, which can be overridden later at the deployment time.

### Scope

Specify the scope of the parameter by selecting either **Project** or **Package** option. Project parameters are used to supply any external input the project receives to one or more packages in the project. Package parameters allow you to modify package execution without having to edit and redeploy the package.

### Sensitive

Specify whether the parameter is a sensitive by checking or clearing the check box. Sensitive parameter values are encrypted in the catalog and appear as a NULL value when viewed with Transact-SQL or SQL Server Management Studio.

### Required

Specify whether the parameter requires that a value, other than the design default, is specified before the package can execute.

## Set parameter values after the project is deployed

The Deployment Wizard allows you to set server default parameter values when you deploy your project to the catalog. After your project is in the catalog, you can use SQL Server Management Studio (SSMS) Object Explorer or Transact-SQL to set server default values.

### Set server defaults with SSMS Object Explorer

1. Select and right-click the project under the **Integration Services** node.
2. Click **Properties** to open the **Project Properties** dialog window.
3. Open the parameters page by clicking **Parameters** under **Select a page**.
4. Select the desired parameter in the **Parameters** list. Note: The **Container** column helps distinguish project parameters from package parameters.
5. In the **Value** column, specify the desired server default parameter value.

## Set server defaults with Transact-SQL

To set server defaults with Transact-SQL, use the [catalog.set\\_object\\_parameter\\_value \(SSISDB Database\)](#) stored procedure. To view current server defaults, query the [catalog.object\\_parameters \(SSISDB Database\)](#) view. To clear a server default value, use the [catalog.clear\\_object\\_parameter\\_value \(SSISDB Database\)](#) stored procedure.

## Related Content

Blog entry, [SSIS Quick Tip: Required Parameters](#), on [mattmasson.com](#).

# Integration Services (SSIS) Connections

4/14/2017 • 16 min to read • [Edit Online](#)

Microsoft SQL Server Integration Services packages use connections to perform different tasks and to implement Integration Services features:

- Connecting to source and destination data stores such as text, XML, Excel workbooks, and relational databases to extract and load data.
- Connecting to relational databases that contain reference data to perform exact or fuzzy lookups.
- Connecting to relational databases to run SQL statements such as SELECT, DELETE, and INSERT commands and also stored procedures.
- Connecting to SQL Server to perform maintenance and transfer tasks such as backing up databases and transferring logins.
- Writing log entries in text and XML files and SQL Server tables and package configurations to SQL Server tables.
- Connecting to SQL Server to create temporary work tables that some transformations require to do their work.
- Connecting to Analysis Services projects and databases to access data mining models, process cubes and dimensions, and run DDL code.
- Specifying existing or creating new files and folders to use with Foreach Loop enumerators and tasks.
- Connecting to message queues and to Windows Management Instrumentation (WMI), SQL Server Management Objects (SMO), Web, and mail servers.

To make these connections, Integration Services uses connection managers, as described in the next section.

## Connection Managers

Integration Services uses the connection manager as a logical representation of a connection. At design time, you set the properties of a connection manager to describe the physical connection that Integration Services creates when the package runs. For example, a connection manager includes the **ConnectionString** property that you set at design time; at run time, a physical connection is created using the value in the connection string property.

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.

SQL Server Integration Services provides different types of connection managers that enable packages to connect to a variety of data sources and servers:

- There are built-in connection managers that Setup installs when you install Integration Services.
- There are connection managers that are available for download from the Microsoft website.
- You can create your own custom connection manager if the existing connection managers do not meet your needs.

### Package level and project level connection managers

A connection manager can be created at the package level or at the project level. The connection manager created at the project level is available all the packages in the project. Whereas, connection manager created at the package level is available to that specific package.

You use connection managers that are created at the project level in place of data sources, to share connections to sources. To add a connection manager at the project level, the Integration Services project must use the project deployment model. When a project is configured to use this model, the **Connection Managers** folder appears in **Solution Explorer**, and the **Data Sources** folder is removed from **Solution Explorer**.

#### NOTE

If you want to use data sources in your package, you need to convert the project to the package deployment model.

For more information about the two models, and about converting a project to the project deployment model, see [Deploy Integration Services \(SSIS\) Projects and Packages](#).

### Built-in Connection Managers

The following table lists the connection manager types that SQL Server Integration Services provides.

TYPE	DESCRIPTION	TOPIC
ADO	Connects to ActiveX Data Objects (ADO) objects.	<a href="#">ADO Connection Manager</a>
ADO.NET	Connects to a data source by using a .NET provider.	<a href="#">ADO.NET Connection Manager</a>
CACHE	Reads data from the data flow or from a cache file (.caw), and can save data to the cache file.	<a href="#">Cache Connection Manager</a>
DQS	Connects to a Data Quality Services server and a Data Quality Services database on the server.	<a href="#">DQS Cleansing Connection Manager</a>
EXCEL	Connects to an Excel workbook file.	<a href="#">Excel Connection Manager</a>
FILE	Connects to a file or a folder.	<a href="#">File Connection Manager</a>
FLATFILE	Connect to data in a single flat file.	<a href="#">Flat File Connection Manager</a>
FTP	Connect to an FTP server.	<a href="#">FTP Connection Manager</a>
HTTP	Connects to a webserver.	<a href="#">HTTP Connection Manager</a>
MSMQ	Connects to a message queue.	<a href="#">MSMQ Connection Manager</a>
MSOLAP100	Connects to an instance of SQL Server Analysis Services or an Analysis Services project.	<a href="#">Analysis Services Connection Manager</a>
MULTIFILE	Connects to multiple files and folders.	<a href="#">Multiple Files Connection Manager</a>
MULTIFLATFILE	Connects to multiple data files and folders.	<a href="#">Multiple Flat Files Connection Manager</a>

TYPE	DESCRIPTION	TOPIC
OLEDB	Connects to a data source by using an OLE DB provider.	<a href="#">OLE DB Connection Manager</a>
ODBC	Connects to a data source by using ODBC.	<a href="#">ODBC Connection Manager</a>
SMOServer	Connects to a SQL Server Management Objects (SMO) server.	<a href="#">SMO Connection Manager</a>
SMTP	Connects to an SMTP mail server.	<a href="#">SMTP Connection Manager</a>
SQLMOBILE	Connects to a SQL Server Compact database.	<a href="#">SQL Server Compact Edition Connection Manager</a>
WMI	Connects to a server and specifies the scope of Windows Management Instrumentation (WMI) management on the server.	<a href="#">WMI Connection Manager</a>

### Connection Managers available for download

The following table lists additional types of connection manager that you can download from the Microsoft website.

#### IMPORTANT

The connection managers listed in the following table work only with Microsoft SQL Server 2012 Enterprise and Microsoft SQL Server 2012 Developer.

TYPE	DESCRIPTION	TOPIC
ORACLE	Connects to an Oracle <version info> server.	The Oracle connection manager is the connection manager component of the Microsoft Connector for Oracle by Attunity. The Microsoft Connector for Oracle by Attunity also includes a source and a destination. For more information, see the download page, <a href="#">Microsoft Connectors for Oracle and Teradata by Attunity</a> .
SAPBI	Connects to an SAP NetWeaver BI version 7 system.	The SAP BI connection manager is the connection manager component of the Microsoft Connector for SAP BI. The Microsoft Connector for SAP BI also includes a source and a destination. For more information, see the download page, <a href="#">Microsoft SQL Server 2008 Feature Pack</a> .

TYPE	DESCRIPTION	TOPIC
TERADATA	Connects to a Teradata <version info> server.	The Teradata connection manager is the connection manager component of the Microsoft Connector for Teradata by Attunity. The Microsoft Connector for Teradata by Attunity also includes a source and a destination. For more information, see the download page, <a href="#">Microsoft Connectors for Oracle and Teradata by Attunity</a> .

### Custom Connection Managers

You can also write custom connection managers. For more information, see [Developing a Custom Connection Manager](#).

## Create connection managers

Integration Services includes a variety of connection managers to suit the needs of tasks that connect to different types of servers and data sources. Connection managers are used by the data flow components that extract and load data in different types of data stores, and by the log providers that write logs to a server, SQL Server table, or file. For example, a package with a Send Mail task uses an SMTP connection manager type to connect to a Simple Mail Transfer Protocol (SMTP) server. A package with an Execute SQL task can use an OLE DB connection manager to connect to a SQL Server database. For more information, see [Integration Services \(SSIS\) Connections](#).

To automatically create and configure connection managers when you create a new package, you can use the SQL Server Import and Export Wizard. The wizard also helps you create and configure the sources and destinations that use the connection managers. For more information, see [Create Packages in SQL Server Data Tools](#).

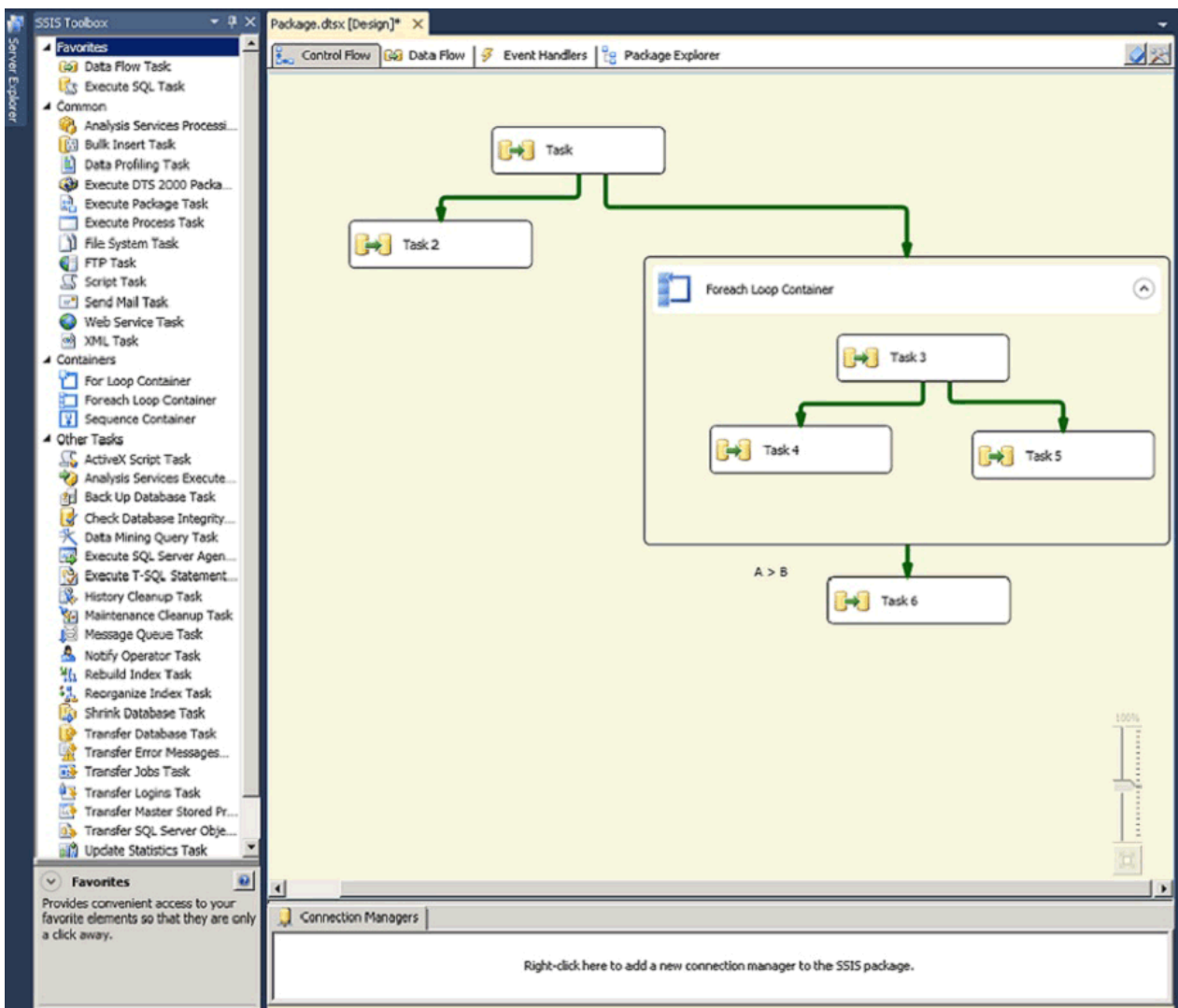
To manually create a new connection manager and add it to an existing package, you use the **Connection Managers** area that appears on the **Control Flow**, **Data Flow**, and **Event Handlers** tabs of SSIS Designer. From the **Connection Manager** area, you choose the type of connection manager to create, and then set the properties of the connection manager by using a dialog box that SSIS Designer provides. For more information, see the section, "Using the Connection Managers Area," later in this topic.

After the connection manager is added to a package, you can use it in tasks, Foreach Loop containers, sources, transformations, and destinations. For more information, see [Integration Services Tasks](#), [Foreach Loop Container](#), and [Data Flow](#).

### Using the Connection Managers Area

You can create connection managers while the **Control Flow**, **Data Flow**, or **Event Handlers** tab of SSIS Designer is active.

The following diagram shows the **Connection Managers** area on the **Control Flow** tab of SSIS Designer.



## 32-Bit and 64-Bit Providers for Connection Managers

Many of the providers that connection managers use are available in 32-bit and 64-bit versions. The Integration Services design environment is a 32-bit environment and you see only 32-bit providers while you are designing a package. Therefore, you can only configure a connection manager to use a specific 64-bit provider if the 32-bit version of the same provider is also installed.

At run time, the correct version is used, and it does not matter that you specified the 32-bit version of the provider at design time. The 64-bit version of the provider can be run even if the package is run in SQL Server Data Tools (SSDT).

Both versions of the provider have the same ID. To specify whether the Integration Services runtime uses an available 64-bit version of the provider, you set the `Run64BitRuntime` property of the Integration Services project. If the `Run64BitRuntime` property is set to **true**, the runtime finds and uses the 64-bit provider; if `Run64BitRuntime` is **false**, the runtime finds and uses the 32-bit provider. For more information about properties you can set on Integration Services projects, see [Integration Services &\(SSIS\) and Studio Environments](#).

## Add a connection manager

### Add a connection manager when you create a package

- Use the SQL Server Import and Export Wizard

In addition to creating and configuring a connection manager, the wizard also helps you create and configure the sources and destinations that use the connection manager. For more information, see [Create Packages in SQL Server Data Tools](#).

### Add a connection manager to an existing package

1. In SQL Server Data Tools (SSDT), open the Integration Services project that contains the package you want.
2. In Solution Explorer, double-click the package to open it
3. In SSIS Designer, click the **Control Flow** tab, the **Data Flow** tab, or the **Event Handler** tab to make the **Connection Managers** area available.
4. Right-click anywhere in the **Connection Managers** area, and then do one of the following:
  - Click the connection manager type to add to the package.

—or—

- If the type that you want to add is not listed, click **New Connection** to open the **Add SSIS Connection Manager** dialog box, select a connection manager type, and then click **OK**.

The custom dialog box for the selected connection manager type opens. For more information about connection manager types and the options that are available, see the following options table.

CONNECTION MANAGER	OPTIONS
<a href="#">ADO Connection Manager</a>	<a href="#">Configure OLE DB Connection Manager</a>
<a href="#">ADO.NET Connection Manager</a>	<a href="#">Configure ADO.NET Connection Manager</a>
<a href="#">Analysis Services Connection Manager</a>	<a href="#">Add Analysis Services Connection Manager Dialog Box UI Reference</a>
<a href="#">Excel Connection Manager</a>	<a href="#">Excel Connection Manager Editor</a>
<a href="#">File Connection Manager</a>	<a href="#">File Connection Manager Editor</a>
<a href="#">Multiple Files Connection Manager</a>	<a href="#">Add File Connection Manager Dialog Box UI Reference</a>
<a href="#">Flat File Connection Manager</a>	<a href="#">Flat File Connection Manager Editor (General Page)</a> <a href="#">Flat File Connection Manager Editor (Columns Page)</a> <a href="#">Flat File Connection Manager Editor (Advanced Page)</a> <a href="#">Flat File Connection Manager Editor (Preview Page)</a>
<a href="#">Multiple Flat Files Connection Manager</a>	<a href="#">Multiple Flat Files Connection Manager Editor (General Page)</a> <a href="#">Multiple Flat Files Connection Manager Editor (Columns Page)</a> <a href="#">Multiple Flat Files Connection Manager Editor (Advanced Page)</a> <a href="#">Multiple Flat Files Connection Manager Editor (Preview Page)</a>
<a href="#">FTP Connection Manager</a>	<a href="#">FTP Connection Manager Editor</a>
<a href="#">HTTP Connection Manager</a>	<a href="#">HTTP Connection Manager Editor (Server Page)</a> <a href="#">HTTP Connection Manager Editor (Proxy Page)</a>



CONNECTION MANAGER	OPTIONS
<a href="#">MSMQ Connection Manager</a>	<a href="#">MSMQ Connection Manager Editor</a>
<a href="#">ODBC Connection Manager</a>	<a href="#">ODBC Connection Manager UI Reference</a>
<a href="#">OLE DB Connection Manager</a>	<a href="#">Configure OLE DB Connection Manager</a>
<a href="#">SMO Connection Manager</a>	<a href="#">SMO Connection Manager Editor</a>
<a href="#">SMTP Connection Manager</a>	<a href="#">SMTP Connection Manager Editor</a>
<a href="#">SQL Server Compact Edition Connection Manager</a>	<a href="#">SQL Server Compact Edition Connection Manager Editor (Connection Page)</a>  <a href="#">SQL Server Compact Edition Connection Manager Editor (All Page)</a>
<a href="#">WMI Connection Manager</a>	<a href="#">WMI Connection Manager Editor</a>

The **Connection Managers** area lists the added connection manager.

- Optionally, right-click the connection manager, click **Rename**, and then modify the default name of the connection manager.
- To save the updated package, click **Save Selected Item** on the **File** menu.

### Add a connection manager at the project level

- In SQL Server Data Tools (SSDT), open the Integration Services project.
- In **Solution Explorer**, right-click **Connection Managers**, and click **New Connection Manager**.
- In the **Add SSIS Connection Manager** dialog box, select the type of connection manager, and then click **Add**.

The custom dialog box for the selected connection manager type opens. For more information about connection manager types and the options that are available, see the following options table.

CONNECTION MANAGER	OPTIONS
<a href="#">ADO Connection Manager</a>	<a href="#">Configure OLE DB Connection Manager</a>
<a href="#">ADO.NET Connection Manager</a>	<a href="#">Configure ADO.NET Connection Manager</a>
<a href="#">Analysis Services Connection Manager</a>	<a href="#">Add Analysis Services Connection Manager Dialog Box UI Reference</a>
<a href="#">Excel Connection Manager</a>	<a href="#">Excel Connection Manager Editor</a>
<a href="#">File Connection Manager</a>	<a href="#">File Connection Manager Editor</a>
<a href="#">Multiple Files Connection Manager</a>	<a href="#">Add File Connection Manager Dialog Box UI Reference</a>

CONNECTION MANAGER	OPTIONS
Flat File Connection Manager	<a href="#">Flat File Connection Manager Editor (General Page)</a> <a href="#">Flat File Connection Manager Editor (Columns Page)</a> <a href="#">Flat File Connection Manager Editor (Advanced Page)</a> <a href="#">Flat File Connection Manager Editor (Preview Page)</a>
Multiple Flat Files Connection Manager	<a href="#">Multiple Flat Files Connection Manager Editor (General Page)</a> <a href="#">Multiple Flat Files Connection Manager Editor (Columns Page)</a> <a href="#">Multiple Flat Files Connection Manager Editor (Advanced Page)</a> <a href="#">Multiple Flat Files Connection Manager Editor (Preview Page)</a>
FTP Connection Manager	<a href="#">FTP Connection Manager Editor</a>
HTTP Connection Manager	<a href="#">HTTP Connection Manager Editor (Server Page)</a> <a href="#">HTTP Connection Manager Editor (Proxy Page)</a>
MSMQ Connection Manager	<a href="#">MSMQ Connection Manager Editor</a>
ODBC Connection Manager	<a href="#">ODBC Connection Manager UI Reference</a>
OLE DB Connection Manager	<a href="#">Configure OLE DB Connection Manager</a>
SMO Connection Manager	<a href="#">SMO Connection Manager Editor</a>
SMTP Connection Manager	<a href="#">SMTP Connection Manager Editor</a>
SQL Server Compact Edition Connection Manager	<a href="#">SQL Server Compact Edition Connection Manager Editor (Connection Page)</a> <a href="#">SQL Server Compact Edition Connection Manager Editor (All Page)</a>
WMI Connection Manager	<a href="#">WMI Connection Manager Editor</a>

The connection manager you added will show up under the **Connections Managers** node in the **Solution Explorer**. It will also appear in the **Connection Managers** tab in the **SSIS Designer** window for all the packages in the project. The name of the connection manager in this tab will have a **(project)** prefix in order to differentiate this project level connection manager from the package level connection managers.

- Optionally, right-click the connection manager in the **Solution Explorer** window under **Connection Managers** node (or) in the **Connection Managers** tab of the **SSIS Designer** window, click **Rename**, and then modify the default name of the connection manager.

#### NOTE

In the **Connection Managers** tab of the **SSIS Designer** window, you won't be able to overwrite the **(project)** prefix from the connection manager name. This is by design.

### Add SSIS Connection Manager dialog box

Use the **Add SSIS Connection Manager** dialog box to select the type of connection to add to a package.

To learn more about connection managers, see [Integration Services \(SSIS\) Connections](#).

#### Options

##### Connection manager type

Select a connection type and then click **Add**, or double-click a connection type, to specify connection properties using the editor for each type of connection.

##### Add

Specify connection properties using the editor for each type of connection.

## Create a parameter for a connection manager property

1. In the **Connection Managers** area, right-click the connection manager that you want to create a parameter for and then click **Parameterize**.
2. Configure the parameter settings in the **Parameterize** dialog box. For more information, see [Parameterize Dialog Box](#).

## Delete a connection manager

### Delete a connection manager from a package

1. In SQL Server Data Tools (SSDT), open the Integration Services project that contains the package you want.
2. In Solution Explorer, double-click the package to open it.
3. In SSIS Designer, click the **Control Flow** tab, the **Data Flow** tab, or the **Event Handler** tab to make the **Connection Managers** area available.
4. Right-click the connection manager that you want to delete, and then click **Delete**.

If you delete a connection manager that a package element, such as an Execute SQL task or an OLE DB source, uses, you will experience the following results:

- An error icon appears on the package element that used the deleted connection manager.
- The package fails to validate.
- The package cannot be run.

5. To save the updated package, click **Save Selected Items** on the **File** menu.

### Delete a shared connection manager (project level connection manager)

1. To delete a project-level connection manager, right-click the connection manager under **Connection Managers** node in the **Solution Explorer** window, and then click **Delete**. SQL Server Data Tools displays the following warning message:

### WARNING

When you delete a project connection manager, packages that use the connection manager might not run. You cannot undo this action. Do you want to delete the connection manager?

2. Click OK to delete the connection manager or Cancel to keep it.

### NOTE

You can also delete a project level connection manager from the **Connection Manager** tab of the **SSIS Designer** window opened for any package in the project. You do so by right-clicking the connection manager in the tab and then by clicking **Delete**.

## Set the Properties of a Connection Manager

All connection managers can be configured using the **Properties** window.

Integration Services also provides custom dialog boxes for modifying the different types of connection managers in Integration Services. The dialog box has a different set of options depending on the connection manager type.

### Modify a connection manager using the Properties window

1. In SQL Server Data Tools (SSDT), open the Integration Services project that contains the package you want.
2. In Solution Explorer, double-click the package to open it.
3. In SSIS Designer, click the **Control Flow** tab, the **Data Flow** tab, or the **Event Handler** tab to make the **Connection Managers** area available.
4. Right-click the connection manager and click **Properties**.
5. In the **Properties** window, edit the property values. The **Properties** window provides access to some properties that are not configurable in the standard editor for a connection manager.
6. Click **OK**.
7. To save the updated package, click **Save Selected Items** on the **File** menu.

### Modify a connection manager using a connection manager dialog box

1. In SQL Server Data Tools (SSDT), open the Integration Services project that contains the package you want.
2. In Solution Explorer, double-click the package to open it.
3. In SSIS Designer, click the **Control Flow** tab, the **Data Flow** tab, or the **Event Handler** tab to make the **Connection Managers** area available.
4. In the **Connection Managers** area, double-click the connection manager to open the **Connection Manager** dialog box. For information about specific connection manager types, and the options available for each type, see the following table.

CONNECTION MANAGER	OPTIONS
<a href="#">ADO Connection Manager</a>	<a href="#">Configure OLE DB Connection Manager</a>
<a href="#">ADO.NET Connection Manager</a>	<a href="#">Configure ADO.NET Connection Manager</a>

CONNECTION MANAGER	OPTIONS
<a href="#">Analysis Services Connection Manager</a>	<a href="#">Add Analysis Services Connection Manager Dialog Box UI Reference</a>
<a href="#">Excel Connection Manager</a>	<a href="#">Excel Connection Manager Editor</a>
<a href="#">File Connection Manager</a>	<a href="#">File Connection Manager Editor</a>
<a href="#">Multiple Files Connection Manager</a>	<a href="#">Add File Connection Manager Dialog Box UI Reference</a>
<a href="#">Flat File Connection Manager</a>	<a href="#">Flat File Connection Manager Editor (General Page)</a> <a href="#">Flat File Connection Manager Editor (Columns Page)</a> <a href="#">Flat File Connection Manager Editor (Advanced Page)</a> <a href="#">Flat File Connection Manager Editor (Preview Page)</a>
<a href="#">Multiple Flat Files Connection Manager</a>	<a href="#">Multiple Flat Files Connection Manager Editor (General Page)</a> <a href="#">Multiple Flat Files Connection Manager Editor (Columns Page)</a> <a href="#">Multiple Flat Files Connection Manager Editor (Advanced Page)</a> <a href="#">Multiple Flat Files Connection Manager Editor (Preview Page)</a>
<a href="#">FTP Connection Manager</a>	<a href="#">FTP Connection Manager Editor</a>
<a href="#">HTTP Connection Manager</a>	<a href="#">HTTP Connection Manager Editor (Server Page)</a> <a href="#">HTTP Connection Manager Editor (Proxy Page)</a>
<a href="#">MSMQ Connection Manager</a>	<a href="#">MSMQ Connection Manager Editor</a>
<a href="#">ODBC Connection Manager</a>	<a href="#">ODBC Connection Manager UI Reference</a>
<a href="#">OLE DB Connection Manager</a>	<a href="#">Configure OLE DB Connection Manager</a>
<a href="#">SMO Connection Manager</a>	<a href="#">SMO Connection Manager Editor</a>
<a href="#">SMTP Connection Manager</a>	<a href="#">SMTP Connection Manager Editor</a>
<a href="#">SQL Server Compact Edition Connection Manager</a>	<a href="#">SQL Server Compact Edition Connection Manager Editor (Connection Page)</a> <a href="#">SQL Server Compact Edition Connection Manager Editor (All Page)</a>
<a href="#">WMI Connection Manager</a>	<a href="#">WMI Connection Manager Editor</a>

5. To save the updated package, click **Save Selected Items** on the **File** menu.

## Related Content

- Video, [Leverage Microsoft Attunity Connector for Oracle to enhance Package Performance](#), on [technet.microsoft.com](#)
- Wiki articles, [SSIS Connectivity](#), on [social.technet.microsoft.com](#)
- Blog entry, [Connecting to MySQL from SSIS](#), on [blogs.msdn.com](#).
- Technical article, [Extracting and Loading SharePoint Data in SQL Server Integration Services](#), on [msdn.microsoft.com](#).
- Technical article, [You get "DTS\\_E\\_CANNOTACQUIRECONNECTIONFROMCONNECTIONMANAGER" error message when using Oracle connection manager in SSIS](#), on [support.microsoft.com](#).

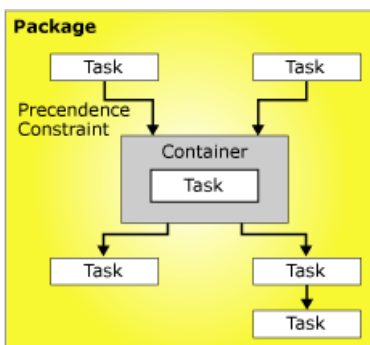
# Control Flow

3/24/2017 • 3 min to read • [Edit Online](#)

A package consists of a control flow and, optionally, one or more data flows. SQL Server Integration Services provides three different types of control flow elements: containers that provide structures in packages, tasks that provide functionality, and precedence constraints that connect the executables, containers, and tasks into an ordered control flow.

For more information, see [Precedence Constraints](#), [Integration Services Containers](#), and [Integration Services Tasks](#).

The following diagram shows a control flow that has one container and six tasks. Five of the tasks are defined at the package level, and one task is defined at the container level. The task is inside a container.



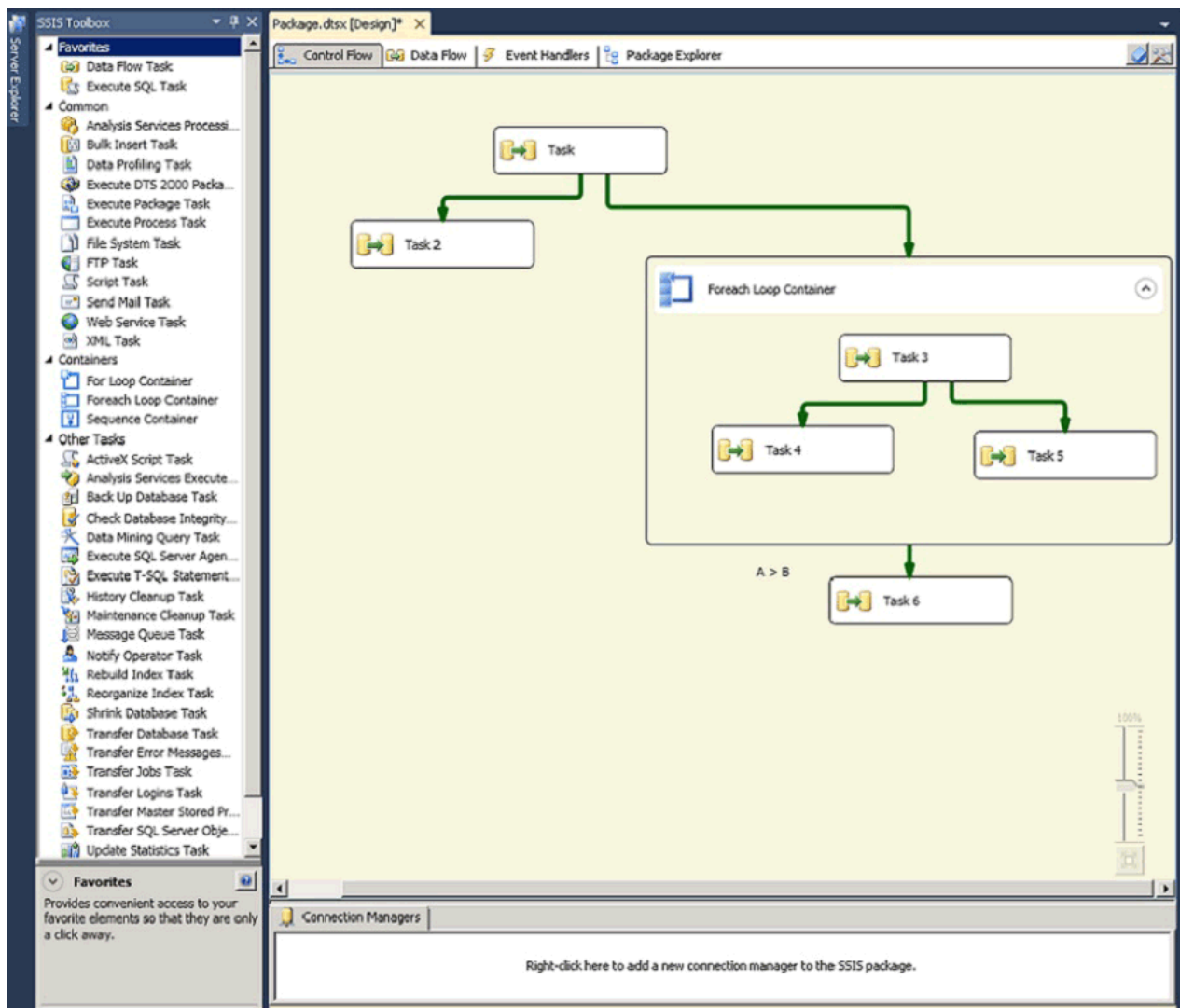
The Integration Services architecture supports the nesting of containers, and a control flow can include multiple levels of nested containers. For example, a package could contain a container such as a Foreach Loop container, which in turn could contain another Foreach Loop container and so on.

Event handlers also have control flows, which are built using the same kinds of control flow elements.

## Control Flow Implementation

You 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 you can add to the control flow.

The following diagram shows the control flow of a simple package in the control flow designer. The control flow shown in the diagram is made up of three package-level tasks and one package-level container that contains three tasks. The tasks and container are connected by using precedence constraints.



Creating a control flow includes the following tasks:

- Adding containers that implement repeating workflows in a package or divide a control flow into subsets.
- Adding tasks that support data flow, prepare data, perform workflow and business intelligence functions, and implement script.

Integration Services includes a variety of tasks that you can use to create control flow that meets the business requirements of the package. If the package has to work with data, the control flow must include at least one Data Flow task. For example, a package might have to extract data, aggregate data values, and then write the results to a data source. For more information, see [Integration Services Tasks](#) and [Add or Delete a Task or a Container in a Control Flow](#).

- Connecting containers and tasks into an ordered control flow by using precedence constraints.

After you add a task or container to the design surface of the **Control Flow** tab, SSIS Designer automatically adds a connector to the item. If a package includes two or more items, tasks or containers, you can join them into a control flow by dragging their connectors from one item to another.

The connector between two items represents a precedence constraint. A precedence constraint defines the relationship between the two connected items. It specifies the order in which tasks and containers are executed at run time and the conditions under which tasks and containers run. For example, a precedence constraint can specify that a task must succeed for the next task in the control flow to run. For more information, see [Precedence Constraints](#).

- Adding connection managers.

Many tasks require a connection to a data source, and you have to add the connection managers that the



task requires to the package. Depending on the enumerator type it uses, the Foreach Loop container may also require a connection manager. You can add the connection managers as you construct the control flow item by item or before you start to construct the control flow. For more information, see [Integration Services \(SSIS\) Connections](#) and [Create Connection Managers](#).

SSIS Designer also includes many design-time features that you can use to manage the design surface and make the control flow self-documenting.

## Related Tasks

- [Add or Delete a Task or a Container in a Control Flow](#)
- [Set the Properties of a Task or Container](#)
- [Group or Ungroup Components](#)

# Data Flow

3/24/2017 • 14 min to read • [Edit Online](#)

SQL Server Integration Services provides three different types of data flow components: sources, transformations, and destinations. Sources extract data from data stores such as tables and views in relational databases, files, and Analysis Services databases. Transformations modify, summarize, and clean data. Destinations load data into data stores or create in-memory datasets.

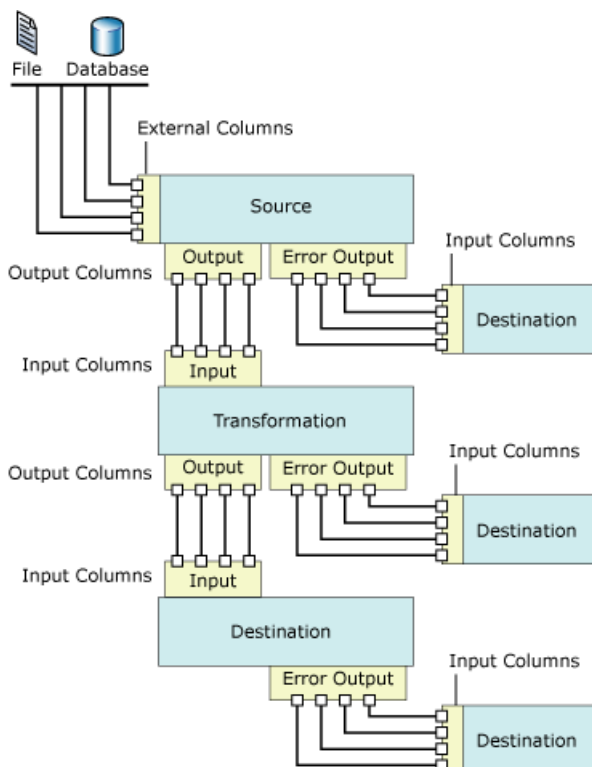
## NOTE

When you use custom providers, you need to update the ProviderDescriptors.xml file with the metadata column values.

Additionally, Integration Services provides paths that connect the output of one component to the input of another component. Paths define the sequence of components, and let you add annotations to the data flow or view the source of the column.

You connect data flow components by connecting the output of sources and destinations to the input of transformations and destinations. When constructing a data flow you typically connect the second and subsequent components as you add them to the data flow. After you connect the component, the input columns are available for use in configuring the component. When no input columns are available, you will have to complete the configuration of the component after it is connected to the data flow. For more information, see [Integration Services Paths](#) and [Connect Components with Paths](#).

The following diagram shows a data flow that has a source, a transformation with one input and one output, and a destination. The diagram includes the inputs, outputs, and error outputs in addition to the input, output, and external columns.



## Data Flow Implementation

Adding a Data Flow task to the control flow of a package is the first step in implementing a data flow in a package. A package can include multiple Data Flow tasks, each with its own data flow. For example, if a package requires that data flows be run in a specified sequence, or that other tasks be performed between the data flows, you must use a separate Data Flow task for each data flow.

After the control flow includes a Data Flow task, you can begin to build the data flow that a package uses. For more information, see [Data Flow Task](#).

Creating a data flow includes the following steps:

- Adding one or more sources to extract data from files and databases, and add connection managers to connect to the sources.
- Adding the transformations that meet the business requirements of the package. A data flow is not required to include transformations.

Some transformations require a connection manager. For example, the Lookup transformation uses a connection manager to connect to the database that contains the lookup data.

- Connecting data flow components by connecting the output of sources and transformations to the input of transformations and destinations.
- Adding one or more destinations to load data into data stores such as files and databases, and adding connection managers to connect to the data sources.
- Configuring error outputs on components to handle problems.

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 cause errors, and the rows that contain the errors can be processed separately using an error flow. For more information about how to use error flows in package data flow, see [Error Handling in Data](#).

- Include annotations to make the data flow self-documenting. For more information, see [Use Annotations in Packages](#).

#### NOTE

When you create a new package, you can also use a wizard to help you configure connection managers, sources, and destinations correctly. For more information, see [Create Packages in SQL Server Data Tools](#).

When the **Data Flow** tab is active, the Toolbox contains the sources, transformations, and destinations that you can add to the data flow.

## Expressions

A number of the data flow components—sources, transformations, and destinations—support the use of property expressions in some of their properties. A property expression is an expression that replaces the value of the property when the package is loaded. At run time, the package uses the updated property values. The expressions are built using the Integration Services expression syntax and can include Integration Services functions, operators, identifiers, and variables. For more information, see [Integration Services \(SSIS\) Expressions](#), [Integration Services \(SSIS\) Expressions](#), and [Use Property Expressions in Packages](#).

If you construct a package in SQL Server Data Tools (SSDT), the properties of any data flow components that support property expressions are exposed on the Data Flow task to which they belong. To add, change, and remove the property expressions of data flow components, click the Data Flow task, and then use the Properties window or the editor for the task to add, change, or delete property expressions. Property expressions for the

Data Flow task itself are managed in the Properties window.

If the data flow contains any components that use expressions, the expressions are also exposed in the Properties window. To view expressions, select the Data Flow task to which the component belongs. You can view properties by categories, or in alphabetical order. If you use the categorized view in the Properties window, any expressions that are not used in a specific property are listed in the **Misc** category. If you use the alphabetical view, expressions are listed in order of the name of the data flow component.

## Sources

In Integration Services, a source is the data flow component that makes data from different external data sources available to the other components in the data flow. You can extract data from flat files, XML files, Microsoft Excel workbooks, and files that contain raw data. You can also extract data by accessing tables and views in databases and by running queries.

A data flow can include a single source or multiple sources.

The source for a data flow typically has one regular output. The regular output contains output columns, which are columns the source adds to the data flow.

The regular output references external columns. An external column is a column in the source. For example, the **MadeFlag** column in the **Product** table of the **AdventureWorks** database is an external column that can be added to the regular output. Metadata for external columns includes such information as the name, data type, and length of the source column.

An error output for a source contains the same columns as the regular output, and also contains two additional columns that provide information about errors. The Integration Services object model does not restrict the number of regular outputs and error outputs that sources can have. Most of the sources that Integration Services includes, except the Script component, have one regular output, and many of the sources have one error output. Custom sources can be coded to implement multiple regular outputs and error outputs.

All the output columns are available as input columns to the next data flow component in the data flow.

You can also write custom sources. For more information, see [Developing a Custom Data Flow Component](#) and [Developing Specific Types of Data Flow Components](#).

The following sources have properties that can be updated by property expressions:

- [ADO NET Source](#)
- [XML Source](#)

### Sources Available for Download

The following table lists additional sources that you can download from the Microsoft website.

SOURCE	DESCRIPTION
Oracle Source	The Oracle source is the source component of the Microsoft Connector for Oracle by Attunity. The Microsoft Connector for Oracle by Attunity also includes a connection manager and a destination. For more information, see the download page, <a href="#">Microsoft Connectors for Oracle and Teradata by Attunity</a> .

SOURCE	DESCRIPTION
SAP BI Source	The SAP BI source is the source component of the Microsoft Connector for SAP BI. The Microsoft Connector for SAP BI also includes a connection manager and a destination. For more information, see the download page, <a href="#">Microsoft SQL Server Feature Pack</a> .
Teradata Source	The Teradata source is the source component of the Microsoft Connector for Teradata by Attunity. The Microsoft Connector for Teradata by Attunity also includes a connection manager and a destination. For more information, see the download page, <a href="#">Microsoft Connectors for Oracle and Teradata by Attunity</a> .

For a demonstration on how to leverage the performance gains of the Microsoft Connector for Oracle by Attunity, see [Performance of Microsoft Connector for Oracle by Attunity \(SQL Server Video\)](#).

## Transformations

The capabilities of transformations vary broadly. Transformations can perform tasks such as updating, summarizing, cleaning, merging, and distributing data. You can modify values in columns, look up values in tables, clean data, and aggregate column values.

The inputs and outputs of a transformation define the columns of incoming and outgoing data. Depending on the operation performed on the data, some transformations have a single input and multiple outputs, while other transformations have multiple inputs and a single output. Transformations can also include error outputs, which provide information about the error that occurred, together with the data that failed: For example, string data that could not be converted to an integer data type. The Integration Services object model does not restrict the number of inputs, regular outputs, and error outputs that transformations can contain. You can create custom transformations that implement any combination of multiple inputs, regular outputs, and error outputs.

The input of a transformation is defined as one or more input columns. Some Integration Services transformations can also refer to external columns as input. For example, the input to the OLE DB Command transformation includes external columns. An output column is a column that the transformation adds to the data flow. Both regular outputs and error outputs contain output columns. These output columns in turn act as input columns to the next component in the data flow, either another transformation or a destination.

The following transformations have properties that can be updated by property expressions:

- [Conditional Split Transformation](#)
- [Derived Column Transformation](#)
- [Fuzzy Grouping Transformation](#)
- [Fuzzy Lookup Transformation](#)
- [OLE DB Command Transformation](#)
- [Percentage Sampling Transformation](#)
- [Pivot Transformation](#)
- [Row Sampling Transformation](#)
- [Sort Transformation](#)
- [Unpivot Transformation](#)

For more information, see [Integration Services Transformations](#).

## Destinations

A destination is the data flow component that writes the data from a data flow to a specific data store, or creates an in-memory dataset. You can load data into flat files, process analytic objects, and provide data to other processes. You can also load data by accessing tables and views in databases and by running queries.

A data flow can include multiple destinations that load data into different data stores.

An Integration Services destination must have at least one input. The input contains input columns, which come from another data flow component. The input columns are mapped to columns in the destination.

Many destinations also have one error output. The error output for a destination contains output columns, which typically contain information about errors that occur when writing data to the destination data store. Errors occur for many different reasons. For example, a column may contain a null value, whereas the destination column cannot be set to null.

The Integration Services object model does not restrict the number of regular inputs and error outputs that destinations can have, and you can create custom destinations that implement multiple inputs and error outputs.

You can also write custom destinations. For more information, see [Developing a Custom Data Flow Component](#) and [Developing Specific Types of Data Flow Components](#).

The following destinations have properties that can be updated by property expressions:

- [Flat File Destination](#)
- [SQL Server Compact Edition Destination](#)

### Destinations Available for Download

The following table lists additional destinations that you can download from the Microsoft website.

SOURCE	DESCRIPTION
Oracle Destination	The Oracle destination is the destination component of the Microsoft Connector for Oracle by Attunity. The Microsoft Connector for Oracle by Attunity also includes a connection manager and a source. For more information, see the download page, <a href="#">Microsoft Connectors for Oracle and Teradata by Attunity</a> .
SAP BI Destination	The SAP BI destination is the destination component of the Microsoft Connector for SAP BI. The Microsoft Connector for SAP BI also includes a connection manager and a source. For more information, see the download page, <a href="#">Microsoft SQL Server Feature Pack</a> .
Teradata Destination	The Teradata destination is the destination component of the Microsoft Connector for Teradata by Attunity. The Microsoft Connector for Teradata by Attunity also includes a connection manager and a source. For more information, see the download page, <a href="#">Microsoft Connectors for Oracle and Teradata by Attunity</a> .

For a demonstration on how to leverage the performance gains of the Microsoft Connector for Oracle by Attunity, see [Performance of Microsoft Connector for Oracle by Attunity \(SQL Server Video\)](#).

## Connection Managers

Many data flow components connect to data sources, and you must add the connection managers that the components require to the package before the component can be configured correctly. You can add the connection managers as you construct the data flow, or before you start to construct the data flow. For more information, see [Integration Services \(SSIS\) Connections](#) and [Create Connection Managers](#).

## External Metadata

When you create a data flow in a package using SSIS Designer, the metadata from the sources and destinations is copied to the external columns on sources and destinations, serving as a snapshot of the schema. When Integration Services validates the package, SSIS Designer compares this snapshot against the schema of the source or destination, and posts errors and warnings, depending on the changes.

The Integration Services project provides an offline mode. When you work offline no connections are made to the sources or destinations the package uses, and the metadata of external columns is not updated.

## Inputs and Outputs

Sources have outputs, destinations have inputs, and transformations have both inputs and outputs. Additionally, many data flow components can be configured to use an error output.

### Inputs

Destinations and transformations have inputs. An input contains one or more input columns, which can refer to external columns if the data flow component has been configured to use them. Inputs can be configured to monitor and control the flow of data: For example, you can specify if the component should fail in response to an error, ignore errors, or redirect error rows to the error output. You can also assign a description to the input or update the input name. In SSIS Designer, inputs are configured by using the **Advanced Editor** dialog box. For more information about the **Advanced Editor**, see [Integration Services User Interface](#).

### Outputs

Sources and transformations always have outputs. An output contains one or more output columns, which can refer to external columns if the data flow component has been configured to use them. Outputs can be configured to provide information useful to downstream processing of the data. For example, you can indicate whether the output is sorted. You can also provide a description for the output, or update the output name. In SSIS Designer, outputs are configured by using the **Advanced Editor** dialog box.

### Error Outputs

Sources, destinations, and transformations can include error outputs. You can specify how the data flow component responds to errors in each input or column by using the **Configure Error Output** dialog box. If an error or data truncation occurs at run time and the data flow component is configured to redirect rows, the data rows with the error are sent to the error output. The error output can be connected to transformations that apply additional transformations or direct data to a different destination. By default, an error output contains the output columns and two error columns: **ErrorCode** and **ErrorColumn**. The output columns contain the data from the row that failed, **ErrorCode** provides the error code, and **ErrorColumn** identifies the failing column.

For more information, see [Error Handling in Data](#).

### Columns

Inputs, outputs, and error outputs are collections of columns. Each column is configurable and depending on the column type—input, output, or external—Integration Services provides different properties for the column. Integration Services provides three different ways of setting column properties: programmatically, by using component-specific dialog boxes, or by using the **Advanced Editor** dialog box.

## Paths

Paths connect data flow components. In SSIS Designer, you can view and modify the path properties, view the output metadata for the path start point, and attach data viewers to a path.

For more information, see [Integration Services Paths](#) and [Debugging Data Flow](#).

## Configuration of Data Flow Components

Data flow components can be configured at the component level; at the input, output, and error output levels; and at the column level.

- At the component level, you set properties that are common to all components, and you set the custom properties of the component.
- At the input, output, and error output levels, you set the common properties of inputs, outputs, and the error output. If the component supports multiple outputs, you can add outputs.
- At the column level, you set the properties that are common to all columns, in addition to any custom properties that the component provides for columns. If the component supports the addition of output columns, you can add columns to outputs.

You can set properties through SSIS Designer or programmatically. In SSIS Designer, you can set element properties using the custom dialog boxes provided for each element type, or by using the Properties window or the **Advanced Editor** dialog box.

For more information about how to set properties by using SSIS Designer, see [Set the Properties of a Data Flow Component](#).

## Related Tasks

[Add or Delete a Component in a Data Flow](#)

[Connect Components in a Data Flow](#)

## Related Content

Video, [Performance of Microsoft Connector for Oracle by Attunity \(SQL Server Video\)](#), on [technet.microsoft.com](https://technet.microsoft.com).



# Integration Services (SSIS) Variables

4/19/2017 • 16 min to read • [Edit Online](#)

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. The precedence constraints that sequence tasks and containers into a workflow can use variables when their constraint definitions include expressions.

You can use variables in Integration Services packages for the following purposes:

- Updating properties of package elements at run time. For example, you 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, you 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.

## System and user-defined variables

Integration Services supports two types of variables: user-defined variables and system variables. User-defined variables are defined by package developers, and system variables are defined by Integration Services. You can create as many user-defined variables as a package requires, but you cannot create additional system variables.

All variables—system and user-defined—can be used in the parameter bindings that the Execute SQL task uses to map variables to parameters in SQL statements. For more information, see [Execute SQL Task](#) and [Parameters and Return Codes in the Execute SQL Task](#).

### NOTE

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

You can create user-defined variables for all Integration Services container types: packages, Foreach Loop containers, For Loop containers, Sequence containers, tasks, and event handlers. User-defined variables are members of the Variables collection of the container.

If you create the package using SSIS Designer, you can see the members of the Variables collections in the **Variables** folders on the **Package Explorer** tab of SSIS Designer. The folders list user-defined variables and system variables.

You can configure user-defined variables in the following ways:

- Provide a name and description for the variable.
- Specify a namespace for the variable.
- Indicate whether the variable raises an event when its value changes.
- Indicate whether the variable is read-only or read/write.
- Use the evaluation result of an expression to set the variable value.
- Create the variable in the scope of the package or a package object such as a task.
- Specify the value and data type of the variable.

The only configurable option on system variables is specifying whether they raise an event when they change value.

A different set of system variables is available for different container types. For more information about the system variables used by packages and their elements, see [System Variables](#).

For more information about real-life use scenarios for variables, see [Use Variables in Packages](#).

## Properties of variables

You can configure user-defined variables by setting the following properties in either the **Variables** window or the **Properties** window. Certain properties are available only in the Properties window.

### NOTE

The only configurable option on system variables is specifying whether they raise an event when they change value.

#### Description

Specifies the description of the variable.

#### EvaluateAsExpression

When the property is set to **True**, the expression provided is used to set the variable value.

#### Expression

Specifies the expression that is assigned to the variable.

#### Name

Specifies the variable name.

#### Namespace

Integration Services provides two namespaces, **User** and **System**. By default, custom variables are in the **User** namespace, and system variables are in the **System** namespace. You can create additional namespaces for user-defined variables and change the name of the **User** namespace, but you cannot change the name of the **System** namespace, add variables to the **System** namespace, or assign system variables to a different namespace.

#### RaiseChangedEvent

When the property is set to **True**, the **OnVariableValueChanged** event is raised when the variable changes value.

#### ReadOnly

When the property is set to **False**, the variable is read\write.

#### Scope

#### NOTE

You can change this property setting only by clicking **Move Variable** in the **Variables** window.

A variable is created within the scope of a package or within the scope of a container, task, or event handler in the package. Because the package container is at the top of the container hierarchy, variables with package scope function like global variables and can be used by all containers in the package. Similarly, variables defined within the scope of a container such as a For Loop container can be used by all tasks or containers within the For Loop container.

If a package runs other packages by using the Execute Package task, the variables defined in the scope of the calling package or the Execute Package task can be made available to the called package by using the Parent Package Variable configuration type. For more information, see [Package Configurations](#).

#### IncludeInDebugDump

Indicate whether the variable value is included in the debug dump files.

For user-defined variables and system variables, the default value for the **IncludeInDebugDump** option is **true**.

However, for user-defined variables, the system resets the **IncludeInDebugDump** option to **false** when the following conditions are met:

- If the **EvaluateAsExpression** variable property is set to **true**, the system resets the **IncludeInDebugDump** option to **false**.

To include the text of the expression as the variable value in the debug dump files, set the **IncludeInDebugDump** option to **true**.

- If the variable data type is changed to a string, the system resets the **IncludeInDebugDump** option to **false**.

When the system resets the **IncludeInDebugDump** option to **false**, this might override the value selected by the user.

#### Value

The value of a user-defined variable can be a literal or an expression. A variable includes options for setting the variable value and the data type of the value. The two properties must be compatible: for example, the use of a string value together with an integer data type is not valid.

If the variable is configured to evaluate as an expression, you must provide an expression. At run time, the expression is evaluated, and the variable is set to the evaluation result. For example, if a variable uses the expression `DATEPART("month", GETDATE())` the value of the variable is the number equivalent of the month for the current date. The expression must be a valid expression that uses the SSIS expression grammar syntax. When an expression is used with variables, the expression can use literals and the operators and functions that the expression grammar provides, but the expression cannot reference the columns from a data flow in the package. The maximum length of an expression is 4000 characters. For more information, see [Integration Services \(SSIS\) Expressions](#).

#### ValueType

#### NOTE

The property value appears in the **Data type** column in the **Variables** window.

Specifies the data type of the variable value.

# Scenarios for using variables

Variables are used in many different ways in Integration Services packages. You will probably find that package development does not progress far before you have to add a user-defined variable to your package to implement the flexibility and manageability your solution requires. Depending on the scenario, system variables are also commonly used.

**Property Expressions** Use variables to provide values in the property expressions that set the properties of packages and package objects. For example, the expression, `SELECT * FROM @varTableName` includes the variable `varTableName` that updates the SQL statement that an Execute SQL task runs. The expression, `DATEPART("d", GETDATE()) == 1? @[User::varPackageFirst]:@[User::varPackageOther]`, updates the package that the Execute Package task runs, by running the package specified in the `varPackageFirst` variable on the first day of the month and running the package specified in the `varPackageOther` variable on other days. For more information, see [Use Property Expressions in Packages](#).

**Data Flow Expressions** Use variables to provide values in the expressions that the Derived Column and Conditional Split transformations use to populate columns, or to direct data rows to different transformation outputs. For example, the expression, `@varSalutation + LastName`, concatenates the value in the `VarSalutation` variable and the `LastName` column. The expression, `Income < @HighIncome`, directs data rows in which the value of the `Income` column is less than the value in the `HighIncome` variable to an output. For more information, see [Derived Column Transformation](#), [Conditional Split Transformation](#), and [Integration Services \(SSIS\) Expressions](#).

**Precedence Constraint Expressions** Provide values to use in precedence constraints to determine whether a constrained executable runs. The expressions can be used either together with an execution outcome (success, failure, completion), or instead of an execution outcome. For example, if the expression, `@varMax > @varMin`, evaluates to **true**, the executable runs. For more information, see [Add Expressions to Precedence Constraints](#).

**Parameters and Return Codes** Provide values to input parameters, or store the values of output parameters and return codes. You do this by mapping the variables to parameters and return values. For example, if you set the variable `varProductId` to 23 and run the SQL statement, `SELECT * from Production.Product WHERE ProductID = ?`, the query retrieves the product with a `ProductID` of 23. For more information, see [Execute SQL Task](#) and [Parameters and Return Codes in the Execute SQL Task](#).

**For Loop Expressions** Provide values to use in the initialization, evaluation, and assignment expressions of the For Loop. For example, if the variable `varCount` is 2 and `varMaxCount` is 10, the initialization expression is `@varCount`, the evaluation expression is `@varCount < @varMaxCount`, and the assignment expression is `@varCount =@varCount +1`, then the loop repeats 8 times. For more information, see [For Loop Container](#).

**Parent Package Variable Configurations** Pass values from parent packages to child packages. Child packages can access variables in the parent package by using parent package variable configurations. For example, if the child package must use the same date as the parent package, the child package can define a parent package variable configuration that specifies a variable set by the GETDATE function in the parent package. For more information, see [Execute Package Task](#) and [Package Configurations](#).

**Script Task and Script Component** Provide a list of read-only and read/write variable to the Script task or Script component, update the read/write variables within the script, and then use the updated values in or outside the script. For example, in the code, `numberOfCars = CType(Dts.Variables("NumberOfCars").Value, Integer)`, the script variable `numberOfCars` is updated by the value in the variable, `NumberOfCars`. For more information, see [Using Variables in the Script Task](#).

## Add a variable

1. In SQL Server Data Tools (SSDT), open the Integration Services package you want to work with.
2. In Solution Explorer, double-click the package to open it.

3. In SSIS Designer, to define the scope of the variable, do one of the following:
  - To set the scope to the package, click anywhere on the design surface of the **Control Flow** tab.
  - To set the scope to an event handler, select an executable and an event handler on the design surface of the **Event Handler** tab.
  - To set the scope to a task or container, on the design surface of the **Control Flow** tab or the **Event Handler** tab, click a task or container.
4. On the **SSIS** menu, click **Variables**. You can optionally display the **Variables** window by mapping the View.Variables command to a key combination of your choosing on the **Keyboard** page of the **Options** dialog box.
5. In the **Variables** window, click the **Add Variable** icon. The new variable is added to the list.
6. Optionally, click the **Grid Options** icon, select additional columns to show in the **Variables Grid Options** dialog box, and then click **OK**.
7. Optionally, set the variable properties. For more information, see [Set the Properties of a User-Defined Variable](#).
8. To save the updated package, click **Save Selected Items** on the **File** menu.

### Add Variable dialog box

Use the **Add Variable** dialog box to specify the properties of a new variable.

#### Options

##### Container

Select a container in the list. The container defines the scope of the variable. The container can be either the package or an executable in the package.

##### Name

Type the variable name.

##### Namespace

Specify the namespace of the variable. By default, user-defined variables are in the **User** namespace.

##### Value type

Select a data type.

##### Value

Type a value. The value must be compatible with the data type specified in the **Value type** option.

##### Read-only

Select to make the variable read-only.

## Delete a variable

1. In SQL Server Data Tools (SSDT), open the Integration Services project that contains the package you want.
2. In Solution Explorer, right-click the package to open it.
3. On the **SSIS** menu, click **Variables**. You can optionally display the **Variables** window by mapping the View.Variables command to a key combination of your choosing on the **Keyboard** page of the **Options** dialog box.
4. Select the variable to delete, and then click **Delete Variable**.

If you don't see the variable in the Variables window, click **Grid Options** and then select **Show variables**

of all scopes.

5. If the **Confirm Deletion of Variables** dialog box opens, click **Yes** to confirm.
6. To save the updated package, click **Save Selected Items** on the **File** menu.

## Change the scope of a variable

1. In SQL Server Data Tools (SSDT), open the Integration Services project that contains the package you want.
2. In Solution Explorer, right-click the package to open it.
3. On the **SSIS** menu, click **Variables**. You can optionally display the **Variables** window by mapping the View.Variables command to a key combination of your choosing on the **Keyboard** page of the **Options** dialog box.
4. Select the variable and then click **Move Variable**.

If you don't see the variable in the Variables window, click **Grid Options** and then select **Show variables of all scopes**.

5. In the **Select New Scope** dialog box, select the package or a container, task, or event handler in the package, to change the variable scope.
6. To save the updated package, click **Save Selected Items** on the **File** menu.

## Set the properties of a user-defined variable

To set the properties of a user-defined variable in Integration Services, you can use one of the following features:

- Variables window.
- Properties window. The **Properties** window lists properties for configuring variables that are not available in the **Variables** window: Description, EvaluateAsExpression, Expression, ReadOnly, ValueType, and IncludeInDebugDump.

### NOTE

Integration Services also provides a set of system variables whose properties cannot be updated, with the exception of the RaiseChangedEvent property.

### Set expressions on variables

When you use the **Properties** window to set expressions on a user-defined variable:

- The value of a variable can be set by the Value or the Expression property. By default, the EvaluateAsExpression property is set to **False** and the value of the variable is set by the Value property. To use an expression to set the value, you must first set EvaluateAsExpression to **True**, and then provide an expression in the Expression property. The Value property is automatically set to the evaluation result of the expression.
- The ValueType property contains the data type of the value in the Value property. When Value is set by an expression, ValueType is automatically updated to a data type that is compatible with the evaluation result of the expression. For example, if Value contains 0 and ValueType property contains **Int32** and you then set Expression to GETDATE(), Value contains the current date and time and ValueType is set to **DateTime**.
- The **Properties** window for the variable provides access to the **Expression Builder** dialog box. You can use this tool to build, validate, and evaluate expressions. For more information, see [Expression Builder](#) and [Integration Services \(SSIS\) Expressions](#).

When you use the **Variables** window to set expressions on a user-defined variable:

- To use an expression to set the variable value, first confirm that the variable data type is compatible with the evaluation result of the expression and then provide an expression in the **Expression** column of the **Variables** window. The **EvaluateAsExpression** property in the **Properties** window is automatically set to **True**.
- When you assign an expression to a variable, a special icon marker displays next to the variable. This special icon marker also displays next to connection managers and tasks that have expressions set on them.
- The **Variables** window for the variable provides access to the **Expression Builder** dialog box. You can use this tool to build, validate, and evaluate expressions. For more information, see [Expression Builder](#) and [Integration Services \(SSIS\) Expressions](#).

In both the **Variables** and **Properties** window, if you assign an expression to the variable, and **EvaluateAsExpression** is set to **True**, you cannot change the variable data type.

### Set the Namespace and Name properties

The values of the **Name** and **Namespace** properties must begin with an alphabetic character letter as defined by the Unicode Standard 2.0, or an underscore (`_`). *Subsequent characters can be letters or numbers as defined in the Unicode Standard 2.0, or the underscore (`_`).*

### Set Variable Properties in the Variables Window

1. In SQL Server Data Tools (SSDT), open the Integration Services project that contains the package you want.
2. In Solution Explorer, right-click the package to open it.
3. On the **SSIS** menu, click **Variables**.

You can optionally display the **Variables** window by mapping the `View.Variables` command to a key combination of your choosing on the **Keyboard** page of the **Options** dialog box.

4. Optionally, in the **Variables** window click **Grid Options**, and then select the columns to appear in the **Variables** window and select the filters to apply to the list of variables.
5. Select the variable in the list, and then update values in the **Name**, **Data Type**, **Value**, **Namespace**, **Raise Change Event**, **Description**, and **Expression** columns.
6. Select the variable in the list, and then click **Move Variable** to change the scope.
7. To save the updated package, on the **File** menu, click **Save Selected Items**.

### Set Variable Properties in the Properties Window

1. In SQL Server Data Tools (SSDT), open the Integration Services project that contains the package you want.
2. In Solution Explorer, right-click the package to open it.
3. On the **View** menu, click **Properties Window**.
4. In SSIS Designer, click the **Package Explorer** tab and expand the Package node.
5. To modify variables with package scope, expand the Variables node; otherwise, expand the Event Handlers or Executables nodes until you locate the Variables node that contains the variable that you want to modify.
6. Click the variable whose properties you want to modify.
7. In the **Properties** window, update the read/write variable properties. Some properties are read/read only for user-defined variables.

For more information about the properties, see [Integration Services \(SSIS\) Variables](#).

8. To save the updated package, on the **File** menu, click **Save Selected Items**.

## Update a variable dynamically with configurations

To dynamically update variables, you can create configurations for the variables, deploy the configurations with the package, and then update the variable values in the configuration file when you deploy the packages. At run time, the package uses the updated variable values. For more information, see [Create Package Configurations](#).

## Related Tasks

[Use the Values of Variables and Parameters in a Child Package](#)

[Map Query Parameters to Variables in a Data Flow Component](#)



# Variables Window

3/24/2017 • 3 min to read • [Edit Online](#)

Use the **Variables** window to create and modify user-defined variables and view system variables.

By default, the **Variables** window is located below the **Connection Managers** area in the SSIS Designer, in SQL Server Data Tools (SSDT). If you don't see the **Variables** window, click **Variables** on the **SSIS** menu to display the window.

You can optionally display the **Variables** window by mapping the View.Variables command to a key combination of your choosing on the **Keyboard** page of the **Options** dialog box.

## NOTE

The values of the **Name** and **Namespace** properties must begin with an alphabetic character letter as defined by the Unicode Standard 2.0, or an underscore (). *Subsequent characters can be letters or numbers as defined in the Unicode Standard 2.0, or the underscore (\).*

## Options

### Add Variable

Add a user-defined variable.

### Move Variable

Click a variable in the list, and then click **Move Variable** to change the variable scope. In the **Select New Scope** dialog box, select the package or a container, task, or event handler in the package, to change the variable scope.

For more information about variable scope, see [Integration Services \(SSIS\) Variables](#).

### Delete Variable

Select a variable from the list, and then click **Delete Variable**.

### Grid Options

Click to open the **Variable Grid Options** dialog box where you can change the column selection and apply filters to the **Variables** window. For more information, see [Variable Grid Options](#).

### Name

View the variable name. You can update the name for user-defined variables.

### Scope

View the scope of the variable. A variable has either the scope of the entire package, or the scope of a container or task. The scope of the variable must be sufficient so that the variable is visible to any other tasks or components that need to read or set its value.

You can change the scope by clicking the variable and then clicking **Move Variable** in the **Variables** window.

### Data Type

View the data type of the variable. You can select a data type from the list for user-defined variables.

## NOTE

If you assign an expression to the variable, you cannot change the data type.

## Value

View the variable value. You can update the value for user-defined variables. This value can be a literal or an expression, and the value can be a multi-line string. To assign an expression to the variable, click the ellipse button that is next to the **Expression** column in the **Variables** window.

## Namespace

View the namespace name. User-defined variables are initially created in the **User** namespace, but you can change the namespace name in the **Namespace** field. To display this column, click **Grid Options**.

## Raise Change Event

Indicate whether to raise the **OnVariableValueChanged** event when a value changes. You can update the value for user-defined and system variables. By default, the **Variables** window does not list this column. To display this column, click **Grid Options**.

## Description

View the variable description. You can change the description for user-defined variables. By default, the **Variables** window does not list this column. To display this column, click **Grid Options**.

## Expression

View the expression assigned to the variable. To assign an expression, click the ellipse button.

If you assign an expression to a variable, a special icon marker displays next to the variable. This special icon marker also displays next to connection managers and tasks that have expressions set on them.

# Variable Grid Options dialog box

Use the **Variable Grid Options** dialog box to select the columns that will display in the **Variables** window and to select the filters to apply to the list of variables. For more information about the corresponding variable properties, see [Integration Services \(SSIS\) Variables](#).

## Options for Filter

### Show system variables

Select to list system variables in the **Variables** window. System variables are predefined. You cannot add or delete system variables. You can modify the **RaiseChangedEvent** property setting.

This list is color coded. System variables are gray, and user-defined variables are black.

### Show variables of all scopes

Select to show variables within the scope the package, and within the scope of containers, tasks, and event handlers in the package. Clear this option to show only variables within the scope of the package and within the scope of a selected container, task, or event handler.

For more information about variable scope, see [Integration Services \(SSIS\) Variables](#).

## Options for Columns

Select the columns that you want to appear in the **Variables** window.

- **Scope**
- **Data type**
- **Value**
- **Namespace**
- **Raise event when variable value changes**
- **Description**

- **Expression**

## See Also

[Integration Services \(SSIS\) Variables](#)

[Use Variables in Packages](#)

[Integration Services \(SSIS\) Expressions](#)

[Generating Dump Files for Package Execution](#)

# System Variables

3/24/2017 • 5 min to read • [Edit Online](#)

SQL Server Integration Services provides a set of system variables that store information about the running package and its objects. These variables can be used in expressions and property expressions to customize packages, containers, tasks, and event handlers.

All variables—system and user-defined— can be used in the parameter bindings that the Execute SQL task uses to map variables to parameters.

## System Variables for Packages

The following table describes the system variables that Integration Services provides for packages.

SYSTEM VARIABLE	DATA TYPE	DESCRIPTION
<b>CancelEvent</b>	Int32	The handle to a Windows Event object that the task can signal to indicate that the task should stop running.
<b>ContainerStartTime</b>	DateTime	The start time of the container.
<b>CreationDate</b>	DateTime	The date that the package was created.
<b>CreatorComputerName</b>	String	The computer on which the package was created.
<b>CreatorName</b>	String	The name of the person who built the package.
<b>ExecutionInstanceGUID</b>	String	The unique identifier of the executing instance of a package.
<b>FailedConfigurations</b>	String	The names of package configurations that have failed.
<b>IgnoreConfigurationsOnLoad</b>	Boolean	Indicates whether package configurations are ignored when loading the package.
<b>InteractiveMode</b>	Boolean	Indicates whether the package is run in interactive mode. If a package is running in SSIS Designer, this property is set to <b>True</b> . If a package is running using the <b>DTEXEC</b> command prompt utility, the property is set to <b>False</b> .
<b>LocaleId</b>	Int32	The locale that the package uses.
<b>MachineName</b>	String	The name of the computer on which the package is running.

SYSTEM VARIABLE	DATA TYPE	DESCRIPTION
<b>OfflineMode</b>	Boolean	Indicates whether the package is in offline mode. Offline mode does not acquire connections to data sources.
<b>PackageID</b>	String	The unique identifier of the package.
<b>PackageName</b>	String	The name of the package.
<b>StartTime</b>	DateTime	The time that the package started to run.
<b>ServerExecutionID</b>	Int64	<p>Execution ID for the package that is executed on the Integration Services server.</p> <p>The default value is zero. The value is changed only if the package is executed by ISServerExec on the Integration Services Server. When there is a child package, the value is passed from the parent package to child package.</p>
<b>UserName</b>	String	The account of the user who started the package. The user name is qualified by the domain name.
<b>VersionBuild</b>	Int32	The package version.
<b>VersionComment</b>	String	Comments about the package version.
<b>VersionGUID</b>	String	The unique identifier of the version.
<b>VersionMajor</b>	Int32	The major version of the package.
<b>VersionMinor</b>	Int32	The minor version of the package.

## System Variables for Containers

The following table describes the system variables that Integration Services provides for the For Loop, Foreach Loop, and Sequence containers.

SYSTEM VARIABLE	DATA TYPE	DESCRIPTION	CONTAINER
<b>LocaleId</b>	Int32	The locale that the container uses.	For Loop container  Foreach Loop container  Sequence container

## System Variables for Tasks

The following table describes the system variables that Integration Services provides for tasks.

SYSTEM VARIABLE	DATA TYPE	DESCRIPTION
<b>CreationName</b>	String	The name of the task.
<b>LocaleId</b>	Int32	The locale that the task uses.
<b>TaskID</b>	String	The unique identifier of a task instance.
<b>TaskName</b>	String	The name of the task instance.
<b>TaskTransactionOption</b>	Int32	The transaction option that the task uses.

## System Variables for Event Handlers

The following table describes the system variables that Integration Services provides for event handlers. Not all variables are available to all event handlers.

SYSTEM VARIABLE	DATA TYPE	DESCRIPTION	EVENT HANDLER
<b>Cancel</b>	Boolean	Indicates whether the event handler stops running when an error, warning, or query cancellation occurs.	OnError event handler  OnWarning event handler  OnQueryCancel event handler
<b>ErrorCode</b>	Int32	The error identifier.	OnError event handler  OnInformation event handler  OnWarning event handler
<b>ErrorDescription</b>	String	The description of the error.	OnError event handler  OnInformation event handler  OnWarning event handler
<b>ExecutionStatus</b>	Boolean	The current execution status.	OnExecStatusChanged event handler
<b>ExecutionValue</b>	DBNull	The execution value.	OnTaskFailed event handler
<b>LocaleId</b>	Int32	The locale that the event handler uses.	All event handlers
<b>PercentComplete</b>	Int32	The percentage of completed work.	OnProgress event handler

SYSTEM VARIABLE	DATA TYPE	DESCRIPTION	EVENT HANDLER
<b>ProgressCountHigh</b>	Int32	The high part of a 64-bit value that indicates the total number of operations processed by the OnProgress event.	OnProgress event handler
<b>ProgressCountLow</b>	Int32	The low part of a 64-bit value that indicates the total number of operations processed by the OnProgress event.	OnProgress event handler
<b>ProgressDescription</b>	String	Description of the progress.	OnProgress event handler
<b>Propagate</b>	Boolean	Indicates whether the event is propagated to a higher level event handler.  Note: The value of the <b>Propagate</b> variable is disregarded during the validation of the package. If you set <b>Propagate</b> to <b>False</b> in a child package, this does not prevent an event from propagating up to the parent package.	All event handlers
<b>SourceDescription</b>	String	The description of the executable in the event handler that raised the event.	All event handlers
<b>SourceID</b>	String	The unique identifier of the executable in the event handler that raised the event.	All event handlers
<b>SourceName</b>	String	The name of the executable in the event handler that raised the event.	All event handlers
<b>VariableDescription</b>	String	The variable description.	OnVariableValueChanged event handler
<b>VariableID</b>	String	The unique identifier of the variable.	OnVariableValueChanged event handler

## System Variables in Parameter Bindings

It is frequently useful to save the values of system variables in tables when the package is run. For example, a package that dynamically creates a table and writes the GUID of the package execution instance that created the table in a table column.

If you use system variables to map to parameters in the SQL statement that an Execute SQL task uses, it is important that you set the data type of each parameter binding to the data type of the system variable. Otherwise,

the values of system variables may be translated incorrectly. For example, if the **ExecutionInstanceGUID** system variable, which has the string data type and contains a string that represents the GUID of the executing instance of a package, is used in a parameter binding with the GUID data type, the GUID of the package instance will be translated incorrectly.

This rule applies to user-defined variables as well. But, whereas the data types of system variables cannot be changed and you have to tailor your use of these variables to fit the data types, user-defined are more flexible. The user-defined variables that are used in parameter bindings are usually defined with data types that are compatible with the data types of parameters to which they are mapped.

## Related Tasks

[Map Query Parameters to Variables in an Execute SQL Task](#)



# Integration Services (SSIS) Expressions

3/24/2017 • 3 min to read • [Edit Online](#)

An expression is a combination of symbols—identifiers, literals, functions, and operators—that yields a single data value. Simple expressions can be a single constant, variable, or function. More frequently, expressions are complex, using multiple operators and functions and referencing multiple columns and variables. In Integration Services, expressions can be used to define conditions for CASE statements, create and update values in data columns, assign values to variables, update or populate properties at run time, define constraints in precedence constraints, and provide the expressions used by the For Loop container.

Expressions are based on an expression language, and the expression evaluator. The expression evaluator parses the expression and determines whether the expression follows the rules of the expression language. For more information about the expression syntax and supported literals and identifiers, see the following topics.

- [Syntax \(SSIS\)](#)
- [Literals \(SSIS\)](#)
- [Identifiers \(SSIS\)](#)

## Components that Use Expressions

The following elements in Integration Services can use expressions:

- The Conditional Split transformation implements a decision structure based on expressions to direct data rows to different destinations. Expressions used in a Conditional Split transformation must evaluate to **true** or **false**. For example, rows that meet the condition in the expression "Column1 > Column2" can be routed to a separate output.
- The Derived Column transformation uses values created by using expressions either to populate new columns in a data flow, or to update existing columns. For example, the expression `Column1 + "ABC"` can be used to update a value or to create a new value with the concatenated string.
- Variables use an expression to set their value. For example, `GETDATE()` sets the value of the variable to the current date.
- Precedence constraints can use expressions to specify the conditions that determine whether the constrained task or container in a package runs. Expressions used in a precedence constraint must evaluate to **true** or **false**. For example, the expression `@A > @B` compares two user-defined variables to determine whether the constrained task runs.
- The For Loop container can use expressions to build the initialization, evaluation, and the incrementing statements that the looping structure uses. For example, the expression `@Counter = 1` initializes the loop counter.

Expressions can also be used to update the values of properties of packages, containers such as the For Loop and Foreach Loop, tasks, package and project level connection managers, log providers, and Foreach enumerators. For example, using a property expression, the string "localhost.AdventureWorks" can be assigned to the `ConnectionString` property of the Execute SQL task. For more information, see [Use Property Expressions in Packages](#).

## Icon Markers for Expressions

In SQL Server Data Tools (SSDT), a special icon marker displays next to connection managers, variables, and tasks that have expressions set on them. The **HasExpressions** property is available on all SSIS objects that support expressions, with the exception of variables. The property enables you to easily identify which objects have expressions.

## Expression Builder

The expression builder is a graphical tool for building expressions. It is available in the **Conditional Split Transformation Editor**, **Derived Column Transformation Editor** dialog boxes, and in the **Expression Builder** dialog box, is a graphical tool for building expressions.

The expression builder provides folders that contain package-specific elements, and folders that contain the functions, type casts, and operators that the expression language provides. The package-specific elements include system variables and user-defined variables. In the **Conditional Split Transformation Editor** and **Derived Column Transformation Editor** dialog boxes, you can also view data columns. To build expressions for the transformations, you can drag items from the folders to the **Condition** or **Expression** column or you can type the expression directly in the column. The expression builder automatically adds needed syntax elements such as the @ prefix on variable names.

### NOTE

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

Variables have scope, and the **Variables** folder in the expression builder lists only variables that are in scope and available to use. For more information, see [Integration Services \(SSIS\) Variables](#).

## Related Tasks

[Use an Expression in a Data Flow Component](#)

## Related Content

Technical article, [SSIS Expression Examples](#), on social.technet.microsoft.com

## See Also

[SQL Server Integration Services](#)

# Integration Services (SSIS) Event Handlers

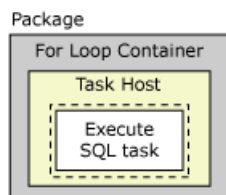
3/24/2017 • 6 min to read • [Edit Online](#)

At run time, executables (packages and Foreach Loop, For Loop, Sequence, and task host containers) raise events. For example, an **OnError** 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:

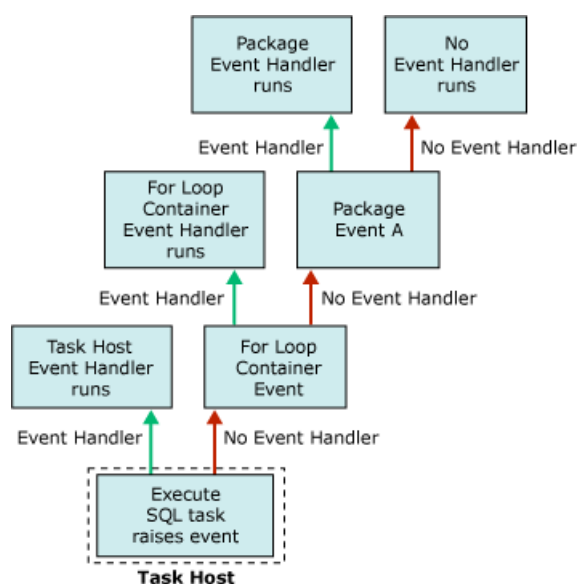
- Clean up temporary data storage when a package or task finishes running.
- Retrieve system information to assess resource availability before a package runs.
- Refresh data in a table when a lookup in a reference table fails.
- Send an e-mail message when an error or a warning occurs or when a task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

If an event has no event handler, the event is raised to the next container up the container hierarchy in a package. If this container has an event handler, the event handler runs in response to the event. If not, the event is raised to the next container up the container hierarchy.

The following diagram shows a simple package that has a For Loop container that contains one Execute SQL task.



Only the package has an event handler, for its **OnError** event. If an error occurs when the Execute SQL task runs, the **OnError** event handler for the package runs. The following diagram shows the sequence of calls that causes the **OnError** event handler for the package to execute.



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.

## Event Handler Content

Creating an event handler is similar to building a package; an event handler has tasks and containers, which are sequenced into a control flow, and an event handler can also include data flows. The SSIS Designer includes the **Event Handlers** tab for creating custom event handlers.

You can also create event handlers programmatically. For more information, see [Handling Events Programmatically](#).

## Run-Time Events

The following table lists the event handlers that Integration Services provides, and describes the run-time events that cause the event handler to run.

EVENT HANDLER	EVENT
<b>OnError</b>	The event handler for the <b>OnError</b> event. This event is raised by an executable when an error occurs.
<b>OnExecStatusChanged</b>	The event handler for the <b>OnExecStatusChanged</b> event. This event is raised by an executable when its execution status changes.
<b>OnInformation</b>	The event handler for the <b>OnInformation</b> 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.
<b>OnPostExecute</b>	The event handler for the <b>OnPostExecute</b> event. This event is raised by an executable immediately after it has finished running.
<b>OnPostValidate</b>	The event handler for the <b>OnPostValidate</b> event. This event is raised by an executable when its validation is finished.
<b>OnPreExecute</b>	The event handler for the <b>OnPreExecute</b> event. This event is raised by an executable immediately before it runs.
<b>OnPreValidate</b>	The event handler for the <b>OnPreValidate</b> event. This event is raised by an executable when its validation starts.
<b>OnProgress</b>	The event handler for the <b>OnProgress</b> event. This event is raised by an executable when measurable progress is made by the executable.

EVENT HANDLER	EVENT
<b>OnQueryCancel</b>	The event handler for the <b>OnQueryCancel</b> event. This event is raised by an executable to determine whether it should stop running.
<b>OnTaskFailed</b>	The event handler for the <b>OnTaskFailed</b> event. This event is raised by a task when it fails.
<b>OnVariableValueChanged</b>	The event handler for the <b>OnVariableValueChanged</b> 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 <b>RaiseChangeEvent</b> property for the variable to <b>False</b> . For more information, see <a href="#">Integration Services (SSIS) Variables</a> .
<b>OnWarning</b>	The event handler for the <b>OnWarning</b> event. This event is raised by an executable when a warning occurs.

## Add an event handler to a package

At run time, containers and tasks raise events. You can create custom event handlers that respond to these events by running a workflow when the event is raised. For example, you can create an event handler that sends an e-mail message when a task fails.

An event handler is similar to a package. Like a package, an event handler can provide scope for variables, and includes a control flow and optional data flows. You can build event handlers for packages, the Foreach Loop container, the For Loop container, the Sequence container, and all tasks.

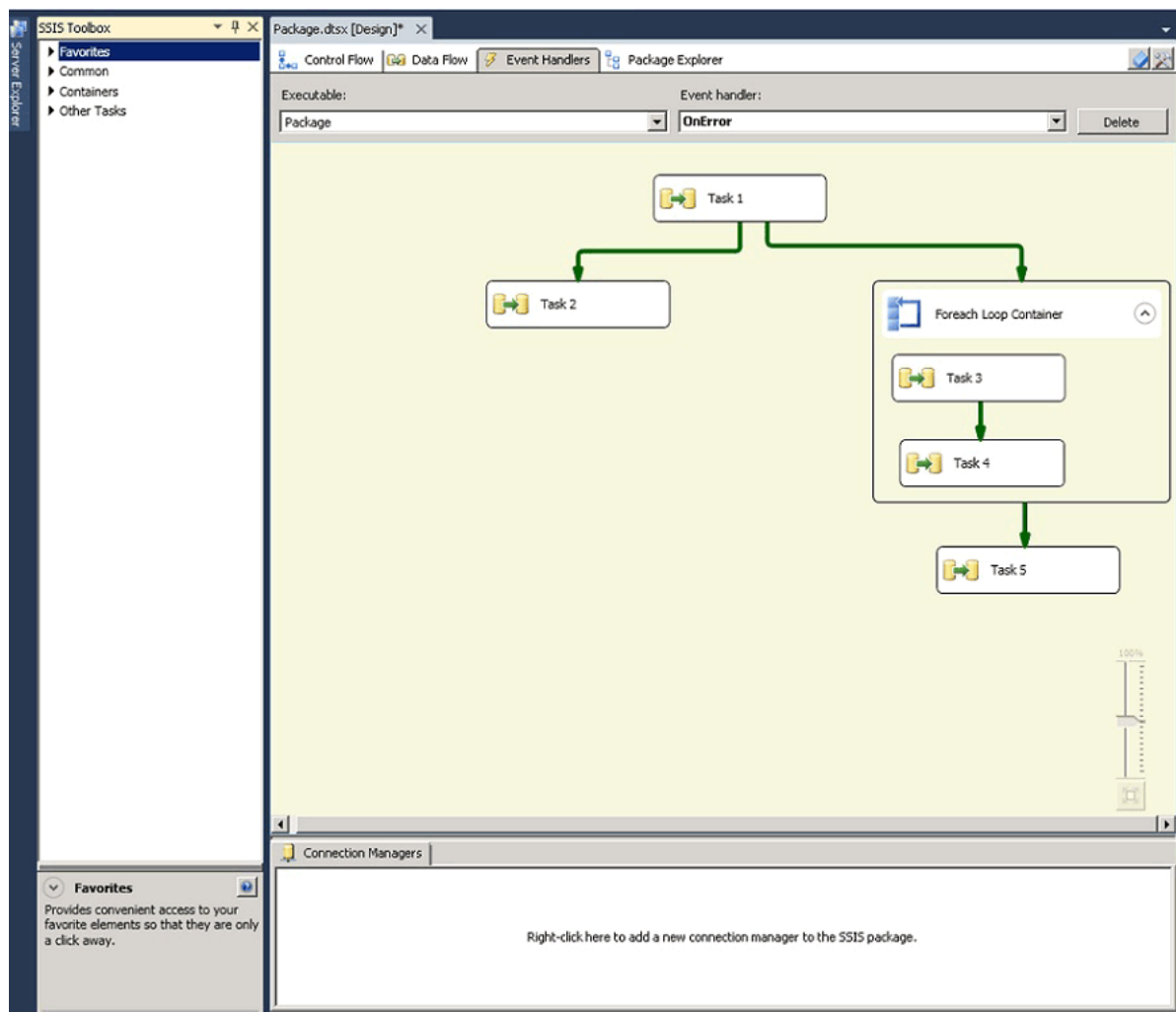
You create event handlers by using the design surface of the **Event Handlers** tab in SSIS Designer.

When the **Event Handlers** tab is active, the **Control Flow Items** and **Maintenance Plan Tasks** nodes of the Toolbox in SSIS Designer contain the task and containers for building the control flow in the event handler. The **Data Flow Sources, Transformations, and Data Flow Destinations** nodes contain the data sources, transformations, and destinations for building the data flows in the event handler. For more information, see [Control Flow](#) and [Data Flow](#).

The **Event Handlers** tab also includes the **Connections** Managers area where you can create and modify the connection managers that event handlers use to connect to servers and data sources. For more information, see [Create Connection Managers](#).

### Add an event handler on the Event Handlers tab

1. In SQL Server Data Tools (SSDT), open the Integration Services project that contains the package you want.
2. In Solution Explorer, double-click the package to open it.
3. Click the **Event Handlers** tab.



Creating the control flow and data flows in an event handler is similar to creating the control flow and data flows in a package. For more information, see [Control Flow](#) and [Data Flow](#).

4. In the **Executable** list, select the executable for which you want to create an event handler.
5. In the **Event handler** list, select the event handler you want to build.
6. Click the link on the design surface of the **Event Handler** tab.
7. Add control flow items to the event handler, and connect items using a precedence constraint by dragging the constraint from one control flow item to another. For more information, see [Control Flow](#).
8. Optionally, add a Data Flow task, and on the design surface of the **Data Flow** tab, create a data flow for the event handler. For more information, see [Data Flow](#).
9. On the **File** menu, click **Save Selected Items** to save the package.

## Set the properties of an event handler

You can set properties in the **Properties** window of SQL Server Data Tools (SSDT) or programmatically.

For information about how to set these properties in SQL Server Data Tools (SSDT), see [Set the Properties of a Task or Container](#).

For information about programmatically setting these properties, see [DtsEventHandler](#).

## Related Tasks

For information about how to add an event handler to a package, see [Add an Event Handler to a Package](#).

# Integration Services (SSIS) Queries

3/24/2017 • 6 min to read • [Edit Online](#)

The Execute SQL task, the OLE DB source, the OLE DB destination, and the Lookup transformation can use SQL queries. In the Execute SQL task, the SQL statements can create, update, and delete database objects and data; run stored procedures; and perform SELECT statements. In the OLE DB source and the Lookup transformation, the SQL statements are typically SELECT statements or EXEC statements. The latter most frequently run stored procedures that return result sets.

A query can be parsed to establish whether it is valid. When parsing a query that uses a connection to SQL Server, the query is parsed, executed, and the execution outcome (success or failure) is assigned to the parsing outcome. If the query uses a connection to a data other than SQL Server, the statement is parsed only.

You can provide the SQL statement in the following ways:

1. Enter it directly in the designer.
2. Specify a connection to a file contains the statement.
3. Specify a variable that contains the statement.

## Direct Input SQL

Query Builder is available in the user interface for the Execute SQL task, the OLE DB source, the OLE DB destination, and the Lookup transformation. Query Builder offers the following advantages:

- Work visually or with SQL commands.

Query Builder includes graphical panes that compose your query visually and a text pane that displays the SQL text of your query. You can work in either the graphical or text panes. Query Builder synchronizes the views so that the query text and graphical representation always match.

- Join related tables.

If you add more than one table to your query, Query Builder automatically determines how the tables are related and constructs the appropriate join command.

- Query or update databases.

You can use Query Builder to return data using Transact-SQL SELECT statements, or to create queries that update, add, or delete records in a database.

- View and edit results immediately.

You can execute your query and work with a recordset in a grid that lets you scroll through and edit records in the database.

Although Query Builder is visually limited to creating SELECT queries, you can type the SQL for other types of statements such as DELETE and UPDATE statements in the text pane. The graphical pane is automatically updated to reflect the SQL statement that you typed.

You can also provide direct input by typing the query in the task or data flow component dialog box or the Properties window.

For more information, see [Query Builder](#).

# SQL in Files

The SQL statement for the Execute SQL task can also reside in a separate file. For example, you can write queries using tools such as the Query Editor in SQL Server Management Studio, save the query to a file, and then read the query from the file when running a package. The file can contain only the SQL statements to run and comments. To use a SQL statement stored in a file, you must provide a file connection that specifies the file name and location. For more information, see [File Connection Manager](#).

## SQL in Variables

If the source of the SQL statement in the Execute SQL task is a variable, you provide the name of the variable that contains the query. The Value property of the variable contains the query text. You set the ValueType property of the variable to a string data type and then type or copy the SQL statement into the Value property. For more information, see [Integration Services \(SSIS\) Variables](#) and [Use Variables in Packages](#).

## Query Builder dialog box

Use the **Query Builder** dialog box to create a query for use in the Execute SQL task, the OLE DB source and the OLE DB destination, and the Lookup transformation.

You can use Query Builder to perform the following tasks:

- **Working with a graphical representation of a query or with SQL commands** Query Builder includes a pane that displays your query graphically and a pane that displays the SQL text of your query. You can work in either the graphical pane or the text pane. Query Builder synchronizes the views so that they are always current.
- **Joining related tables** If you add more than one table to your query, Query Builder automatically determines how the tables are related and constructs the appropriate join command.
- **Querying or updating databases** You can use Query Builder to return data by using Transact-SQL SELECT statements and to create queries that update, add, or delete records in a database.
- **Viewing and editing results immediately** You can run your query and work with a recordset in a grid that allows you to scroll through and edit records in the database.

The graphical tools in the **Query Builder** dialog box let you construct queries using drag-and-drop operations. By default, the Query Builder dialog box constructs SELECT queries, but you can also build INSERT, UPDATE, or DELETE queries. All types of SQL statements can be parsed and run in the **Query Builder** dialog box. For more information about SQL statements in packages, see [Integration Services \(SSIS\) Queries](#).

To learn more about the Transact-SQL language and its syntax, see [Transact-SQL Reference \(Database Engine\)](#).

You can also use variables in a query to provide values to an input parameter, to capture values of output parameters, and to store return codes. To learn more about using variables in the queries that packages use, see [Execute SQL Task](#), [OLE DB Source](#), and [Integration Services \(SSIS\) Queries](#). To learn more about using variables in the Execute SQL Task, see [Parameters and Return Codes in the Execute SQL Task](#) and [Result Sets in the Execute SQL Task](#).

The Lookup and Fuzzy lookup transformations can also use variables with parameters and return codes. The information about the OLE DB source applies to these two transformations also.

### Options

#### Toolbar

Use the toolbar to manage datasets, select panes to display, and control query functions.



VALUE	DESCRIPTION
Show/Hide Diagram Pane	Shows or hides the <b>Diagram</b> pane.
Show/Hide Grid Pane	Shows or hides the <b>Grid</b> pane.
Show/Hide SQL Pane	Shows or hides the <b>SQL</b> pane.
Show/Hide Results Pane	Shows or hides the <b>Results</b> pane.
Run	Runs the query. Results are displayed in the result pane.
Verify SQL	Verifies that the SQL statement is valid.
Sort Ascending	Sorts output rows on the selected column in the grid pane, in ascending order.
Sort Descending	Sorts output rows on the selected column in the grid pane, in descending order.
Remove Filter	Select a column name in the grid pane, and then click <b>Remove Filter</b> to remove sort criteria for the column.
Use Group By	Adds GROUP BY functionality to the query.
Add Table	Adds a new table to the query.

## Query Definition

The query definition provides a toolbar and panes in which to define and test the query.

PANE	DESCRIPTION
<b>Diagram</b> pane	<p>Displays the query in a diagram. The diagram shows the tables included in the query, and how they are joined. Select or clear the check box next to a column in a table to add or remove it from the query output.</p> <p>When you add tables to the query, Query Builder creates joins between tables based on tables, depending on the keys in the table. To add a join, drag a field from one table onto a field in another table. To manage a join, right-click the join, and then select a menu option.</p> <p>Right-click the <b>Diagram</b> pane to add or remove tables, select all the tables, and show or hide panes.</p>
<b>Grid</b> pane	Displays the query in a grid. You can use this pane to add to and remove columns from the query and change the settings for each column.
<b>SQL</b> pane	Displays the query as SQL text. Changes made in the <b>Diagram</b> pane and the <b>Grid</b> pane will appear here, and changes made here will appear in the <b>Diagram</b> pane and the <b>Grid</b> pane.

PANE	DESCRIPTION
<b>Results</b> pane	Displays the results of the query when you click <b>Run</b> on the toolbar.

# Integration Services Transactions

3/24/2017 • 7 min to read • [Edit Online](#)

Packages use transactions to bind the database actions that tasks perform into atomic units, and by doing this maintain data integrity. All Microsoft Integration Services container types—packages, the For Loop, Foreach Loop, and Sequence containers, and the task hosts that encapsulate each task—can be configured to use transactions. Integration Services provides three options for configuring transactions: **NotSupported**, **Supported**, and **Required**.

- **Required** indicates that the container starts a transaction, unless one is already started by its parent container. If a transaction already exists, the container joins the transaction. For example, if a package that is not configured to support transactions includes a Sequence container that uses the **Required** option, the Sequence container would start its own transaction. If the package were configured to use the **Required** option, the Sequence container would join the package transaction.
- **Supported** indicates that the container does not start a transaction, but joins any transaction started by its parent container. For example, if a package with four Execute SQL tasks starts a transaction and all four tasks use the **Supported** option, the database updates performed by the Execute SQL tasks are rolled back if any task fails. If the package does not start a transaction, the four Execute SQL tasks are not bound by a transaction, and no database updates except the ones performed by the failed task are rolled back.
- **NotSupported** indicates that the container does not start a transaction or join an existing transaction. A transaction started by a parent container does not affect child containers that have been configured to not support transactions. For example, if a package is configured to start a transaction and a For Loop container in the package uses the **NotSupported** option, none of the tasks in the For Loop can roll back if they fail.

You configure transactions by setting the `TransactionOption` property on the container. You can set this property by using the **Properties** window in SQL Server Data Tools (SSDT), or you can set the property programmatically.

## NOTE

The **TransactionOption** property influences whether or not the value of the **IsolationLevel** property requested by a container is applied. For more information, see the description of the **IsolationLevel** property in the topic, [Setting Package Properties](#).

## Configure a package to use transactions

When you configure a package to use transactions, you have two options:

- Have a single transaction for the package. In this case, it is the package itself that *initiates* this transaction, whereas individual tasks and containers in the package participate in this single transaction.
- Have multiple transactions in the package. In this case, the package supports transactions, but individual tasks and containers in the package actually initiate the transactions.

The following procedures describe how to configure both options.

### Configure a package to use a single transaction

In this option, the package itself initiates a single transaction. You configure the package to initiate this transaction by setting the `TransactionOption` property of the package to **Required**.

Next, you enlist specific tasks and containers in this single transaction. To enlist a task or container in a transaction, you set the TransactionOption property of that task or container to **Supported**.

1. In SQL Server Data Tools (SSDT), open the Integration Services project that contains the package you want to configure to use a transaction.
2. In Solution Explorer, double-click the package to open it.
3. Click the **Control Flow** tab.
4. Right-click anywhere in the background of the control flow design surface, and then click **Properties**.
5. In the **Properties** window, set the TransactionOption property to **Required**.
6. On the design surface of the **ControlFlow** tab, right-click the task or the container that you want to enroll in the transaction, and then click **Properties**.
7. In the **Properties** window, set the TransactionOption property to **Supported**.

#### NOTE

To enlist a connection in a transaction, enroll the tasks that use the connection in the transaction. For more information, see [Integration Services \(SSIS\) Connections](#).

8. Repeat steps 6 and 7 for each task and container that you want to enroll in the transaction.

### Configure a package to use multiple transactions

In this option, the package itself supports transactions but does not start a transaction. You configure the package to support transactions by setting the TransactionOption property of the package to **Supported**.

Next, you configure the desired tasks and containers inside the package to initiate or participate in transactions. To configure a task or container to initiate a transaction, you set the TransactionOption property of that task or container to **Required**.

1. In SQL Server Data Tools (SSDT), open the Integration Services project that contains the package you want to configure to use transaction.s
2. In Solution Explorer, double-click the package to open it.
3. Click the **Control Flow** tab.
4. Right-click anywhere in the background of the control flow design surface, and then click **Properties**.
5. In the **Properties** window, set the TransactionOption property to **Supported**.

#### NOTE

The package supports transactions, but the transactions are started by task or containers in the package.

6. On the design surface of the **ControlFlow** tab, right-click the task or the container in the package for which you want to start a transaction, and then click **Properties**.
7. In the **Properties** window, set the TransactionOption property to **Required**.
8. If a transaction is started by a container, right-click the task or the container that you want to enroll in the transaction, and then click **Properties**.
9. In the **Properties** window, set the TransactionOption property to **Supported**.

#### NOTE

To enlist a connection in a transaction, enroll the tasks that use the connection in the transaction. For more information, see [Integration Services \(SSIS\) Connections](#).

10. Repeat steps 6 through 9 for each task and container that starts a transaction.

## Multiple transactions in a package

It is possible for a package to include unrelated transactions in an Integration Services package. Any time a container in the middle of a nested container hierarchy does not support transactions, the containers above or below it in the hierarchy start separate transactions if they are configured to support transactions. The transactions commit or roll back in order from the innermost task in the nested container hierarchy to the package. However, after the inner transaction commits, it does not roll back if an outer transaction is aborted.

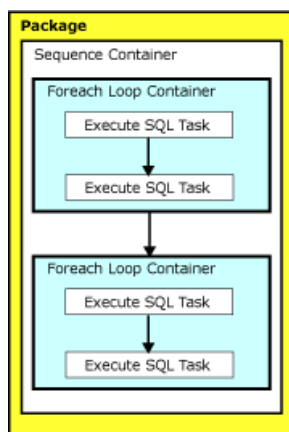
### Example of multiple transactions in a package

For example, a package has a Sequence container that holds two Foreach Loop containers, and each container include two Execute SQL tasks. The Sequence container supports transactions, the Foreach Loop containers do not, and the Execute SQL tasks do. In this example, each Execute SQL task would start its own transaction and would not roll back if the transaction on the Sequence task was aborted.

The TransactionOption properties of the Sequence container, Foreach Loop container and the Execute SQL tasks are set as follows:

- The TransactionOption property of the Sequence container is set to **Required**.
- The TransactionOption properties of the Foreach Loop containers are set to **NotSupported**.
- The TransactionOption properties of the Execute SQL tasks are set to **Required**.

The following diagram shows the five unrelated transactions in the package. One transaction is started by the Sequence container and four transactions are started by the Execute SQL tasks.



## Inherited transactions

A package can run another package by using the Execute Package task. The child package, which is the package run by the Execute Package task, may create its own package transaction, or it may inherit the parent package transaction.

A child package inherits the parent package transaction if both of the following are true:

- The package is invoked by an Execute Package task.
- The Execute Package task that invoked the package also joined the parent package transaction.

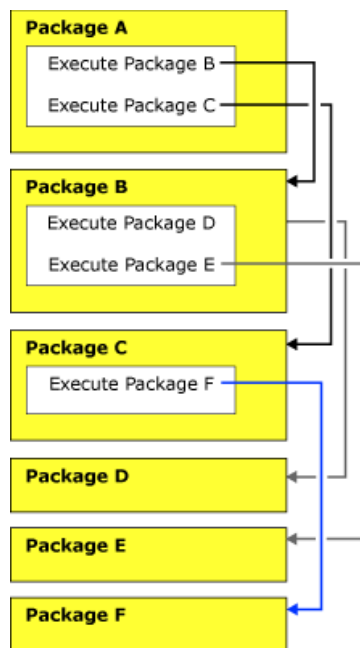
Containers and tasks in the child package cannot join the parent package transaction unless the child package itself joins the transaction.

### Example of inherited transactions

In the following diagram, there are three packages that all use transactions. Each package contains multiple tasks. To emphasize the behavior of the transactions, only the Execute Package tasks are shown. Package A runs packages B and C. In turn, package B runs packages D and E, and package C runs package F.

Packages and tasks have the following transaction attributes:

- **TransactionOption** is set to **Required** on packages A and C
- **TransactionOption** is set to **Supported** on packages B and D, and on the tasks Execute Package B, Execute Package D, and Execute Package F.
- **TransactionOption** is set to **NotSupported** on package E, and on the tasks Execute Package C and Execute Package E.



Only packages B, D, and F can inherit transactions from their parent packages.

Packages B and D inherit the transaction that was started by package A.

Package F inherits the transaction that was started by package C.

Packages A and C control their own transactions.

Package E does not use transactions.

## External Resources

- Blog entry, [How to Use Transactions in SQL Server Integration Services SSIS](#), on [www.mssqltips.com](#)

## See Also

[Inherited Transactions](#)

[Multiple Transactions](#)

# Integration Services (SSIS) Server and Catalog

4/19/2017 • 1 min to read • [Edit Online](#)

After you design and test packages in SQL Server Data Tools, you can deploy the projects that contain the packages to the Integration Services server.

The Integration Services server is an instance of the SQL Server Database Engine that hosts the **SSISDB** database. The database stores the following objects: packages, projects, parameters, permissions, server properties, and operational history.

The **SSISDB** database exposes the object information in public views that you can query. The database also provides stored procedures that you can call to manage the objects.

Before you can deploy the projects to the Integration Services server, you need to create the **SSISDB** catalog.

For an overview of the SSISDB catalog functionality, see [SSIS Catalog](#).

## High Availability

Like other user databases, the **SSISDB** database does support database mirroring and replication. For more information about mirroring and replication, see [Database Mirroring \(SQL Server\)](#).

You can also provide high-availability of SSISDB and its contents by making use of SSIS and Always On Availability Groups. For more information, see this blog post by Matt Masson, [SSIS with Always On](#), at [blogs.msdn.com](#).

## Integration Services Server in SQL Server Management Studio

When you connect to an instance of the SQL Server Database Engine that hosts the **SSISDB** database, you see the following objects in Object Explorer:

- **SSISDB Database**

The **SSISDB** database appears under the **Databases** node in Object Explorer. You can query the views and call the stored procedures that manage the Integration Services server and the objects that are stored on the server.

- **Integration Services Catalogs**

Under the **Integration Services Catalogs** node there are folders for Integration Services projects and environments.

## Related Tasks

- [View the List of Packages on the Integration Services Server](#)
- [Deploy Integration Services \(SSIS\) Projects and Packages](#)
- [Run Integration Services \(SSIS\) Packages](#)

## Related Content

Blog entry, [SSIS with Always On](#), at [blogs.msdn.com](#).

# SSIS Catalog

5/9/2017 • 27 min to read • [Edit Online](#)

The **SSISDB** catalog is the central point for working with Integration Services (SSIS) projects that you've deployed to the Integration Services server. For example, you set project and package parameters, configure environments to specify runtime values for packages, execute and troubleshoot packages, and manage Integration Services server operations.

The objects that are stored in the **SSISDB** catalog include projects, packages, parameters, environments, and operational history.

You inspect objects, settings, and operational data that are stored in the **SSISDB** catalog, by querying the views in the **SSISDB** database. You manage the objects by calling stored procedures in the **SSISDB** database or by using the UI of the **SSISDB** catalog. In many cases, the same task can be performed in the UI or by calling a stored procedure.

To maintain the **SSISDB** database, it is recommended that you apply standard enterprise policies for managing user databases. For information about creating maintenance plans, see [Maintenance Plans](#).

The **SSISDB** catalog and the **SSISDB** database support Windows PowerShell. For more information about using SQL Server with Windows PowerShell, see [SQL Server PowerShell](#). For examples of how to use Windows PowerShell to complete tasks such as deploying a project, see the blog entry, [SSIS and PowerShell in SQL Server 2012](#), on [blogs.msdn.com](#).

For more information about viewing operations data, see [Monitor Running Package and Other Operations](#).

You access the **SSISDB** catalog in SQL Server Management Studio by connecting to the SQL Server Database Engine and then expanding the **Integration Services Catalogs** node in Object Explorer. You access the **SSISDB** database in SQL Server Management Studio by expanding the Databases node in Object Explorer.

## NOTE

You cannot rename the **SSISDB** database.

## NOTE

If the SQL Server instance that the **SSISDB** database is attached to, stops or does not respond, the ISServerExec.exe process ends. A message is written to a Windows Event log.

If the SQL Server resources failover as part of a cluster failover, the running packages do not restart. You can use checkpoints to restart packages. For more information, see [Restart Packages by Using Checkpoints](#).

## Features and capabilities

- [Catalog Object Identifiers](#)
- [Catalog Configuration](#)
- [Permissions](#)
- [Folders](#)
- [Projects and Packages](#)



- [Parameters](#)
- [Server Environments, Server Variables, and Server Environment References](#)
- [Executions and Validations](#)

## Catalog Object Identifiers

When you create a new object in the catalog, assign a name to the object. The object name is an identifier. SQL Server defines rules for which characters can be used in an identifier. Names for the following objects must follow identifier rules.

- Folder
- Project
- Environment
- Parameter
- Environment Variable

### **Folder, Project, Environment**

Consider the following rules when renaming a folder, project, or environment.

- Invalid characters include ASCII/Unicode characters 1 through 31, quote ("), less than (<), greater than (>), pipe (|), backspace (\b), null (\0), and tab (\t).
- The name might not contain leading or trailing spaces.
- @ is not allowed as the first character, but subsequent characters might use @.
- The length of the name must be greater than 0 and less than or equal to 128.

### **Parameter**

Consider the following rules when naming a parameter.

- The first character of the name must be a letter as defined in the Unicode Standard 2.0, or an underscore (\_).
- Subsequent characters can be letters or numbers as defined in the Unicode Standard 2.0, or an underscore (\_).

### **Environment Variable**

Consider the following rules when naming an environment variable.

- Invalid characters include ASCII/Unicode characters 1 through 31, quote ("), less than (<), greater than (>), pipe (|), backspace (\b), null (\0), and tab (\t).
- The name might not contain leading or trailing spaces.
- @ is not allowed as the first character, but subsequent characters might use @.
- The length of the name must be greater than 0 and less than or equal to 128.
- The first character of the name must be a letter as defined in the Unicode Standard 2.0, or an underscore (\_).
- Subsequent characters can be letters or numbers as defined in the Unicode Standard 2.0, or an underscore (\_).

## Catalog Configuration

You fine-tune how the catalog behaves by adjusting the catalog properties. Catalog properties define how sensitive data is encrypted, and how operations and project versioning data is retained. To set catalog properties, use the **Catalog Properties** dialog box or call the [catalog.configure\\_catalog \(SSISDB Database\)](#) stored procedure. To view the properties, use the dialog box or query [catalog.catalog\\_properties \(SSISDB Database\)](#). You access the dialog box by right-clicking **SSISDB** in Object Explorer.

## Operations and Project Version Cleanup

Status data for many of the operations in the catalog is stored in internal database tables. For example, the catalog tracks the status of package executions and project deployments. To maintain the size of the operations data, the **SSIS Server Maintenance Job** in SQL Server Management Studio is used to remove old data. This SQL Server Agent job is created when Integration Services is installed.

You can update or redeploy an Integration Services project by deploying it with the same name to the same folder in the catalog. By default, each time you redeploy a project, the **SSISDB** catalog retains the previous version of the project. To maintain the size of the operations data, the **SSIS Server Maintenance Job** is used to remove old versions of projects.

To run the **SSIS Server Maintenance Job**, SSIS creates the SQL Server login **##MS\_SSISServerCleanupJobLogin##**. This login is only for internal use by SSIS.

The following **SSISDB** catalog properties define how this SQL Server Agent job behaves. You can view and modify the properties by using the **Catalog Properties** dialog box or by using [catalog.catalog\\_properties \(SSISDB Database\)](#) and [catalog.configure\\_catalog \(SSISDB Database\)](#).

### Clean Logs Periodically

The job step for operations cleanup runs when this property is set to **True**.

### Retention Period (days)

Defines the maximum age of allowable operations data (in days). Older data are removed.

The minimum value is one day. The maximum value is limited only by the maximum value of the SQL Server **int** data. For information about this data type, see [int, bigint, smallint, and tinyint \(Transact-SQL\)](#).

### Periodically Remove Old Versions

The job step for project version cleanup runs when this property is set to **True**.

### Maximum Number of Versions per Project

Defines how many versions of a project are stored in the catalog. Older versions of projects are removed.

### Encryption Algorithm

The **Encryption Algorithm** property specifies the type of encryption that is used to encrypt sensitive parameter values. You can choose from the following types of encryption.

- AES\_256 (default)
- AES\_192
- AES\_128
- DESX
- TRIPLE\_DES\_3KEY
- TRIPLE\_DES
- DES

When you deploy an Integration Services project to the Integration Services server, the catalog automatically encrypts the package data and sensitive values. The catalog also automatically decrypts the

data when you retrieve it. The SSISDB catalog uses the **ServerStorage** protection level. For more information, see [Access Control for Sensitive Data in Packages](#).

Changing the encryption algorithm is a time-intensive operation. First, the server has to use the previously specified algorithm to decrypt all configuration values. Then, the server has to use the new algorithm to re-encrypt the values. During this time, there cannot be other Integration Services operations on the server. Thus, to enable Integration Services operations to continue uninterrupted, the encryption algorithm is a read-only value in the dialog box in Management Studio.

To change the **Encryption Algorithm** property setting, set the **SSISDB** database to the single-user mode and then call the catalog.config catalog stored procedure. Use ENCRYPTION\_ALGORITHM for the *property\_name* argument. For the supported property values, see [catalog.catalog\\_properties \(SSISDB Database\)](#). For more information about the stored procedure, see [catalog.config catalog \(SSISDB Database\)](#).

For more information about single-user mode, see [Set a Database to Single-user Mode](#). For information about encryption and encryption algorithms in SQL Server, see the topics in the section, [SQL Server Encryption](#).

A database master key is used for the encryption. The key is created when you create the catalog.

The following table lists the property names shown in the **Catalog Properties** dialog box and the corresponding properties in the database view.

PROPERTY NAME (CATALOG PROPERTIES DIALOG BOX)	PROPERTY NAME (DATABASE VIEW)
Encryption Algorithm Name	ENCRYPTION_ALGORITHM
Clean Logs Periodically	OPERATION_CLEANUP_ENABLED
Retention Period (days)	RETENTION_WINDOW
Periodically Remove Old Versions	VERSION_CLEANUP_ENABLED
Maximum Number of Versions per Project	MAX_PROJECT_VERSIONS
Server-wide Default Logging Level	SERVER_LOGGING_LEVEL

## Permissions

Projects, environments, and packages are contained in folders that are securable objects. You can grant permissions to a folder, including the **MANAGE\_OBJECT\_PERMISSIONS** permission. **MANAGE\_OBJECT\_PERMISSIONS** enables you to delegate the administration of folder contents to a user without having to grant the user membership to the **ssis\_admin** role. You can also grant permissions to projects, environments, and operations. Operations include initializing Integration Services, deploying projects, creating and starting executions, validating projects and packages, and configuring the **SSISDB** catalog.

For more information about database roles, see [Database-Level Roles](#).

The SSISDB catalog uses a DDL trigger, **ddl\_cleanup\_object\_permissions**, to enforce the integrity of permissions information for SSIS securables. The trigger fires when a database principal, such as a database user, database role, or a database application role, is removed from the SSISDB database.

If the principal has granted or denied permissions to other principals, revoke the permissions given by the grantor, before the principal can be removed. Otherwise, an error message is returned when the system tries to remove the principal. The trigger removes all permission records where the database principal is a grantee.

It is recommended that the trigger is not disabled because it ensures that there are no orphaned permission records after a database principal is dropped from the **SSISDB** database.

## Managing Permissions

You can manage permissions by using the SQL Server Management Studio UI, stored procedures, and the [Microsoft.SqlServer.Management.IntegrationServices](#) namespace.

To manage permissions using the SQL Server Management Studio UI, use the following dialog boxes.

- For a folder, use the **Permissions** page of the [Folder Properties Dialog Box](#).
- For a project, use the **Permissions** page in the [Project Properties Dialog Box](#).
- For an environment, use the **Permissions** page in the [NIB: Environment Properties Dialog Box](#).

To manage permissions using Transact-SQL, call [catalog.grant\\_permission \(SSISDB Database\)](#), [catalog.deny\\_permission \(SSISDB Database\)](#) and [catalog.revoke\\_permission \(SSISDB Database\)](#). To view effective permissions for the current principal for all objects, query [catalog.effective\\_object\\_permissions \(SSISDB Database\)](#). This topic provides descriptions of the different types of permissions. To view permissions that have been explicitly assigned to the user, query [catalog.explicit\\_object\\_permissions \(SSISDB Database\)](#).

## Folders

A folder contains one or more projects and environments in the **SSISDB** catalog. You can use the [catalog.folders \(SSISDB Database\)](#) view to access information about folders in the catalog. You can use the following stored procedures to manage folders.

- [catalog.create\\_folder \(SSISDB Database\)](#)
- [catalog.delete\\_folder \(SSISDB Database\)](#)
- [catalog.rename\\_folder \(SSISDB Database\)](#)
- [catalog.set\\_folder\\_description \(SSISDB Database\)](#)

## Projects and Packages

Each project can contain multiple packages. Both projects and packages can contain parameters and references to environments. You can access the parameters and environment references by using the [Configure Dialog Box](#).

You can carry out other project tasks by calling the following stored procedures.

- [catalog.delete\\_project \(SSISDB Database\)](#)
- [catalog.deploy\\_project \(SSISDB Database\)](#)
- [catalog.get\\_project \(SSISDB Database\)](#)
- [\[catalog.move\\_project \(\(SSISDB Database\)\)\]\(../Topic/catalog.move\\_project%20\(\(SSISDB%20Database\).md\)](#)
- [catalog.restore\\_project \(SSISDB Database\)](#)

These views provide details about packages, projects, and project versions.

- [catalog.projects \(SSISDB Database\)](#)
- [catalog.packages \(SSISDB Database\)](#)
- [catalog.object\\_versions \(SSISDB Database\)](#)

# Parameters

You use parameters to assign values to package properties at the time of package execution. To set the value of a package or project parameter and to clear the value, call [catalog.set\\_object\\_parameter\\_value \(SSISDB Database\)](#) and [catalog.clear\\_object\\_parameter\\_value \(SSISDB Database\)](#). To set the value of a parameter for an instance of execution, call [catalog.set\\_execution\\_parameter\\_value \(SSISDB Database\)](#). You can retrieve default parameter values by calling [catalog.get\\_parameter\\_values \(SSISDB Database\)](#).

These views show the parameters for all packages and projects, and parameter values that are used for an instance of execution.

- [catalog.object\\_parameters \(SSISDB Database\)](#)
- [catalog.execution\\_parameter\\_values \(SSISDB Database\)](#)

## Server Environments, Server Variables, and Server Environment References

Server environments contain server variables. The variable values can be used when a package is executed or validated on the Integration Services server.

The following stored procedures enable you to perform many other management tasks for environments and variables.

- [catalog.create\\_environment \(SSISDB Database\)](#)
- [catalog.delete\\_environment \(SSISDB Database\)](#)
- [catalog.move\\_environment \(SSISDB Database\)](#)
- [catalog.rename\\_environment \(SSISDB Database\)](#)
- [catalog.set\\_environment\\_property \(SSISDB Database\)](#)
- [catalog.create\\_environment\\_variable \(SSISDB Database\)](#)
- [catalog.delete\\_environment\\_variable \(SSISDB Database\)](#)
- [catalog.set\\_environment\\_variable\\_property \(SSISDB Database\)](#)
- [catalog.set\\_environment\\_variable\\_value \(SSISDB Database\)](#)

By calling the [catalog.set\\_environment\\_variable\\_protection \(SSISDB Database\)](#) stored procedure, you can set the sensitivity bit for a variable.

To use the value of a server variable, specify the reference between the project and the server environment. You can use the following stored procedures to create and delete references. You can also indicate whether the environment can be located in the same folder as the project or in a different folder.

- [catalog.create\\_environment\\_reference \(SSISDB Database\)](#)
- [catalog.delete\\_environment\\_reference \(SSISDB Database\)](#)
- [catalog.set\\_environment\\_reference\\_type \(SSISDB Database\)](#)

For more details about environments and variables, query these views.

- [catalog.environments \(SSISDB Database\)](#)
- [catalog.environment\\_variables \(SSISDB Database\)](#)
- [catalog.environment\\_references \(SSISDB Database\)](#)

# Executions and Validations

An execution is an instance of a package execution. Call [catalog.create\\_execution \(SSISDB Database\)](#) and [catalog.start\\_execution \(SSISDB Database\)](#) to create and start an execution. To stop an execution or a package/project validation, call [catalog.stop\\_operation \(SSISDB Database\)](#).

To cause a running package to pause and create a dump file, call the `catalog.create_execution_dump` stored procedure. A dump file provides information about the execution of a package that can help you troubleshoot execution issues. For more information about generating and configuring dump files, see [Generating Dump Files for Package Execution](#).

For details about executions, validations, messages that are logged during operations, and contextual information related to errors, query these views.

- [catalog.executions \(SSISDB Database\)](#)
- [catalog.operations \(SSISDB Database\)](#)
- [catalog.operation\\_messages \(SSISDB Database\)](#)
- [catalog.extended\\_operation\\_info \(SSISDB Database\)](#)
- [catalog.event\\_messages](#)
- [catalog.event\\_message\\_context](#)

You can validate projects and packages by calling the [catalog.validate\\_project \(SSISDB Database\)](#) and [catalog.validate\\_package \(SSISDB Database\)](#) stored procedures. The [catalog.validations \(SSISDB Database\)](#) view provides details about validations such as the server environment references that are considered in the validation, whether it is a dependency validation or a full validation, and whether the 32-bit runtime or the 64-bit runtime is used to run the package.

## Create the SSIS Catalog

After you design and test packages in SQL Server Data Tools, you can deploy the projects that contain the packages to an Integration Services server. Before you can deploy the projects to the Integration Services server, the server must contain the **SSISDB** catalog. The installation program for SQL Server 2012 does not automatically create the catalog; you need to manually create the catalog by using the following instructions.

You can create the SSISDB catalog in SQL Server Management Studio. You also create the catalog programmatically by using Windows PowerShell.

### To create the SSISDB catalog in SQL Server Management Studio

1. Open SQL Server Management Studio.
2. Connect to the SQL Server Database Engine.
3. In Object Explorer, expand the server node, right-click the **Integration Services Catalogs** node, and then click **Create Catalog**.
4. Click **Enable CLR Integration**.

The catalog uses CLR stored procedures.

5. Click **Enable automatic execution of Integration Services stored procedure at SQL Server startup** to enable the [catalog.startup](#) stored procedure to run each time the SSIS server instance is restarted.

The stored procedure performs maintenance of the state of operations for the SSISDB catalog. It fixes the status of any packages there were running if and when the SSIS server instance goes down.

6. Enter a password, and then click **Ok**.

The password protects the database master key that is used for encrypting the catalog data. Save the password in a secure location. It is recommended that you also back up the database master key. For more information, see [Back Up a Database Master Key](#).

### To create the SSISDB catalog programmatically

1. Execute the following PowerShell script:

```
# Load the IntegrationServices Assembly
[Reflection.Assembly]::LoadWithPartialName("Microsoft.SqlServer.Management.IntegrationServices")

# Store the IntegrationServices Assembly namespace to avoid typing it every time
$ISNamespace = "Microsoft.SqlServer.Management.IntegrationServices"

Write-Host "Connecting to server ..."

# Create a connection to the server
$sqlConnectionString = "Data Source=localhost;Initial Catalog=master;Integrated Security=SSPI;"
$sqlConnection = New-Object System.Data.SqlClient.SqlConnection $sqlConnectionString

# Create the Integration Services object
$integrationServices = New-Object $ISNamespace".IntegrationServices" $sqlConnection

# Provision a new SSIS Catalog
$catalog = New-Object $ISNamespace".Catalog" ($integrationServices, "SSISDB", "P@assword1")
$catalog.Create()
```

For more examples of how to use Windows PowerShell and the [Microsoft.SqlServer.Management.IntegrationServices](#) namespace, see the blog entry, [SSIS and PowerShell in SQL Server 2012](#), on blogs.msdn.com. For an overview of the namespace and code examples, see the blog entry, [A Glimpse of the SSIS Catalog Managed Object Model](#), on blogs.msdn.com.

## Catalog Properties Dialog Box

Use the Catalog Properties dialog box to configure the SSISDB catalog. Catalog properties define how sensitive data is encrypted, how operations and project versioning data is retained, and when validation operations time out. The SSISDB catalog is a central storage and administration point for Integration Services projects, packages, parameters, and environments.

You can also view catalog properties in the catalog.catalog\_property view, and set the properties by using the catalog.configure\_catalog stored procedure. For more information, see [catalog.catalog\\_properties \(SSISDB Database\)](#) and [catalog.configure\\_catalog \(SSISDB Database\)](#).

### What do you want to do?

- [Open the Catalog Properties Dialog Box](#)
- [Configure the Options](#)

### Open the Catalog Properties Dialog Box

1. Open SQL Server Management Studio.
2. Connect Microsoft SQL Server Database Engine.
3. In Object Explorer, expand the **Integration Services** node, right-click **SSISDB**, and then click **Properties**.

### Configure the Options





#### Options

The following table describes certain properties in the dialog box and the corresponding properties in the

catalog.catalog\_property view.

PROPERTY NAME (CATALOG PROPERTIES DIALOG BOX)	PROPERTY NAME (CATALOG.CATALOG_PROPERTY VIEW)	DESCRIPTION
Encryption Algorithm Name	ENCRYPTION_CLEANUP_ENABLED	<p>Specifies the type of encryption that is used to encrypt the sensitive parameter values in the catalog. The following are the possible values:</p> <p>DES</p> <p>TRIPLE_DES</p> <p>TRIPLE_DES_3KEY</p> <p>DESPX</p> <p>AES_128</p> <p>AES_192</p> <p>AES_256 (default)</p>
Validation Timeout (seconds)	VALIDATION_TIMEOUT	<p>Specify the maximum number of seconds a project validation or a package validation can run before it is stopped. The default value is 300 seconds.</p> <p>Performing the validation is an asynchronous operation. The larger the project or package is, the longer it will take to validate.</p> <p>For information on validating projects and packages, see <a href="#">Integration Services Data Types in Expressions</a>.</p>
Clean Logs Periodically	OPERATION_CLEANUP_ENABLED	<p>Set the property to True to indicate that the SQL Server Agent job, operations cleanup, runs. Otherwise, set the property to False.</p>
Retention Period (days)	RETENTION_WINDOW	<p>Specify the maximum age of allowable operations data (in days). Data that is older than the specified number of days will be removed by the SQL Agent job, operations cleanup.</p>
Maximum Number of Versions per Project	MAX_PROJECT_VERSIONS	<p>Specify how many versions of a project will be stored in the catalog. Older versions of projects that exceed the maximum will be removed when the project version cleanup job runs.</p>

## Back up, Restore, and Move the SSIS Catalog

**THIS TOPIC APPLIES TO:**  SQL Server (starting with 2016)  Azure SQL Database  Azure SQL Data Warehouse  Parallel Data Warehouse

SQL Server 2017 Integration Services (SSIS) includes the SSISDB database. You query views in the SSISDB



database to inspect objects, settings, and operational data that are stored in the **SSISDB** catalog. This topic provides instructions for backing up and restoring the database.

The **SSISDB** catalog stores the packages that you've deployed to the Integration Services server. For more information about the catalog, see [SSIS Catalog](#).

### To Back up the SSIS Database

1. Open SQL Server Management Studio and connect to an instance of SQL Server.
2. Back up the master key for the SSISDB database, by using the BACKUP MASTER KEY Transact-SQL statement. The key is stored in a file that you specify. Use a password to encrypt the master key in the file.

For more information about the statement, see [BACKUP MASTER KEY \(Transact-SQL\)](#).

In the following example, the master key is exported to the `c:\temp\directory\RCTestInstKey` file. The `LS2Setup!` password is used to encrypt the master key.

```
backup master key to file = 'c:\temp\directory\RCTestInstKey'
encryption by password = 'LS2Setup!'
```

3. Back up the SSISDB database by using the **Backup Database** dialog box in SQL Server Management Studio. For more information, see [How to: Back Up a Database \(SQL Server Management Studio\)](#).
4. Generate the CREATE LOGIN script for `##MS_SSISServerCleanupJobLogin##`, by doing the following. For more information, see [CREATE LOGIN \(Transact-SQL\)](#).
  - a. In Object Explorer in SQL Server Management Studio, expand the **Security** node and then expand the **Logins** node.
  - b. Right-click `##MS_SSISServerCleanupJobLogin##`, and then click **Script Login as > CREATE To > New Query Editor Window**.
5. If you will be restoring the SSISDB database to an SQL Server instance where the SSISDB catalog was never created, generate the CREATE PROCEDURE script for `sp_ssis_startup`, by doing the following. For more information, see [CREATE PROCEDURE \(Transact-SQL\)](#).
  - a. In Object Explorer, expand the **Databases** node and then expand the **master > Programmability > Stored Procedures** node.
  - b. Right click `dbo.sp_ssis_startup`, and then click **Script Stored Procedure as > CREATE To > New Query Editor Window**.
6. Confirm that SQL Server Agent has been started
7. If you will be restoring the SSISDB database to an SQL Server instance where the SSISDB catalog was never created, generate a script for the SSIS Server Maintenance Job by doing the following. The script is created in SQL Server Agent automatically when the SSISDB catalog is created. The job helps clean up cleanup operation logs outside the retention window and remove older versions of projects.
  - a. In Object Explorer, expand the **SQL Server Agent** node and then expand the **Jobs** node.
  - b. Right click SSIS Server Maintenance Job, and then click **Script Job as > CREATE To > New Query Editor Window**.

### To Restore the SSIS Database

1. If you are restoring the SSISDB database to an SQL Server instance where the SSISDB catalog was never created, enable common language runtime (clr) by running the `sp_configure` stored procedure. For more information, see [sp\\_configure \(Transact-SQL\)](#) and [clr enabled Option](#).

```
use master
    sp_configure 'clr enabled', 1
reconfigure
```

2. If you are restoring the SSISDB database to an SQL Server instance where the SSISDB catalog was never created, create the asymmetric key and the login from the asymmetric key, and grant UNSAFE permission to the login.

```
Create Asymmetric key MS_SQLEnableSystemAssemblyLoadingKey
FROM Executable File = 'C:\Program Files\Microsoft SQL
Server\110\DTS\Binn\Microsoft.SqlServer.IntegrationServices.Server.dll'
```

Integration Services CLR stored procedures require UNSAFE permissions to be granted to the login because the login requires additional access to restricted resources, such as the Microsoft Win32 API. For more information about the UNSAFE code permission, see [Creating an Assembly](#).

```
Create Login MS_SQLEnableSystemAssemblyLoadingUser
FROM Asymmetric key MS_SQLEnableSystemAssemblyLoadingKey

Grant unsafe Assembly to MS_SQLEnableSystemAssemblyLoadingUser
```

3. Restore the SSISDB database from the backup by using the **Restore Database** dialog box in SQL Server Management Studio. For more information, see the following topics.
  - [Restore Database \(General Page\)](#)
  - [Restore Database \(Files Page\)](#)
  - [Restore Database \(Options Page\)](#)
4. Execute the scripts that you created in the [To Back up the SSIS Database](#) for ##MS\_SSISServerCleanupJobLogin##, sp\_ssis\_startup, and SSIS Server Maintenance Job. Confirm that SQL Server Agent has been started.
5. Run the following statement to set the sp\_ssis\_startup procedure for autoexecution. For more information, see [sp\\_procoption \(Transact-SQL\)](#).

```
EXEC sp_procoption N'sp_ssis_startup','startup','on'
```

6. Map the SSISDB user ##MS\_SSISServerCleanupJobUser## (SSISDB database) to ##MS\_SSISServerCleanupJobLogin##, by using the **Login Properties** dialog box in SQL Server Management Studio.
7. Restore the master key by using one of the following methods. For more information about encryption, see [Encryption Hierarchy](#).

- **Method 1**

Use this method if you've already performed a backup of the database master key, and you have the password used to encrypt the master key.

```
Restore master key from file = 'c:\temp\RCTestInstKey'  
Decryption by password = 'LS2Setup!' -- 'Password used to encrypt the master key during  
SSISDB backup'  
Encryption by password = 'LS3Setup!' -- 'New Password'  
Force
```

#### NOTE

Confirm that the SQL Server service account has permissions to read the backup key file.

#### NOTE

You will see the following warning message displayed in SQL Server Management Studio if the database master key has not yet been encrypted by the service master key. Ignore the warning message.

**The current master key cannot be decrypted. The error was ignored because the FORCE option was specified.**

The FORCE argument specifies that the restore process should continue even if the current database master key is not open. For the SSISDB catalog, because the database master key has not been opened on the instance where you are restoring the database, you will see this message.

#### • Method 2

Use this method if you have the original password that was used to create SSISDB.

```
open master key decryption by password = 'LS1Setup!' --'Password used when creating SSISDB'  
Alter Master Key Add encryption by Service Master Key
```

8. Determine whether the SSISDB catalog schema and the Integration Services binaries (ISServerExec and SQLCLR assembly) are compatible, by running [catalog.check\\_schema\\_version](#).
9. To confirm that the SSISDB database has been restored successfully, perform operations against the SSISDB catalog such as running packages that have been deployed to the Integration Services server. For more information, see [Run Integration Services \(SSIS\) Packages](#).

#### To Move the SSIS Database

- Follow the instructions for moving user databases. For more information, see [Move User Databases](#).

Ensure that you back up the master key for the SSISDB database and protect the backup file. For more information, see [To Back up the SSIS Database](#).

Ensure that the Integration Services (SSIS) relevant objects are created in the new SQL Server instance where the SSISDB catalog has not yet been created.

## Upgrade the SSIS Catalog (SSISDB)

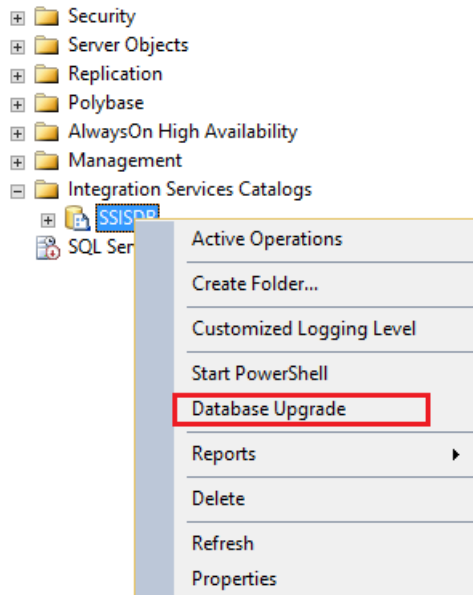
Run the SSISDB Upgrade Wizard to upgrade the SSIS Catalog database, SSISDB, when the database is older than the current version of the SQL Server instance. This occurs when one of the following conditions is true.

- You restored the database from an older version of SQL Server.
- You did not remove the database from an Always On Availability Group before upgrading the SQL Server instance. This prevents the automatic upgrade of the database. For more info, see [Upgrading SSISDB in an availability group](#).

The wizard can only upgrade the database on a local server instance.

### Upgrade the SSIS Catalog (SSISDB) by running the SSISDB Upgrade Wizard

1. Backup the SSIS Catalog database, SSISDB.
2. In SQL Server Management Studio, expand the local server, and then expand **Integration Services Catalogs**.
3. Right-click on **SSISDB**, and then select **Database Upgrade** to launch the SSISDB Upgrade Wizard.

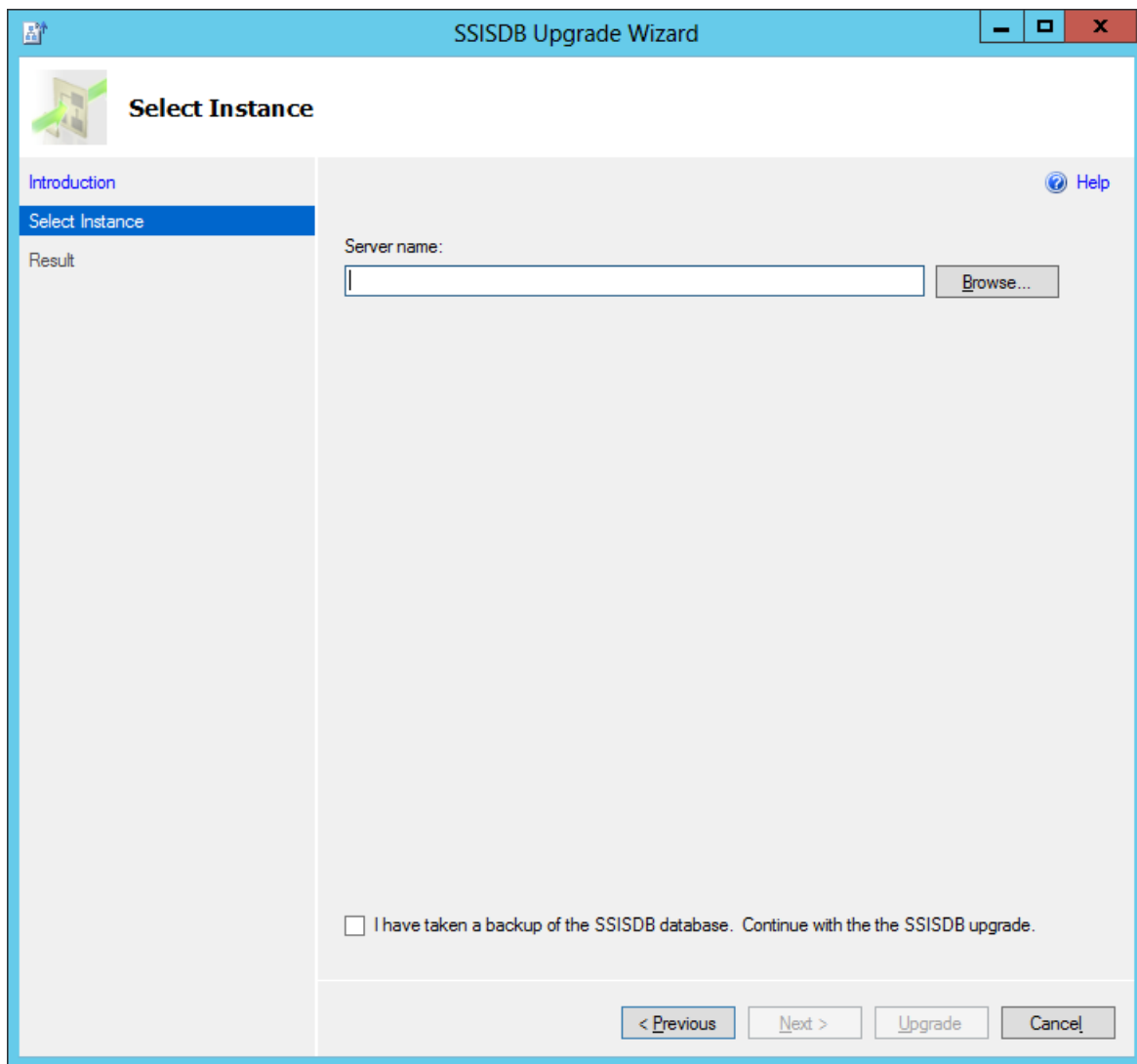


4. On the **Select Instance** page, select a SQL Server instance on the local server.

#### IMPORTANT

The wizard can only upgrade the database on a local server instance.

Select the checkbox to indicate that you have backed up the SSISDB database before running the wizard.



The image shows the 'Select Instance' window of the SSISDB Upgrade Wizard. The window has a blue title bar with the text 'SSISDB Upgrade Wizard' and standard Windows window controls. On the left, there is a sidebar with a tree view containing 'Introduction', 'Select Instance' (which is highlighted), and 'Result'. The main area of the window is light gray. At the top left of this area is a small icon of a server rack. To its right is the title 'Select Instance'. In the top right corner of the main area is a 'Help' link with a question mark icon. Below the title, there is a 'Server name:' label followed by a text input field and a 'Browse...' button. At the bottom of the main area, there is a checkbox with the text 'I have taken a backup of the SSISDB database. Continue with the the SSISDB upgrade.' Below this, there is a row of four buttons: '< Previous', 'Next >', 'Upgrade', and 'Cancel'.

SSISDB Upgrade Wizard

**Select Instance**

Introduction  
Select Instance  
Result

Help

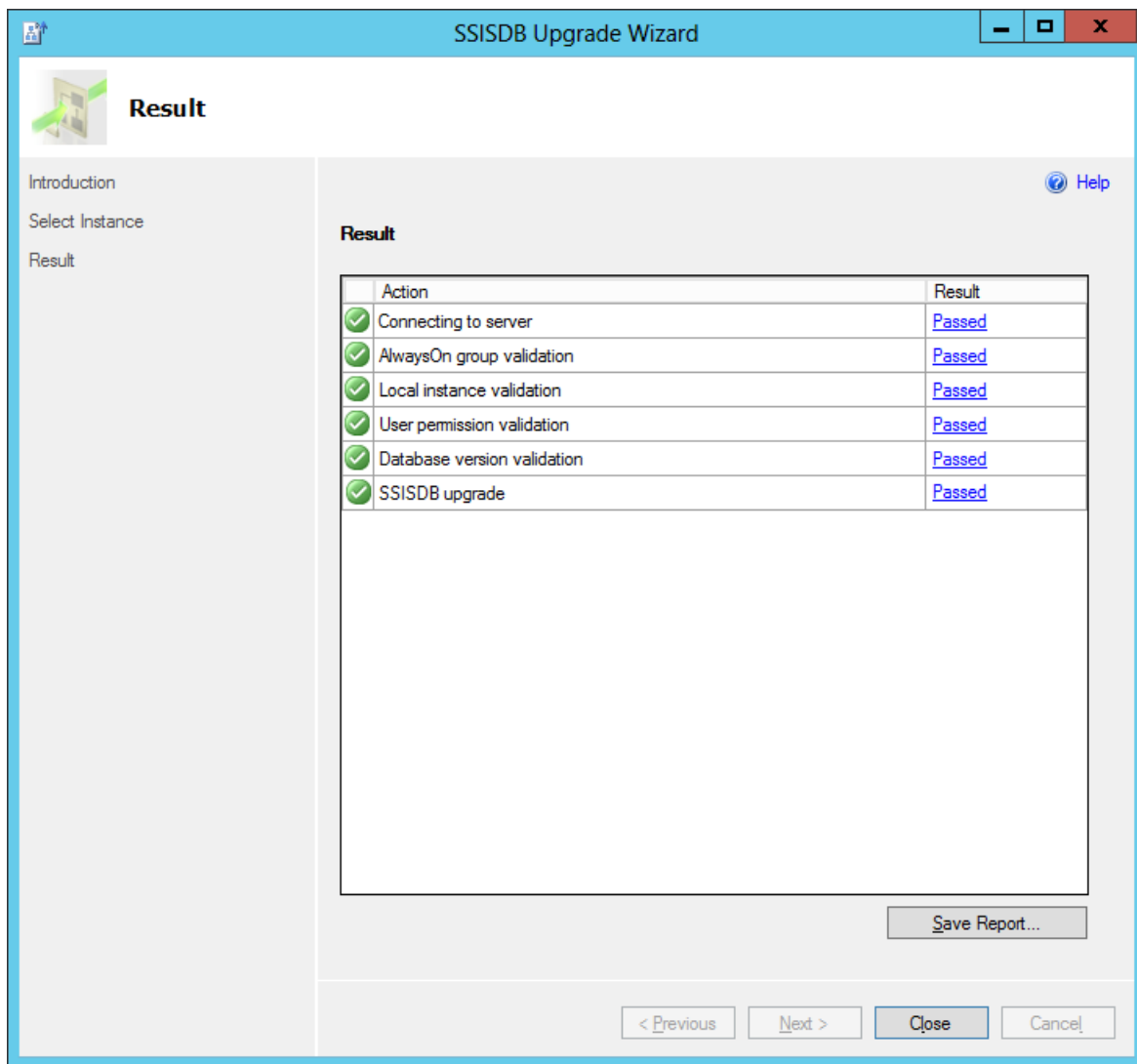
Server name:

Browse...

☐ I have taken a backup of the SSISDB database. Continue with the the SSISDB upgrade.

< Previous   Next >   Upgrade   Cancel

5. Select **Upgrade** to upgrade the SSIS Catalog database.
6. On the **Result** page, review the results.



## Always On for SSIS Catalog (SSISDB)

The Always On Availability Groups feature is a high-availability and disaster-recovery solution that provides an enterprise-level alternative to database mirroring. An availability group supports a failover environment for a discrete set of user databases, known as availability databases, that fail over together. For more information, see [Always On Availability Groups](#).

In order to provide the high-availability for the SSIS catalog (SSISDB) and its contents (projects, packages, execution logs, etc.), you can add the SSISDB database (just the same as any other user database) to an Always On Availability Group. When a failover occurs, one of the secondary nodes automatically becomes the new primary node.

### IMPORTANT

When a failover occurs, packages that were running do not restart or resume.

#### In this section:

1. [Prerequisites](#)
2. [Configure SSIS support for Always On](#)
3. [Upgrading SSISDB in an availability group](#)

#### Prerequisites

You must perform the following pre-requisite steps before enabling Always On support for the SSISDB database.

1. Set up a Windows failover cluster. See [Installing the Failover Cluster Feature and Tools for Windows Server 2012](#) blog post for instructions. You should install the feature and tools on all cluster nodes.
2. Install SQL Server 2016 with Integration Services (SSIS) feature on each node of the cluster.
3. Enable Always On Availability Groups for each SQL Server instance. See [Enable Always On Availability Groups](#) for details.

### Configure SSIS support for Always On

- [Step 1: Create Integration Services Catalog](#)
- [Step 2: Add SSISDB to an Always On Availability Group](#)
- [Step 3: Enable SSIS support for Always On](#)

#### IMPORTANT

- You must perform these steps on the **primary node** of the availability group.
- You must enable **SSIS support for Always On** *after* you add SSISDB to an Always On Availability Group.

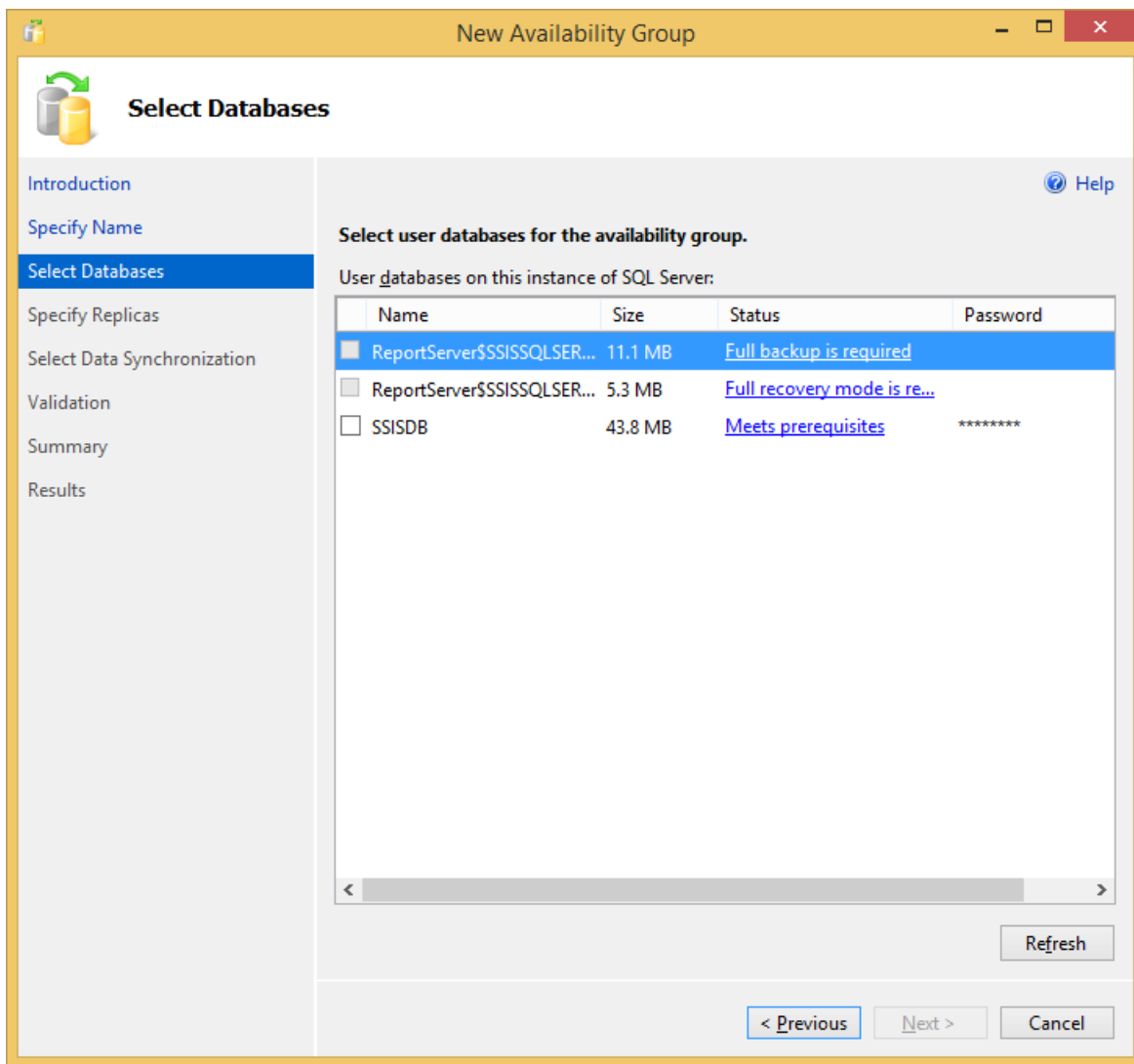
#### Step 1: Create Integration Services Catalog

1. Launch **SQL Server Management Studio** and connect to a SQL Server instance in the cluster that you want to set as the **primary node** of Always On high availability group for SSISDB.
2. In Object Explorer, expand the server node, right-click the **Integration Services Catalogs** node, and then click **Create Catalog**.
3. Click **Enable CLR Integration**. The catalog uses CLR stored procedures.
4. Click **Enable automatic execution of Integration Services stored procedure at SQL Server startup** to enable the [catalog.startup](#) stored procedure to run each time the SSIS server instance is restarted. The stored procedure performs maintenance of the state of operations for the SSISDB catalog. It fixes the status of any packages there were running if and when the SSIS server instance goes down.
5. Enter a **password**, and then click **Ok**. The password protects the database master key that is used for encrypting the catalog data. Save the password in a secure location. It is recommended that you also back up the database master key. For more information, see [Back Up a Database Master Key](#).

#### Step 2: Add SSISDB to an Always On Availability Group

Adding the SSISDB database to an Always On Availability Group is almost same as adding any other user database into an availability group. See [Use the Availability Group Wizard](#).

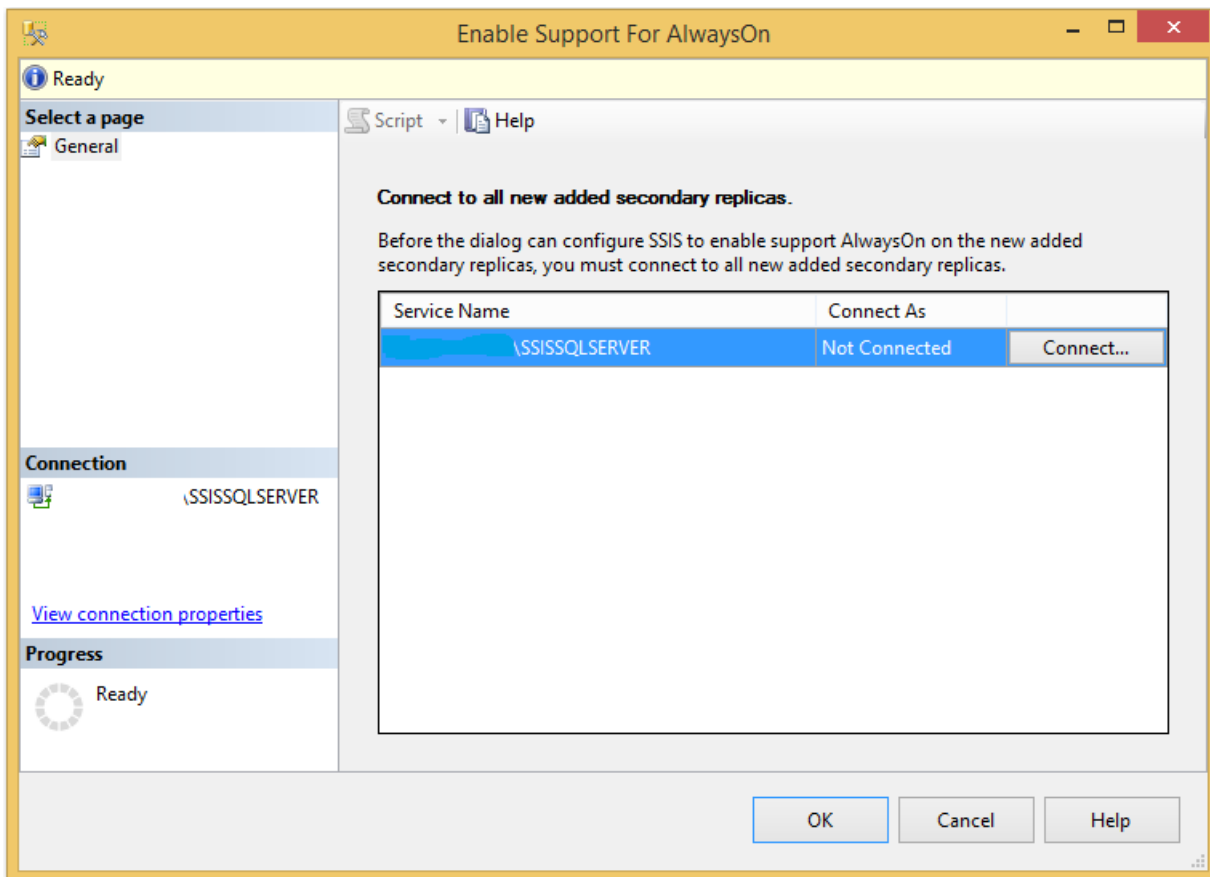
You need to provide the password that you specified while creating the SSIS Catalog in the **Select Databases** page of the **New Availability Group** wizard.



### Step 3: Enable SSIS support for Always On

After you create the Integration Service Catalog, right click the **Integration Service Catalogs** node, and click **Enable Always On Support....** You should see the following **Enable Support for Always On** dialog box. If this menu item is disabled, confirm that you have all the prerequisites installed and click **Refresh**.





#### WARNING

Auto-failover of SSISDB database is not supported until you enable SSIS Support for Always On.

The newly added secondary replicas from the Always On availability group will be shown in the table. Click **Connect...** button for each replica in the list and enter authentication credentials to connect to the replica. The user account must be a member of sysadmin group on each replica to enable SSIS support for Always On. After you successfully connect to each replica, click **OK** to enable SSIS support for Always On.

If the **Enable Always On support** option on the context menu appears to be disabled after you've completed the other prerequisites, try these things:

1. Refresh the context menu by clicking the **Refresh** option.
2. Make sure you are connecting to the primary node. You have to enable Always On support on the primary node.
3. Make sure the SQL Server version is 13.0 or higher. SSIS supports Always On only on SQL Server 2016 and later versions.

#### Upgrading SSISDB in an availability group

If you're upgrading SQL Server from a previous version, and SSISDB is in an Always On availability group, your upgrade may be blocked by the "SSISDB in Always On Availability Group check" rule. This blocking occurs because upgrade runs in single-user mode, while an availability database must be a multi-user database. Therefore, during upgrade or patching, all availability databases including SSISDB are taken offline and are not upgraded or patched. To let upgrade continue, you have to first remove SSISDB from the availability group, then upgrade or patch each node, then add SSISDB back to the availability group.

If you are blocked by the "SSISDB in Always On Availability Group check" rule, you have to follow these steps to upgrade SQL Server.

1. Remove the SSISDB database from the availability group. For more info, see [Remove a Secondary Database](#)

from an Availability Group (SQL Server) and [Remove a Primary Database from an Availability Group \(SQL Server\)](#).

2. Click **Re-run** in the upgrade wizard. The “SSISDB in Always On Availability Group check” rule will pass.
3. Click the **Next** to continue the upgrade.
4. After you have upgraded all the nodes, add the SSISDB database back to the Always On availability group. For more info, see [Add a Database to an Availability Group \(SQL Server\)](#).

If you're not blocked when you upgrade SQL Server, and SSISDB is in an Always On availability group, you have to upgrade SSISDB separately after you upgrade the SQL Server database engine. Use the SSIS Upgrade Wizard to upgrade the SSISDB as described in the following procedure.

5. Move the SSISDB database out of the availability group, or delete the availability group if SSISDB is the only database in the availability group. You have to launch **SQL Server Management Studio** on the **primary node** of the availability group to perform this task.
6. Remove the SSISDB database from all **replica nodes**.
7. Upgrade the SSISDB database on the **primary node**. In **Object Explorer** in SQL Server Management Studio, expand **Integration Services Catalogs**, right-click **SSISDB**, and then select **Database Upgrade**. Follow the instructions in the **SSISDB Upgrade Wizard** to upgrade the database. You have to launch the **SSIDB Upgrade Wizard** locally on the **primary node**.
8. Follow the instructions in [Step 2: Add SSISDB to an Always On Availability Group](#) to add the SSISDB back to an availability group.
9. Follow the instructions in [Step 3: Enable SSIS support for Always On](#).

## Related Content

- Blog entry, [SSIS and PowerShell in SQL Server 2012](#), on blogs.msdn.com.
- Blog entry, [SSIS Catalog Access Control Tips](#), on blogs.msdn.com.
- Blog entry, [A Glimpse of the SSIS Catalog Managed Object Model](#), on blogs.msdn.com.

# Integration Services (SSIS) Catalog Transact-SQL Reference

3/24/2017 • 1 min to read • [Edit Online](#)

[Views \(Integration Services Catalog\)](#)

[Stored Procedures \(Integration Services Catalog\)](#)

[Functions \(Integration Services Catalog\)](#)

# View the List of Packages on the Integration Services Server

3/24/2017 • 1 min to read • [Edit Online](#)

You can view the list of packages that are stored on the Integration Services server in one of two ways.

Transact-SQL access

To view the list of packages that are stored on the server, query the view, [catalog.packages \(SSISDB Database\)](#).

In SQL Server Management Studio

To view packages stored on the server by using Object Explorer in SQL Server Management Studio, follow the procedure below.

## To view packages using SQL Server Management Studio

1. In SQL Server Management Studio, connect to the Integration Services server. That is, connect to the instance of the SQL Server Database Engine that hosts the Integration Services database.
2. In Object Explorer, expand the tree to display the **Integration Services Catalogs** node.
3. Expand the **Integration Services Catalogs** node to display the **SSISDB** node.
4. Expand the **SSISDB** node to display a list of one or more folders. Each folder contains one or more projects in the **Projects** folder, and each project contains one more packages in the **Packages** folder.

# Project Properties Dialog Box

3/24/2017 • 1 min to read • [Edit Online](#)

An Integration Services project is a unit of deployment. Each project can contain packages, parameters, and environment references. A project is a securable object and can define permissions for database principals. When a project is re-deployed, the previous version of the project can be stored in the Integration Services catalog.

Project parameters and package parameters can be used to assign values to properties within packages at the time of execution. Some parameters require values before the package can be executed. Parameter values that reference environment variables require that the project has the corresponding environment reference prior to execution.

## What do you want to do?

- [Open the Project Properties dialog box](#)
- [Set the options on the General page](#)
- [Set the options on the Permissions page](#)

## Open the Project Properties dialog box

1. In SQL Server Management Studio, connect to the Integration Services server.

You're connecting to the instance of the SQL Server Database Engine that hosts the SSISDB database.

2. In Object Explorer, expand the tree to display the **Integration Services Catalogs** node.
3. Expand the **SSISDB** node.
4. Expand the folder that contains the project that you want to set properties for.
5. Right-click the project, and then click **Properties**.

## Set the options on the General page

Use the General page to view project properties.

### Name

Lists the project name.

### Identifier

Lists the project ID.

### Description

Displays the optional description of the project.

### Project Version

Lists the project version.

### Deployment Date

Lists the date and time the project was deployed or redeployed.

## Set the options on the Permissions page

Use the **Permissions** page to view and set explicit permissions for the project.

Browse

Click **Browse** to select users and roles that you want to set permissions for, by using the **Browse All Principals** dialog box.

**Name**

Lists the name of the user or role.

**Type**

Lists the type of user or role.

**Permission**

Lists the permissions.

**Grantor**

Lists the user or role that grants the permission.

**Grant**

When **Grant** is selected, the permission is granted for the selected user or role.

**Deny**

When **Deny** is selected, the permission is denied for the selected user or role.

# Project Versions Dialog Box

3/24/2017 • 1 min to read • [Edit Online](#)

Use the **Project Versions** dialog box to view versions of a project and to restore a previous version.

You can also view previous versions in the [catalog.object\\_versions \(SSISDB Database\)](#) view, and use the [catalog.restore\\_project \(SSISDB Database\)](#) stored procedure to restore previous versions.

## What do you want to do?

- [Open the Project Versions dialog box](#)
- [Restore a Project Version](#)

## Open the Project Versions dialog box

1. In SQL Server Management Studio, connect to the Integration Services server.

That is, connect to the instance of the SQL Server Database Engine that hosts the Integration Services database.

2. In Object Explorer, expand the tree to display the **Integration Services Catalogs** node.
3. Expand the **Integration Services Catalogs** node to display the **SSISDB** node.
4. The **SSISDB** node contains one or more folders that each contain one or more projects. Expand the folder that contains the project you are interested in.
5. Right-click the project, and then click **Versions**.

In the **Project Versions** dialog box, the **Versions** table displays the list of versions of the project that have been deployed on the server, with the date and time the version was deployed, the date and time the version was restored (if it was restored), the version description, and a version identifier. The currently active version is indicated with a check mark in the **Current** column of the table.

## Restore a Project Version

To restore a previous version of a project, select the version in the **Versions** table, and then click **Restore to Selected Version**. The project is restored to the selected version and that version is indicated with a check mark in the **Current** column of the **Versions** table.

# Browse All Principals Dialog Box

3/24/2017 • 1 min to read • [Edit Online](#)

Use the **Browse All Principals** dialog box to select a database principal to change the principal's permissions on the selected project or on all projects contained in a selected folder.

## What do you want to do?

- [Open the Browse All Principals dialog box](#)
- [Configure the Options](#)

## Open the Browse All Principals dialog box

1. In SQL Server Management Studio, connect to the Integration Services server.

You're connecting to the instance of the SQL Server Database Engine that hosts the SSISDB catalog.

2. In Object Explorer, expand the tree to display the **Integration Services Catalogs** node.
3. Expand the **SSISDB** node.
4. To change the principal's permissions on all projects contained in a selected folder, right-click the folder and then click **Properties**.

To change the principal's permissions on a selected project, expand the folder that contains the project, right-click the project, and then click **Properties**.

5. Select the **Permissions** page, and then click **Browse**.

## Configure the Options

This page displays the principals from the catalog view, sys.database\_principals, of the SSISDB database.

Selecting principals adds them to the **Logins or roles** list on the **Permissions** page of the parent dialog box when you click **OK** and close the **Browse All Principals** dialog box. After adding principals to the **Logins or roles** list, you can change their permissions on the selected project.

### Selection column

Select to add the principal to the **Logins or roles** list on the **Permissions** page of the parent dialog box.

### Icon column

An icon that corresponds to the **Type** of the principal.

### Name

The name of the principal.

### Type

The type of the principal. The common types are:

- SQL User
- Windows User
- Database Role



# Package Properties Dialog Box

3/24/2017 • 1 min to read • [Edit Online](#)

Use the **Package Properties** dialog box to view properties for packages that are stored on the Integration Services server.

For more information, see [Integration Services \(SSIS\) Server](#).

## What do you want to do?

- [Open the Package Properties dialog box](#)
- [Configure the Options](#)

## Open the Package Properties dialog box

1. In SQL Server Management Studio, connect to the Integration Services server.

You're connecting to the instance of the SQL Server Database Engine that hosts the SSISDB database.

2. In Object Explorer, expand the tree to display the **Integration Services Catalogs** node.
3. Expand the **SSISDB** node.
4. Expand the folder that contains the package you want to view properties for.
5. Right-click the package, and then select **Properties**.

## Configure the Options

Use the **General** page to view the properties of the selected package.

All properties on the **General** page are read-only.

### Name

Displays the name of the package.

### Identifier

Lists the package ID.

### Entry Point

The value of **True** indicates that the package is started directly. The value of **False** indicates that the package is started by another package using the Execute Package task. The default value is **True**.

You set this property in SQL Server Data Tools (SSDT) for both the parent package and the child packages by right-clicking the package in Solution Explorer and then clicking **Entry-point Package**.

### Description

Displays the optional description of the package.

# Validate Dialog Box

3/24/2017 • 1 min to read • [Edit Online](#)

Use the **Validate** dialog box to check for common problems in Integration Services a project or package.

If there is a problem, a message is displayed at the top of the dialog box. Otherwise, the term Ready displays at the top.

## What do you want to do?

- [Open the Validate dialog box](#)
- [Set the options on the General page](#)

## Open the Validate dialog box

1. In SQL Server Management Studio, connect to the Integration Services server.

You're connecting to the instance of the SQL Server Database Engine that hosts the SSISDB database.

2. In Object Explorer, expand the tree to display the **Integration Services Catalogs** node.
3. Expand the **SSISDB** node.
4. Expand the folder that contains the project or package you want to validate.
5. Right-click the package or package, and then click **Validate**.

## Set the options on the General page

### Environment

Select the environment that you want to use to validate the project or package.

### 32-bit runtime

Select to use a 32-bit runtime to execute a package.

The **Parameters** tab lists the parameter values that you use to validate the project or package. The following are the options on the Parameters tab.

### Container

Lists the object that contains the parameter.

### Parameter

Lists the name of the parameters

### Value

Lists the parameter value.

The **Connection Managers** tab lists the connection manager property values that you use to validate the project or package.

The following are the options on the **Connection Managers** tab.

### Container

Lists the object that contains the connection manager.

### Name

Lists the connection manager name.

**Property name**

Lists the name of the connection manager property.

**Value**

Lists the value assigned to the connection manager property.

# Configure Dialog Box

3/24/2017 • 2 min to read • [Edit Online](#)

Use the **Configure** dialog box to configure parameters, connection managers, and references to environments, for packages and projects.

## What do you want to do?

- [Open the Configure Dialog Box](#)
- [Set the options on the Parameters page](#)
- [Set the options on the References page](#)

## Open the Configure Dialog Box

1. In SQL Server Management Studio, connect to the Integration Services server.

You're connecting to the instance of the SQL Server Database Engine that hosts the SSISDB database.

2. In Object Explorer, expand the tree to display the **Integration Services Catalogs** node.
3. Expand the **SSISDB** node.
4. Expand the folder that contains the package or project that you want to configure.
5. Right-click the package or project, and then click **Configure**.

## Set the options on the Parameters page

Use the **Parameters** page to view parameter names and values, and to modify the values.

Select the scope of the parameters displayed in the **Parameters** and **Connection Managers** tabs, in the **Scope** drop-down list.

The following is a list of the options in the **Parameters** tab.

### Container

Lists the object that contains the parameter.

### Name

Lists the parameter name.

### Value

Lists the parameter value. Click the ellipsis button to change the value in the **Set Parameter Value** dialog box.

The following is a list of the options in the **Connection Managers** tab. You use this tab to change values for connection manager properties. Parameters are automatically generated on the SSIS server for the properties.

### Container

Lists the object that contains the connection manager.

### Name

Lists the connection manager name.

### Property name

Lists the name of the connection manager property.

## **Value**

Lists the value assigned to the connection manager property. Click the ellipsis button to change the value in the **Set Parameter Value** dialog box. You can enter a literal value, map an environment variable that contains the value you want to use, or use the default value from the package.

## Set the options on the References page

Use the **References** page to add and remove references to environments, and to access environment properties.

An environment specifies runtime values for packages contained in the projects you've deployed to Integration Services server.

## **Environment**

Lists the environment.

## **Environment Folder**

Lists the folder that contains the environment.

## **Open**

Click to open the **Environment Properties** dialog box.

## **Add**

Click to add a reference to an environment. In the **Browse Environments** dialog box click an environment and then click **OK**.

You can select an environment contained in any project folder under the **SSISDB** node.

## **Remove**

Click an environment listed in the **References** area, and then click **Remove**.

# Set Parameter Value Dialog Box

3/24/2017 • 1 min to read • [Edit Online](#)

Use the **Set Parameter Value** dialog box to set values for parameters and connection manager properties, for projects and packages.

## What do you want to do?

- [Open the Set Parameter Value dialog box](#)
- [Configure the options](#)

## Open the Set Parameter Value dialog box

1. In SQL Server Management Studio, connect to the Integration Services server.

You're connecting to the instance of the SQL Server Database Engine that hosts the SSISDB database.

2. In Object Explorer, expand the tree to display the **Integration Services Catalogs** node.
3. Expand the **SSISDB** node.
4. Right-click a package or project, click **Configure**, and then click the ellipsis button in the **Parameters** tab or in the **Connection Managers** tab.

## Configure the options

### Parameter

Lists the parameter name.

### Type

Lists the data type of the parameter value.

### Description

Shows an optional description for the parameter.

### Edit value

Select this option to edit the parameter value.

### Use default value from package

Select this option to use the default parameter value stored in the package.

### Use environment variable

Select this option to use a variable value stored in the environment, which is referenced by the project or package. To add an environment reference to a project or package, use the **Configure** dialog box. For more information, see [Configure Dialog Box](#).

# Folder Properties Dialog Box

3/24/2017 • 1 min to read • [Edit Online](#)

A folder contains projects and environments in the **SSISDB** catalog. Each folder defines permissions that apply to the contents of the folder. For more information about Integration Services permissions, see [catalog.grant\\_permission \(SSISDB Database\)](#).

## To Set Folder Description and Permissions

1. Right-click the folder and select **Properties**.
2. On the **General** page, select **Description** under **General** and enter an optional description.
3. On the **Permissions** page, click **Browse...**, select one or more database principals, and click **OK**.
4. Select a name under **Logins or roles** and specify the appropriate permissions under **Permissions**.
5. Click **OK** to accept the changes and close the **Folders Properties** dialog box.

## See Also

[Integration Services \(SSIS\) Server](#)

[catalog.grant\\_permission \(SSISDB Database\)](#)

# Integration Services (SSIS) Scale Out

5/2/2017 • 13 min to read • [Edit Online](#)

Integration Services Scale Out provides high performance package execution by distributing executions to multiple machines. You are able to submit a request for multiple package executions in SQL Server Management Studio. These packages will be executed in parallel, in a scale out mode.

SSIS Scale Out consists of a SSIS Scale Out Master and several SSIS Scale Out Workers. The Scale Out Master is responsible for Scale Out management and receives package execution requests from users. Scale Out Workers pull execution tasks from the Scale Out Master and do the package execution work. For more information, see [Scale Out Master](#), [Scale Out Worker](#).

Integration Services Scale Out can run on one machine, where a Scale Out Master and a Scale Out Worker are set up side-by-side on the machine. Scale Out can also run on multiple machines, where each Scale Out Worker is on a different machine.

- [Walkthrough: Set up Integration Services Scale Out](#)

Scale Out supports running multiple packages in the SSISDB catalog in parallel. For more details, see [Run packages in Scale Out](#).

## Walkthrough: Set Up Integration Services Scale Out

Set up Integration Services Scale Out by completing the following tasks.

### NOTE

If you are installing Scale Out on one computer, install the Scale Out Master and Scale Out Worker features at the same time. When you install the features at the same time, the endpoint is automatically generated to connect to Scale Out Master.

- [Install Scale Out Master](#)
- [Install Scale Out Worker](#)
- [Install Scale Out Worker client certificate](#)
- [Open firewall port](#)
- [Start SQL Server Scale Out Master and Worker service](#)
- [Enable Scale Out Master](#)
- [Enable SQL Server Authentication mode](#)
- [Enable Scale Out Worker](#)

### Install Scale Out Master

To enable the functionality of Scale Out Master, you must install Database Engine Services, Integration Services, and its Scale Out Master feature when you set up SQL Server.

For information on setting up Database Engine Services and Integration Services, see [Install SQL Server Database Engine](#), and [Install Integration Services](#).



## NOTE

During Database Engine installation, select Mixed Mode for Authentication mode on the Database Engine Configuration page.

**To install the Scale Out Master feature, use the SQL Server installation wizard or the command prompt.**

- Steps for the SQL Server installation wizard

1. On the **Feature Selection** page, select **Scale Out Master**, which is listed under Integration Services.

**Feature Selection**

Select the Evaluation features to install.

Product Key  
License Terms  
Global Rules  
Microsoft Update  
Product Updates  
Install Setup Files  
Install Rules  
**Feature Selection**  
Feature Rules  
Instance Configuration  
Server Configuration  
Database Engine Configuration  
Integration Services Scale Out ...  
Feature Configuration Rules  
Ready to Install  
Installation Progress  
Complete

Features:

Shared Features

- ☐ R Server (Standalone)
- ☐ Data Quality Client
- ☐ Client Tools Connectivity
- ☒ Integration Services
  - ☒ **Scale Out Master**
  - ☐ Scale Out Worker
- ☐ Client Tools Backwards Compatibility
- ☐ Client Tools SDK
- ☐ Documentation Components
- ☐ Distributed Replay Controller
- ☐ Distributed Replay Client
- ☐ SQL Client Connectivity SDK

Select All Unselect All

Instance root directory: C:\Program Files\Microsoft SQL Server\

Shared feature directory: C:\Program Files\Microsoft SQL Server\

Shared feature directory (x86): C:\Program Files (x86)\Microsoft SQL Server\

Feature description:

Includes Scale Out Master for Integration Services Scale Out.

Prerequisites for selected features:

Already installed:  
Windows PowerShell 3.0 or higher

To be installed from media:

Disk Space Requirements

Drive C: 1231 MB required, 111577 MB available

< Back Next > Cancel

2. On the **Server Configuration** page, select the account to run **SQL Server Integration Services Scale Out Master service** and select the **Startup Type**.

### Server Configuration

Specify the service accounts and collation configuration.

Product Key  
License Terms  
Global Rules  
Microsoft Update  
Product Updates  
Install Setup Files  
Install Rules  
Feature Selection  
Feature Rules  
Instance Configuration  
**Server Configuration**  
Database Engine Configuration  
Integration Services Scale Out ...  
Feature Configuration Rules  
Ready to Install  
Installation Progress  
Complete

Service Accounts Collation

Microsoft recommends that you use a separate account for each SQL Server service.

Service	Account Name	Password	Startup Type
SQL Server Agent	NT Service\SQLSERVERAGE...		Manual
SQL Server Database Engine	NT Service\MSSQLSERVER		Automatic
SQL Server Integration Services 14.0	NT Service\MSDtsServer140		Automatic
SQL Server Integration Services Scale ...	NT Service\SSISScaleOutMa...		Automatic
SQL Server Browser	NT AUTHORITY\LOCAL SER...		Disabled

☐ Grant Perform Volume Maintenance Task privilege to SQL Server Database Engine Service

This privilege enables instant file initialization by avoiding zeroing of data pages. This may lead to information disclosure by allowing deleted content to be accessed.

[Click here for details](#)

< Back
Next >
Cancel

- On the **Integration Services Scale Out Master Configuration** page, specify the port number that Scale Out Master uses to communicate with Scale Out Worker. The default port number is 8391.

### Integration Services Scale Out Master Configuration

Specify the Integration Services Scale Out Master port and certificate thumbprint.

Product Key  
License Terms  
Global Rules  
Microsoft Update  
Product Updates  
Install Setup Files  
Install Rules  
Feature Selection  
Feature Rules  
Instance Configuration  
Server Configuration  
Database Engine Configuration  
**Integration Services Scale Ou...**  
Feature Configuration Rules  
Ready to Install  
Installation Progress  
Complete

Provide a port which is part of the endpoint (e.g. https://[MachineName]:8391) exposed by scale out master for the communication with scale out workers.

Port Number:

Select a SSL certificate that protects the communication between scale out master and scale out workers. A default self-signed certificate is created if no thumbprint is input.

☒ Create a new SSL certificate  
☐ Use an existing SSL certificate

< Back
Next >
Cancel

- Specify the SSL certificate used to protect the communication between Scale Out Master and Scale Out Worker by doing one of the following.
  - Let the setup process create a default, self-signed SSL certificate by clicking **Create a new SSL certificate**. The default certificate is installed under Trusted Root Certification Authorities, Local Computer.
  - Select an existing SSL Certificate on the local computer by clicking **Use an existing SSL certificate** and then clicking **Browse**. The thumbprint of the certificate appears in the text box. Clicking **Browse** displays certificates that are stored in Trusted Root Certification Authorities, Local

Computer. The certificate you select must be stored here.

**Integration Services Scale Out Master Configuration**

Specify the Integration Services Scale Out Master port and certificate thumbprint.

Product Key  
License Terms  
Global Rules  
Microsoft Update  
Product Updates  
Install Setup Files  
Install Rules  
Feature Selection  
Feature Rules  
Instance Configuration  
Server Configuration  
Database Engine Configuration  
**Integration Services Scale Ou...**  
Feature Configuration Rules  
Ready to Install  
Installation Progress  
Complete

Provide a port which is part of the endpoint (e.g. https://[MachineName]:8391) exposed by scale out master for the communication with scale out workers.

Port Number:

Select a SSL certificate that protects the communication between scale out master and scale out workers. A default self-signed certificate is created if no thumbprint is input.

☐ Create a new SSL certificate  
☒ Use an existing SSL certificate

< Back   Next >   Cancel

5. Finish the SQL Server installation wizard.

- Steps for the command prompt

Follow the instructions in [Install SQL Server from the Command Prompt](#). Set the Scale Out Master related parameters by doing the following.

1. Add IS\_Master to the parameter /FEATURES
2. Configure Scale Out Master by specifying the following parameters and their values:  
/ISMASTERSVCACCOUNT, /ISMASTERSVCPASSWORD, /ISMASTERSVCSTARTUPTYPE,  
/ISMASTERSVCPORT, /ISMASTERSVCTHUMBPRINT(optional).

### Install Scale Out Worker

To enable the functionality of Scale Out Worker, you must install Integration Services and its Scale Out Worker feature in SQL Server setup.

**To install the Scale Out Worker feature, use the SQL Server installation wizard or the command prompt.**

- Steps for the SQL Server installation wizard

1. On the **Feature Selection** page, select **Scale Out Worker**, which is listed under Integration Services.

## Feature Selection

Select the Evaluation features to install.

Product Key  
License Terms  
Global Rules  
Microsoft Update  
Product Updates  
Install Setup Files  
Install Rules  
**Feature Selection**  
Feature Rules  
Server Configuration  
Integration Services Scale Out ...  
Feature Configuration Rules  
Ready to Install  
Installation Progress  
Complete

Features:

Shared Features

- ☐ R Server (Standalone)
- ☐ Data Quality Client
- ☐ Client Tools Connectivity
- ☒ Integration Services
  - ☐ Scale Out Master
  - ☒ **Scale Out Worker**
- ☐ Client Tools Backwards Compatibility
- ☐ Client Tools SDK
- ☐ Documentation Components
- ☐ Distributed Replay Controller
- ☐ Distributed Replay Client
- ☐ SQL Client Connectivity SDK

Select All Unselect All

Instance root directory: C:\Program Files\Microsoft SQL Server\
Shared feature directory: C:\Program Files\Microsoft SQL Server\
Shared feature directory (x86): C:\Program Files (x86)\Microsoft SQL Server\

Feature description:

Includes Scale Out Worker for Integration Services Scale Out.

Prerequisites for selected features:

Already installed:

- Microsoft .NET Framework 4.5

To be installed from media:

Disk Space Requirements

Drive C: 307 MB required, 111567 MB available

< Back Next > Cancel

- On the **Server Configuration** page, select the account to run **SQL Server Integration Services Scale Out Worker service** and select the **Startup Type**.

## Server Configuration

Specify the service accounts and collation configuration.

Product Key  
License Terms  
Global Rules  
Microsoft Update  
Product Updates  
Install Setup Files  
Install Rules  
Feature Selection  
Feature Rules  
**Server Configuration**  
Integration Services Scale Out ...  
Feature Configuration Rules  
Ready to Install  
Installation Progress  
Complete

Service Accounts

Microsoft recommends that you use a separate account for each SQL Server service.

Service	Account Name	Passw...	Startup Type
SQL Server Integration Services 14.0	NT Service\MsDtsServer140		Automatic
<b>SQL Server Integration Services Scale ...</b>	NT Service\SSISScaleOutWor...		Automatic

< Back Next > Cancel

- On the **Integration Services Scale Out Worker Configuration** page, specify the endpoint to connect to Scale Out Master.
  - For a **one computer** environment, the endpoint is automatically generated when Scale Out Master and Scale Out Worker are installed at the same time.
  - For a **multiple computers** environment, the endpoint consists of the name or IP of the computer with Scale Out Master installed and the port number specified during the Scale Out Master installation.

### Integration Services Scale Out Worker Configuration

Specify the Integraion Services Scale Out Worker master endpoint and certificate.

<ul style="list-style-type: none"> <li>Product Key</li> <li>License Terms</li> <li>Global Rules</li> <li>Microsoft Update</li> <li>Product Updates</li> <li>Install Setup Files</li> <li>Install Rules</li> <li>Feature Selection</li> <li>Feature Rules</li> <li>Server Configuration</li> <li><b>Integration Services Scale Ou...</b></li> <li>Feature Configuration Rules</li> <li>Ready to Install</li> <li>Installation Progress</li> <li>Complete</li> </ul>	<p>Provide the endpoint to connect to scale out master (e.g. https://[MachineName]:[Port]):</p> <input type="text" value="https://MasterMachine:8391"/> <p>Select the client SSL certificate used to validate wether the SSL server certificate of scale out master is authentic. This is only required when the server certificate is self-signed and scale out worker is not on the same machine with scale out master.</p> <div> <input type="text"/> <input type="button" value="Browse..."/> </div>
--	--

- For a **multiple computers** environment, specify the client SSL certificate that is used to validate Scale Out Master. For a **one computer** environment, there's no need to specify the client SSL certificate.

#### NOTE

When the SSL certificate used by Scale Out Master is self-signed, a corresponding client SSL certificate is required to be installed on the computer with Scale Out Worker. If you provide the file path for the client SSL Certificate on the **Integration Services Scale Out Worker Configuration** page, the certificate will be installed automatically; otherwise, you have to install the certificate manually later.

Click **Browse** to find the certificate file (\*.cer). To use the default SSL certificate, select the SSISScaleOutMaster.cer file located under <drive>:\Program Files\Microsoft SQL Server\140\DTS\Binn on the computer on which Scale Out Master is installed.

5. Finish the SQL Server installation wizard.

- Steps for the command prompt

Follow the instructions in [Install SQL Server from the Command Prompt](#). Set the Scale Out Worker related parameters by doing the following.

1. Add IS\_Worker to the parameter /FEATURES
2. Configure Scale Out Worker specifying the following parameters and their values:  
/ISWORKERSVCAccount, /ISWORKERSVCPASSWORD, /ISWORKERSVCSTARTUPTYPE,  
/ISWORKERSVCMaster(optional), /ISWORKERSVCCERT(optional).

### Install Scale Out Worker client certificate

During the installation of Scale Out Worker, a worker certificate will be automatically created and installed on the computer. Also, a corresponding client certificate, SSISScaleOutWorker.cer, is installed under <drive>:\Program Files\Microsoft SQL Server\140\DTS\Binn. For Scale Out Master to authenticate the Scale Out Worker, you must add this client certificate to the Root store of the local computer with Scale Out Master.

To add the client certificate to the Root store, double click the .cer file and then click **Install Certificate** in the Certificate dialog box. The **Certificate Import Wizard** displays.

## Open firewall port

Open the port specified during the Scale Out Master installation and the port of SQL Server (1433, by default), using Windows Firewall on the Scale Out Master computer.

## Start SQL Server Scale Out Master and Worker services

If the startup type of the services is not set to Automatic during installation, start the services: SQL Server Integration Services Scale Out Master 14.0 (SSISScaleOutMaster140) and SQL Server Integration Services Scale Out Worker 14.0 (SSISScaleOutWorker140).

#### NOTE

After you open the firewall port, you also need to restart the Scale Out Worker service.

### Enable Scale Out Master

Click **Enable this server as SSIS scale out master** in the **Create Catalog** dialog when you create the SSISDB catalog in SQL Server Management Studio.

## Enable SQL Server Authentication mode

If SQL Server authentication is not enabled during the Database Engine installation, enable SQL Server authentication mode on the SQL Server instance that hosts the SSISDB catalog.

Package execution is not blocked when SQL Server authentication is disabled. However, the execution log will not be able to write to SSISDB.

### Enable Scale Out Worker

To enable a Scale Out Worker, execute the `[catalog].[enable_worker_agent]` stored procedure with **WorkerAgentId** as the parameter.

You get the **WorkerAgentId** value from the `[catalog].[worker_agents]` database view in SSISDB, after Scale Out Worker registers with Scale Out Master. Registration takes several minutes once the the Scale Out Master and Worker services are started.

#### Example

This example enables the Scale Out Worker on computerA.

```
SELECT WorkerAgentId, computerName FROM [catalog].[worker_agents]
GO
-- Result: --
-- WorkerAgentId          computerName --
-- 6583054A-E915-4C2A-80E4-C765E79EF61D  computerA   --

EXEC [catalog].[enable_worker_agent] '6583054A-E915-4C2A-80E4-C765E79EF61D'
GO
```

### Next Steps

The set up of the Scale Out feature is finished. You can now run packages in Scale Out. For more information, see [Execute Packages in Integration Services \(SSIS\) Scale Out](#).

## Integration Services (SSIS) Scale Out Master

Scale Out Master manages the Scale Out system through the SSISDB Catalog and the Scale Out Master service.

The SSISDB Catalog stores all the information for Scale Out Workers, packages and executions. It provides the interface to enable a Scale Out Worker and execute packages in Scale Out. For more information, see [Walkthrough: Set up Integration Services Scale Out](#), [Run Packages in Integration Services](#).

Scale Out Master service is a Windows service that is responsible for the communication with Scale Out Workers. It exchanges the status of package executions with Scale Out Workers through HTTPS and operates on the data in SSISDB.

### Scale Out related SQL views and stored procedures in SSISDB

#### Views:

[\[catalog\].\[master\\_properties\]](#), [\[catalog\].\[worker\\_agents\]](#).

#### Stored procedures:

- For Scale Out Worker management:  
[\[catalog\].\[disable\\_worker\\_agent\]](#), [\[catalog\].\[enable\\_worker\\_agent\]](#).
- For executing packages in Scale Out:  
[\[catalog\].\[create\\_execution\]](#), [\[catalog\].\[add\\_execution\\_worker\]](#), [\[catalog\].\[start\\_execution\]](#).

#### Configure SQL Server Integration Services Scale Out Master service

Scale Out Master service can be configured using the <driver>:\Program Files\Microsoft SQL Server\140\DTS\Binn\MasterSettings.config file. The service must be restarted after updating the configuration file.

CONFIGURATION	DESCRIPTION	DEFAULT VALUE
PortNumber	The network port number used to communicate with a Scale Out Worker.	8391
SSLCertThumbprint	The thumbprint of the SSL certificate used to protect the communication with a Scale Out Worker.	The thumbprint of the SSL certificate specified during the Scale Out Master installation
InstanceName	The name of the SQL Server instance that contains the SSISDB catalog. MSSQLSERVER is the name of the default SQL Server instance.	The name of the SQL Server instance that is installed with the Scale Out Master
CleanupCompletedJobsIntervalInMs	The interval for cleaning up completed execution jobs, in milliseconds.	43200000
DealWithExpiredTasksIntervalInMs	The interval for dealing with expired execution jobs, in milliseconds.	300000
MasterHeartbeatIntervalInMs	The interval for the Scale Out Master heartbeat, in milliseconds. This specifies the interval that Scale Out Master updates its online status in the SSISDB catalog.	30000

#### View Scale Out Master service log

The Scale Out Master service log file is located in the <driver>:\Users\[account]\AppData\Local\SSIS\Cluster\Master folder path.

The [account] folder refers to the account running Scale Out Master service. By default, this account is SSISScaleOutMaster140.

## Integration Services (SSIS) Scale Out Worker

Scale Out Worker runs a SQL Server Integration Services Scale Out Worker service to pull execution tasks from Scale Out Master and, executes the packages locally with ISServerExec.exe.

#### Configure SQL Server Integration Services Scale Out Worker service

Scale Out Worker service can be configured using the <driver>:\Program Files\Microsoft SQL Server\140\DTS\Binn\WorkerSettings.config file. The service must be restarted after updating the configuration file.



CONFIGURATION	DESCRIPTION	DEFAULT VALUE
DisplayName	The display name of the Scale Out Worker. <b>NOT in use in SQL Server 2017 CTP1.</b>	Machine name
Description	The description of the Scale Out Worker. <b>NOT in use in SQL Server 2017 CTP1.</b>	Empty
MasterEndpoint	The endpoint to connect to Scale Out Master.	The endpoint set during the Scale Out Worker installation
MasterHttpsCertThumbprint	The thumbprint of the client SSL certificate used to authenticate Scale Out Master	The thumbprint of the client certificate specified during the Scale Out Worker installation.
WorkerHttpsCertThumbprint	The thumbprint of the certificate for Scale Out Master used to authenticate the Scale Out Worker.	The thumbprint of a certificate created and installed automatically during the Scale Out Worker installation
StoreLocation	The store location of worker certificate.	LocalMachine
StoreName	The store name that worker certificate is in.	My
AgentHeartbeatInterval	The interval of the Scale Out Worker heartbeat.	00:01:00
TaskHeartbeatInterval	The interval of the Scale Out Worker reporting task state.	00:00:10
HeartbeatErrorTolerance	After this time period from last successful task heartbeat, the task is terminated if error response of heartbeat is received.	00:10:00
TaskRequestMaxCPU	The upper limit of CPU for Scale Out Worker to request tasks. <b>NOT in use in SQL Server 2017 CTP1.</b>	70.0
TaskRequestMinMemory	The lower limit of memory in MB for Scale Out Worker to request tasks. <b>NOT in use in SQL Server 2017 CTP1.</b>	100.0
MaxTaskCount	The max number of tasks the Scale Out Worker can hold.	10
LeaseInterval	The lease interval of a task holding by the Scale Out Worker.	00:01:00

CONFIGURATION	DESCRIPTION	DEFAULT VALUE
TasksRootFolder	The folder of task logs. The <driver>:\Users\[account]\AppData\Local\SSIS\Cluster\Tasks folder path is used if the value is empty. [account] is the account running Scale Out Worker service. By default, the account is SSISScaleOutWorker140.	Empty
TaskLogLevel	The task log level of the Scale Out Worker. (Verbose 0x01, Information 0x02, Warning 0x04, Error 0x08, Progress 0x10, CriticalError 0x20, Audit 0x40)	126 (Information,Warning,Error,Progress,CriticalError,Audit)
TaskLogSegment	The time span of a task log file.	00:00:00
TaskLogEnabled	Specifies whether the task log is enabled.	true
ExecutionLogCacheFolder	The folder used to cache package execution log. The <driver>:\Users\[account]\AppData\Local\SSIS\Cluster\Agent\ELogCache folder path is used if the value is empty. [account] is the account running Scale Out Worker service. By default, the account is SSISScaleOutWorker140.	Empty
ExecutionLogMaxBufferLogCount	The max number of execution logs cached, in one execution log buffer in memory.	10000
ExecutionLogMaxInMemoryBufferCount	The max number of execution log buffers in memory for execution logs.	10
ExecutionLogRetryCount	The retry count if execution logging fails.	3
AgentId	Worker agent Id of the Scale Out Worker	Generated automatically

### View Scale Out Worker log

The log file of Scale Out Worker service is in the <driver>:\Users\[account]\AppData\Local\SSIS\Cluster\Agent folder path.

The log location of each individual task is configured in the WorkerSettings.config file by TasksRootFolder. If it is not specified, the log is in the <driver>:\Users\[account]\AppData\Local\SSIS\Cluster\Tasks folder path.

The [account] folder is the account running Scale Out Worker service. By default, the account is SSISScaleOutWorker140.

## Run Packages in Integration Services (SSIS) Scale Out

After the packages are deployed to the Integration Services server, you can execute them in Scale Out.

### Run packages with Execute Package In Scale Out dialog

1. **### Open the Execute Package In Scale Out dialog box ###** In SQL Server Management Studio, connect to the Integration Services server. In Object Explorer, expand the tree to display the nodes under **Integration Services Catalogs**. Right-click the **SSISDB** node or the project or the package you want to run, and then click **Execute in Scale Out**.

## 2. Select packages and set the options

On the **Package Selection** page, you select multiple packages to run and set the environment, parameters, connection managers, and advanced options for each package. Click a package to set these options.

In the **Advanced** tab, you set a Scale Out option called **Retry count**. It sets the number of times a package execution will retry if it fails.

## 3. Select machines

On the **Machine Selection** page, you select the Scale Out Worker machines to run the packages. By default, any machine is allowed to run the packages.

### NOTE

The packages are executed with the credential of the user accounts of the Scale Out Worker services, which are shown on the **Machine Selection** page. By default, the account is NT Service\SSISScaleOutWorker140. You may want to change to your own lab accounts.

## 4. Run the packages and view reports

Click **OK** to start the package executions. To view the execution report for a package, right-click the package in Object Explorer, click **Reports**, click **All Executions**, and find the execution.

### Run packages with stored procedures

1. **### Create executions ###** Call [catalog].[create\_execution] for each package. Set parameter **@runincluster** to True. If not all Scale Out Worker machines are allowed to run the package, set parameter **@useanyworker** to False.
2. **### Set execution parameters ###** Call [catalog].[set\_execution\_parameter\_value] for each execution.
3. **### Set Scale Out Workers ###** Call [catalog].[add\_execution\_worker]. If any machine is allowed to run the package, you do not need to call this stored procedure.
4. **### Start executions ###** Call [catalog].[start\_execution]. Set parameter **@retry\_count** to set the number of times a package execution will retry if it fails.

### Example

The following example runs two packages, package1.dtsx and package2.dtsx, in Scale Out with one Scale Out Worker.

```

Declare @execution_id bigint
EXEC [SSISDB].[catalog].[create_execution] @package_name=N'package1.dtsx', @execution_id=@execution_id OUTPUT,
@folder_name=N'folder1', @project_name=N'project1', @use32bitruntime=False, @reference_id=NULL,
@useanyworker=False, @runincluster=True
Select @execution_id
DECLARE @var0 smallint = 1
EXEC [SSISDB].[catalog].[set_execution_parameter_value] @execution_id, @object_type=50,
@parameter_name=N'LOGGING_LEVEL', @parameter_value=@var0
EXEC [SSISDB].[catalog].[add_execution_worker] @execution_id, @workeragent_id=N'64c020e2-f819-4c2d-a22f-
efb31a91e70a'
EXEC [SSISDB].[catalog].[start_execution] @execution_id, @retry_count=0
GO

Declare @execution_id bigint
EXEC [SSISDB].[catalog].[create_execution] @package_name=N'package2.dtsx', @execution_id=@execution_id OUTPUT,
@folder_name=N'folder2', @project_name=N'project2', @use32bitruntime=False, @reference_id=NULL,
@useanyworker=False, @runincluster=True
Select @execution_id
DECLARE @var0 smallint = 1
EXEC [SSISDB].[catalog].[set_execution_parameter_value] @execution_id, @object_type=50,
@parameter_name=N'LOGGING_LEVEL', @parameter_value=@var0
EXEC [SSISDB].[catalog].[add_execution_worker] @execution_id, @workeragent_id=N'64c020e2-f819-4c2d-a22f-
efb31a91e70a'
EXEC [SSISDB].[catalog].[start_execution] @execution_id, @retry_count=0
GO

```

## Permissions

Running packages in Scale Out requires one the following permissions:

- Membership in the **ssis\_admin** database role
- Membership in the **ssis\_cluster\_executor** database role
- Membership in the **sysadmin** server role

# Integration Services Service (SSIS Service)

4/14/2017 • 20 min to read • [Edit Online](#)

The topics in this section discuss the Integration Services service, a Windows service for managing Integration Services packages. This service is not required to create, save, and run Integration Services packages. SQL Server 2012 supports the Integration Services service for backward compatibility with earlier releases of Integration Services.

Starting in SQL Server 2012, Integration Services stores objects, settings, and operational data in the **SSISDB** database for projects that you've deployed to the Integration Services server using the project deployment model. The Integration Services server, which is an instance of the SQL Server Database Engine, hosts the database. For more information about the database, see [SSIS Catalog](#). For more information about deploying projects to the Integration Services server, see [Deploy Integration Services \(SSIS\) Projects and Packages](#).

## Management capabilities

The Integration Services service is a Windows service for managing Integration Services packages. The Integration Services service is available only in SQL Server Management Studio.

Running the Integration Services service provides the following management capabilities:

- Starting remote and locally stored packages
- Stopping remote and locally running packages
- Monitoring remote and locally running packages
- Importing and exporting packages
- Managing package storage
- Customizing storage folders
- Stopping running packages when the service is stopped
- Viewing the Windows Event log
- Connecting to multiple Integration Services servers

## Startup type

The Integration Services service is installed when you install the Integration Services component of SQL Server. By default, the Integration Services service is started and the startup type of the service is set to automatic. The service must be running to monitor the packages that are stored in the SSIS Package Store. The SSIS Package Store can be either the msdb database in an instance of SQL Server or the designated folders in the file system.

The Integration Services service is not required if you only want to design and execute Integration Services packages. However, the service is required to list and monitor packages using SQL Server Management Studio.

## Manage the service

When you install the Integration Services component of SQL Server, the Integration Services service is also installed. By default, the Integration Services service is started and the startup type of the service is set to automatic. However, you must also install SQL Server Management Studio to use the service to manage stored

and running Integration Services packages.

**NOTE:** To connect directly to an instance of the legacy Integration Services Service, you have to use the version of SQL Server Management Studio (SSMS) aligned with the version of SQL Server on which the Integration Services Service is running. For example, to connect to the legacy Integration Services Service running on an instance of SQL Server 2016, you have to use the version of SSMS released for SQL Server 2016. [Download SQL Server Management Studio \(SSMS\)](#).

In the SSMS **Connect to Server** dialog box, you cannot enter the name of a server on which an earlier version of the Integration Services service is running. However, to manage packages that are stored on a remote server, you do not have to connect to the instance of the Integration Services service on that remote server. Instead, edit the configuration file for the Integration Services service so that SQL Server Management Studio displays the packages that are stored on the remote server.

You can only install a single instance of the Integration Services service on a computer. The service is not specific to a particular instance of the Database Engine. You connect to the service by using the name of the computer on which it is running.

You can manage the Integration Services service by using one of the following Microsoft Management Console (MMC) snap-ins: SQL Server Configuration Manager or Services. Before you can manage packages in SQL Server Management Studio, you must make sure that the service is started.

By default, the Integration Services service is configured to manage packages in the msdb database of the instance of the Database Engine that is installed at the same time as Integration Services. If an instance of the Database Engine is not installed at the same time, the Integration Services service is configured to manage packages in the msdb database of the local, default instance of the Database Engine. To manage packages that are stored in a named or remote instance of the Database Engine, or in multiple instances of the Database Engine, you have to modify the configuration file for the service.

By default, the Integration Services service is configured to stop running packages when the service is stopped. However, the Integration Services service does not wait for packages to stop and some packages may continue running after the Integration Services service is stopped.

If the Integration Services service is stopped, you can continue to run packages using the SQL Server Import and Export Wizard, the SSIS Designer, the Execute Package Utility, and the **dtexec** command prompt utility (dtexec.exe). However, you cannot monitor the running packages.

By default, the Integration Services service runs in the context of the NETWORK SERVICE account.

The Integration Services service writes to the Windows event log. You can view service events in SQL Server Management Studio. You can also view service events by using the Windows Event Viewer.

## Set the properties of the service

The Integration Services service manages and monitors packages in SQL Server Management Studio. When you first install SQL Server Integration Services, the Integration Services service is started and the startup type of the service is set to automatic.

After the Integration Services service has been installed, you can set the properties of the service by using either SQL Server Configuration Manager or the Services MMC snap-in.

To configure other important features of the service, including the locations where it stores and manages packages, you must modify the configuration file of the service.

### To set properties of the Integration Services service by using SQL Server Configuration Manager

1. On the **Start** menu, point to **All Programs**, point to **Microsoft SQL Server**, point to **Configuration Tools**,

and then click **SQL Server Configuration Manager**.

2. In the **SQL Server Configuration Manager** snap-in, locate **SQL Server Integration Services** in the list of services, right-click **SQL Server Integration Services**, and then click **Properties**.
3. In the **SQL Server Integration Services Properties** dialog box you can do the following:
  - Click the **Log On** tab to view the logon information such as the account name.
  - Click the **Service** tab to view information about the service such as the name of the host computer and to specify the start mode of Integration Services service.

**NOTE**

The **Advanced** tab contains no information for Integration Services service.

4. Click **OK**.
5. On the **File** menu, click **Exit** to close the **SQL Server Configuration Manager** snap-in.

**To set properties of the Integration Services service by using Services**

1. In **Control Panel**, if you are using Classic View, click **Administrative Tools**, or, if you are using Category View, click **Performance and Maintenance** and then click **Administrative Tools**.
2. Click **Services**.
3. In the **Services** snap-in, locate **SQL Server Integration Services** in the list of services, right-click **SQL Server Integration Services**, and then click **Properties**.
4. In the **SQL Server Integration Services Properties** dialog box, you can do the following:
  - Click the **General** tab. To enable the service, select either the manual or automatic startup type. To disable the service, select **Disable** in the **Startup type** box. Selecting **Disable** does not stop the service if it is currently running.  
  
If the service is already enabled, you can click **Stop** to stop the service, or click **Start** to start the service.
  - Click the **Log On** tab to view or edit the logon information.
  - Click the **Recovery** tab to view the default computer responses to service failure. You can modify these options to suit your environment.
  - Click the **Dependencies** tab to view a list of dependent services. The Integration Services service has no dependencies.
5. Click **OK**.
6. Optionally, if the startup type is Manual or Automatic, you can right-click **SQL Server Integration Services** and click **Start, Stop, or Restart**.
7. On the **File** menu, click **Exit** to close the **Services** snap-in.

## Grant permissions to the service

In previous versions of SQL Server, by default when you installed SQL Server all users in the Users group had access to the Integration Services service. When you install the current release of SQL Server, users do not have access to the Integration Services service. The service is secure by default. After SQL Server is installed, the administrator must grant access to the service.

### To grant access to the Integration Services service

1. Run Dcomcnfg.exe. Dcomcnfg.exe provides a user interface for modifying certain settings in the registry.
2. In the **Component Services** dialog, expand the Component Services > Computers > My Computer > DCOM Config node.
3. Right-click **Microsoft SQL Server Integration Services 13.0**, and then click **Properties**.
4. On the **Security** tab, click **Edit** in the **Launch and Activation Permissions** area.
5. Add users and assign appropriate permissions, and then click Ok.
6. Repeat steps 4 - 5 for Access Permissions.
7. Restart SQL Server Management Studio.
8. Restart the Integration Services Service.

## Configure the service

When you install Integration Services, the setup process creates and installs the configuration file for the Integration Services service. This configuration file contains the following settings:

- Packages are sent a stop command when the service stops.
- The root folders to display for Integration Services in Object Explorer of SQL Server Management Studio are the MSDB and File System folders.
- The packages in the file system that the Integration Services service manages are located in %ProgramFiles%\Microsoft SQL Server\130\DTS\Packages.

This configuration file also specifies which msdb database contains the packages that the Integration Services service will manage. By default, the Integration Services service is configured to manage packages in the msdb database of the instance of the Database Engine that is installed at the same time as Integration Services. If an instance of the Database Engine is not installed at the same time, the Integration Services service is configured to manage packages in the msdb database of the local, default instance of the Database Engine.

### Default Configuration File Example

The following example shows a default configuration file that specifies the following settings:

- Packages stop running when the Integration Services service stops.
- The root folders for package storage in Integration Services are MSDB and File System.
- The service manages packages that are stored in the msdb database of the local, default instance of SQL Server.
- The service manages packages that are stored in the file system in the Packages folder.

### Example of a Default Configuration File



```
<?xml version="1.0" encoding="utf-8"?>
<DtsServiceConfiguration xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <StopExecutingPackagesOnShutdown>true</StopExecutingPackagesOnShutdown>
  <TopLevelFolders>
    \<Folder xsi:type="SqlServerFolder">
      <Name>MSDB</Name>
      <ServerName>.</ServerName>
    </Folder>
    \<Folder xsi:type="FileSystemFolder">
      <Name>File System</Name>
      <StorePath>..\Packages</StorePath>
    </Folder>
  </TopLevelFolders>
</DtsServiceConfiguration>
```

## Modify the configuration file

You can modify the configuration file to allow packages to continue running if the service stops, to display additional root folders in Object Explorer, or to specify a different folder or additional folders in the file system to be managed by Integration Services service. For example, you can create additional root folders of type, **SqlServerFolder**, to manage packages in the msdb databases of additional instances of Database Engine.

### NOTE

Some characters are not valid in folder names. Valid characters for folder names are determined by the .NET Framework class **System.IO.Path** and the **GetInvalidFilenameChars** field. The **GetInvalidFilenameChars** field provides a platform-specific array of characters that cannot be specified in path string arguments passed to members of the **Path** class. The set of invalid characters can vary by file system. Typically, invalid characters are the quotation mark ("), less than (<) character, and pipe (|) character.

However, you will have to modify the configuration file to manage packages that are stored in a named instance or a remote instance of Database Engine. If you do not update the configuration file, you cannot use **Object Explorer** in SQL Server Management Studio to view packages that are stored in the msdb database on the named instance or the remote instance. If you try to use **Object Explorer** to view these packages, you receive the following error message:

Failed to retrieve data for this request. (Microsoft.SqlServer.SmoEnum)

The SQL Server specified in Integration Services service configuration is not present or is not available. This might occur when there is no default instance of SQL Server on the computer. For more information, see the topic "Configuring the Integration Services Service" in SQL Server 2008 Books Online.

Login Timeout Expired

An error has occurred while establishing a connection to the server. When connecting to SQL Server 2008, this failure may be caused by the fact that under the default settings SQL Server does not allow remote connections.

Named Pipes Provider: Could not open a connection to SQL Server [2]. (MsDtsSvr).

To modify the configuration file for the Integration Services service, you use a text editor.

### IMPORTANT

After you modify the service configuration file, you must restart the service to use the updated service configuration.

## Modified Configuration File Example

The following example shows a modified configuration file for Integration Services. This file is for a named instance of SQL Server called `InstanceName` on a server named `ServerName`.

## Example of a Modified Configuration File for a Named Instance of SQL Server

```
<?xml version="1.0" encoding="utf-8"?>
<DtsServiceConfiguration xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <StopExecutingPackagesOnShutdown>true</StopExecutingPackagesOnShutdown>
  <TopLevelFolders>
    \<Folder xsi:type="SqlServerFolder">
      <Name>MSDB</Name>
      <ServerName>ServerName\InstanceName</ServerName>
    </Folder>
    \<Folder xsi:type="FileSystemFolder">
      <Name>File System</Name>
      <StorePath>..\Packages</StorePath>
    </Folder>
  </TopLevelFolders>
</DtsServiceConfiguration>
```

### Modify the Configuration File Location

The Registry key **HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Microsoft SQL**

**Server\130\SSIS\ServiceConfigFile** specifies the location and name for the configuration file that Integration Services service uses. The default value of the Registry key is **C:\Program Files\Microsoft SQL**

**Server\130\DTs\Binn\MsDtsSrvr.ini.xml**. You can update the value of the Registry key to use a different name and location for the configuration file. Note that the version number in the path (120 for SQL Server SQL Server 2014, 130 for SQL Server 2016, etc.) will vary depending on the SQL Server version.

#### Caution

Incorrectly editing the Registry can cause serious problems that may require you to reinstall your operating system. Microsoft cannot guarantee that problems resulting from editing the Registry incorrectly can be resolved. Before editing the Registry, back up any valuable data. For information about how to back up, restore, and edit the Registry, see the Microsoft Knowledge Base article, [Description of the Microsoft Windows registry](#).

The Integration Services service loads the configuration file when the service is started. Any changes to the Registry entry require that the service be restarted.

## Connect to the local service

Before you connect to the Integration Services service, the administrator must grant you access to the service.

### To connect to the Integration Services Service

1. Open SQL Server Management Studio.
2. Click **Object Explorer** on the **View** menu.
3. On the Object Explorer toolbar, click **Connect**, and then click **Integration Services**.
4. In the **Connect to Server** dialog box, provide a server name. You can use a period (.), (local), or **localhost** to indicate the local server.
5. Click **Connect**.

## Connect to a remote SSIS server

Connecting to an instance of Integration Services on a remote server, from SQL Server Management Studio or another management application, requires a specific set of rights on the server for the users of the application.

## IMPORTANT

To connect directly to an instance of the legacy Integration Services Service, you have to use the version of SQL Server Management Studio (SSMS) aligned with the version of SQL Server on which the Integration Services Service is running. For example, to connect to the legacy Integration Services Service running on an instance of SQL Server 2016, you have to use the version of SSMS released for SQL Server 2016. [Download SQL Server Management Studio \(SSMS\)](#).

To manage packages that are stored on a remote server, you do not have to connect to the instance of the Integration Services service on that remote server. Instead, edit the configuration file for the Integration Services service so that SQL Server Management Studio displays the packages that are stored on the remote server.

## Connecting to Integration Services on a Remote Server

### To connect to Integration Services on a Remote Server

1. Open SQL Server Management Studio.
2. Select **File, Connect Object Explorer** to display the **Connect to Server** dialog box.
3. Select **Integration Services** in the **Server type** list.
4. Type the name of a SQL Server Integration Services server in the **Server name** text box.

## NOTE

The Integration Services service is not instance-specific. You connect to the service by using the name of the computer on which the Integration Services service is running.

5. Click **Connect**.

## NOTE

The **Browse for Servers** dialog box does not display remote instances of Integration Services. In addition, the options available on the **Connection Options** tab of the **Connect to Server** dialog box, which is displayed by clicking the **Options** button, are not applicable to Integration Services connections.

## Eliminating the "Access Is Denied" Error

When a user without sufficient rights attempts to connect to an instance of Integration Services on a remote server, the server responds with an "Access is denied" error message. You can avoid this error message by ensuring that users have the required DCOM permissions.

### To configure rights for remote users on Windows Server 2003 or Windows XP

1. If the user is not a member of the local Administrators group, add the user to the Distributed COM Users group. You can do this in the Computer Management MMC snap-in accessed from the **Administrative Tools** menu.
2. Open Control Panel, double-click **Administrative Tools**, and then double-click **Component Services** to start the Component Services MMC snap-in.
3. Expand the **Component Services** node in the left pane of the console. Expand the **Computers** node, expand **My Computer**, and then click the **DCOM Config** node.
4. Select the **DCOM Config** node, and then select SQL Server Integration Services 11.0 in the list of applications that can be configured.
5. Right-click on SQL Server Integration Services 11.0 and select **Properties**.
6. In the **SQL Server Integration Services 11.0 Properties** dialog box, select the **Security** tab.

7. Under **Launch and Activation Permissions**, select **Customize**, then click **Edit** to open the **Launch Permission** dialog box.
8. In the **Launch Permission** dialog box, add or delete users, and assign the appropriate permissions to the appropriate users and groups. The available permissions are Local Launch, Remote Launch, Local Activation, and Remote Activation. The Launch rights grant or deny permission to start and stop the service; the Activation rights grant or deny permission to connect to the service.
9. Click OK to close the dialog box.
10. Under **Access Permissions**, repeat steps 7 and 8 to assign the appropriate permissions to the appropriate users and groups.
11. Close the MMC snap-in.
12. Restart the Integration Services service.

**To configure rights for remote users on Windows 2000 with the latest service packs**

1. Run **dcomcnfg.exe** at the command prompt.
2. On the **Applications** page of the **Distributed COM Configuration Properties** dialog box, select SQL Server Integration Services 11.0 and then click **Properties**.
3. Select the **Security** page.
4. Use the two separate dialog boxes to configure **Access Permissions** and **Launch Permissions**. You cannot distinguish between remote and local access - Access permissions include local and remote access, and Launch permissions include local and remote launch.
5. Close the dialog boxes and **dcomcnfg.exe**.
6. Restart the Integration Services service.

**Connecting by using a Local Account**

If you are working in a local Windows account on a client computer, you can connect to the Integration Services service on a remote computer only if a local account that has the same name and password and the appropriate rights exists on the remote computer.

**By default the SSIS service does not support delegation**

By default the SQL Server Integration Services service does not support the delegation of credentials, or what is sometimes referred to as a double hop. In this scenario, you are working on a client computer, the Integration Services service is running on a second computer, and SQL Server is running on a third computer. First, SQL Server Management Studio successfully passes your credentials from the client computer to the second computer on which the Integration Services service is running. Then, however, the Integration Services service cannot delegate your credentials from the second computer to the third computer on which SQL Server is running.

You can enable delegation of credentials by granting the **Trust this user for delegation to any service (Kerberos Only)** right to the SQL Server service account, which launches the Integration Services service (ISServerExec.exe) as a child process. Before you grant this right, consider whether it meets the security requirements of your organization.

For more info, see [Getting Cross Domain Kerberos and Delegation working with SSIS Package](#).

## Configure the firewall

The Windows firewall system helps prevent unauthorized access to computer resources over a network connection. To access Integration Services through this firewall, you have to configure the firewall to enable access.

## IMPORTANT

To manage packages that are stored on a remote server, you do not have to connect to the instance of the Integration Services service on that remote server. Instead, edit the configuration file for the Integration Services service so that SQL Server Management Studio displays the packages that are stored on the remote server.

The Integration Services service uses the DCOM protocol. For more information about how the DCOM protocol works through firewalls, see the article, "[Using Distributed COM with Firewalls](#)," in the MSDN Library.

There are many firewall systems available. If you are running a firewall other than Windows firewall, see your firewall documentation for information that is specific to the system you are using.

If the firewall supports application-level filtering, you can use the user interface that Windows provides to specify the exceptions that are allowed through the firewall, such as programs and services. Otherwise, you have to configure DCOM to use a limited set of TCP ports. The Microsoft website link previously provided includes information about how to specify the TCP ports to use.

The Integration Services service uses port 135, and the port cannot be changed. You have to open TCP port 135 for access to the service control manager (SCM). SCM performs tasks such as starting and stopping Integration Services services and transmitting control requests to the running service.

The information in the following section is specific to Windows firewall. You can configure the Windows firewall system by running a command at the command prompt, or by setting properties in the Windows firewall dialog box.

For more information about the default Windows firewall settings, and a description of the TCP ports that affect the Database Engine, Analysis Services, Reporting Services, and Integration Services, see [Configure the Windows Firewall to Allow SQL Server Access](#).

## Configuring a Windows firewall

You can use the following commands to open TCP port 135, add MsDtsSrvr.exe to the exception list, and specify the scope of unblocking for the firewall.

### To configure a Windows firewall using the Command Prompt window

1. Run the command:

```
netsh firewall add portopening protocol=TCP port=135 name="RPC (TCP/135)" mode=ENABLE scope=SUBNET
```

2. Run the command:

```
netsh firewall add allowedprogram program="%ProgramFiles%\Microsoft SQL Server\100\DTS\Binn\MsDtsSrvr.exe" name="SSIS Service" scope=SUBNET
```

## NOTE

To open the firewall for all computers, and also for computers on the Internet, replace scope=SUBNET with scope=ALL.

The following procedure describes how to use the Windows user interface to open TCP port 135, add MsDtsSrvr.exe to the exception list, and specify the scope of unblocking for the firewall.

### To configure a firewall using the Windows firewall dialog box

1. In the Control Panel, double-click **Windows Firewall**.
2. In the **Windows Firewall** dialog box, click the **Exceptions** tab and then click **Add Program**.
3. In the **Add a Program** dialog box, **click Browse**, navigate to the Program Files\Microsoft SQL Server\100\DTS\Binn folder, click MsDtsSrvr.exe, and then click **Open**. Click **OK** to close the **Add a Program** dialog box.

4. On the **Exceptions** tab, click **Add Port**.
5. In the **Add a Port** dialog box, type **RPC(TCP/135)** or another descriptive name in the **Name** box, type **135** in the **Port Number** box, and then select **TCP**.

**IMPORTANT**

Integration Services service always uses port 135. You cannot specify a different port.

6. In the **Add a Port** dialog box, you can optionally click **Change Scope** to modify the default scope.
7. In the **Change Scope** dialog box, select **My network (subnet only)** or type a custom list, and then click **OK**.
8. To close the **Add a Port** dialog box, click **OK**.
9. To close the **Windows Firewall** dialog box, click **OK**.

**NOTE**

To configure the Windows firewall, this procedure uses the **Windows Firewall** item in Control Panel. The **Windows Firewall** item only configures the firewall for the current network location profile. However, you can also configure the Windows firewall by using the **netsh** command line tool or the Microsoft Management Console (MMC) snap-in named Windows firewall with Advanced Security. For more information about these tools, see [Configure the Windows Firewall to Allow SQL Server Access](#).

# Security Overview (Integration Services)

4/25/2017 • 9 min to read • [Edit Online](#)

Security in SQL Server Integration Services consists of several layers that provide a rich and flexible security environment. These security layers include the use of digital signatures, package properties, SQL Server database roles, and operating system permissions. Most of these security features fall into the categories of identity and access control.

## Threat and Vulnerability Mitigation

Although Integration Services includes a variety of security mechanisms, packages and the files that packages create or use could be exploited for malicious purposes.

The following table describes these risks and the proactive steps that you can take to lessen the risks.

THREAT OR VULNERABILITY	DEFINITION	MITIGATION
Package source	The source of a package is the individual or organization that created the package. Running a package from an unknown or untrusted source might be risky.	Identify the source of a package by using a digital signature, and run packages that come from only known, trusted sources. For more information, see <a href="#">Identify the Source of Packages with Digital Signatures</a> .
Package contents	Package contents include the elements in the package and their properties. The properties can contain sensitive data such as a password or a connection string. Package elements such as an SQL statement can reveal the structure of your database.	<p>Control access to a package and to the contents by doing the following steps:</p> <p>1) To control access to the package itself, apply SQL Server security features to packages that are saved to the <b>msdb</b> database in an instance of SQL Server. To packages that are saved in the file system, apply file system security features, such as access controls lists (ACLs).</p> <p>2) To control access to the package's contents, set the protection level of the package.</p> <p>For more information, see <a href="#">Security Overview (Integration Services)</a> and <a href="#">Access Control for Sensitive Data in Packages</a>.</p>
Package output	When you configure a package to use configurations, checkpoints, and logging, the package stores this information outside the package. The information that is stored outside the package might contain sensitive data.	<p>To protect configurations and logs that the package saves to SQL Server database tables, use SQL Server security features.</p> <p>To control access to files, use the access control lists (ACLs) available in the file system.</p> <p>For more information, see <a href="#">Access to Files Used by Packages</a></p>

# Identity Features

By implementing identity features in your packages, you can achieve the following goal:

## **Ensure that you only open and run packages from trusted sources.**

To ensure that you only open and run packages from trusted sources, you first have to identify the source of packages. You can identify the source by signing packages with certificates. Then, when you open or run the packages, you can have Integration Services check for the presence and the validity of the digital signatures. For more information, see [Identify the Source of Packages with Digital Signatures](#).

# Access Control Features

By implementing identity features in your packages, you can achieve the following goal:

## **Ensure that only authorized users open and run packages.**

To ensure that only authorized users open and run packages, you have to control access to the following information:

- Control access to the contents of packages, especially sensitive data.
- Control access to packages and package configurations that are stored in SQL Server.
- Control access to packages and to related files such as configurations, logs, and checkpoint files that are stored in the file system.
- Control access to the Integration Services service and to the information about packages that the service displays in SQL Server Management Studio.

## **Controlling Access to the Contents of Packages**

To help restrict access to the contents of a package, you can encrypt packages by setting the `ProtectionLevel` property of the package. You can set this property to the level of protection that your package requires. For example, in a team development environment, you can encrypt a package by using a password that is known only to the team members who work on the package.

When you set the `ProtectionLevel` property of a package, Integration Services automatically detects sensitive properties and handles these properties according to the specified package protection level. For example, you set the `ProtectionLevel` property for a package to a level that encrypts sensitive information with a password. For this package, Integration Services automatically encrypts the values of all sensitive properties and will not display the corresponding data without the correct password being supplied.

Typically, Integration Services identifies properties as sensitive if those properties contain information, such as a password or a connection string, or if those properties correspond to variables or task-generated XML nodes. Whether Integration Services considers a property sensitive depends on whether the developer of the Integration Services component, such as a connection manager or task, has designated the property as sensitive. Users cannot add properties to, nor can they remove properties from, the list of properties that are considered sensitive. If you write custom tasks, connection managers, or data flow components, you can specify which properties Integration Services should treat as sensitive.

For more information, see [Access Control for Sensitive Data in Packages](#).

## **Controlling Access to Packages**

You can save Integration Services packages to the `msdb` database in an instance of SQL Server, or to the file system as XML files that have the `.dtsx` file name extension. For more information, see [Save Packages](#).

### **Saving Packages to the msdb Database**

Saving the packages to the `msdb` database helps provide security at the server, database, and table levels. In the



msdb database, Integration Services packages are stored in the sysssispackages table. Because the packages are saved to the sysssispackages and sysdtspackages tables in the msdb database, the packages are automatically backed up when you backup the msdb database.

SQL Server packages stored in the msdb database can also be protected by applying the Integration Services database-level roles. Integration Services includes three fixed database-level roles db\_ssisadmin, db\_ssisltduser, and db\_ssisoperator for controlling access to packages. A reader and a writer role can be associated with each package. You can also define custom database-level roles to use in Integration Services packages. Roles can be implemented only on packages that are saved to the msdb database in an instance of SQL Server. For more information, see [Integration Services Roles \(SSIS Service\)](#).

#### **Saving Packages to the File System**

If you store packages to the file system instead of in the msdb database, make sure to secure the package files and the folders that contain package files.

#### **Controlling Access to Files Used by Packages**

Packages that have been configured to use configurations, checkpoints, and logging generate information that is stored outside the package. This information might be sensitive and should be protected. Checkpoint files can be saved only to the file system, but configurations and logs can be saved to the file system or to tables in a SQL Server database. Configurations and logs that are saved to SQL Server are subject to SQL Server security, but information written to the file system requires additional security.

For more information, see [Access to Files Used by Packages](#).

#### **Storing Package Configurations Securely**

Package configurations can be saved to a table in a SQL Server database or to the file system.

Configurations can be saved to any SQL Server database, not just the msdb database. Thus, you are able to specify which database serves as the repository of package configurations. You can also specify the name of the table that will contain the configurations, and Integration Services automatically creates the table with the correct structure. Saving the configurations to a table makes it possible to provide security at the server, database, and table levels. In addition, configurations that are saved to SQL Server are automatically backed up when you back up the database.

If you store configurations in the file system instead of in SQL Server, make sure to secure the folders that contain the package configuration files.

For more information about configurations, see [Package Configurations](#).

#### **Controlling Access to the Integration Services Service**

SQL Server Management Studio uses the SQL Server service to list stored packages. To prevent unauthorized users from viewing information about packages that are stored on local and remote computers, and thereby learning private information, restrict access to computers that run the SQL Server service.

For more information, see [Access to the Integration Services Service](#).

## **Access to Files Used by Packages**

The package protection level does not protect files that are stored outside the package. These files include the following:

- Configuration files
- Checkpoint files
- Log files

These files must be protected separately, especially if they include sensitive information.

#### **Configuration Files**

If you have sensitive information in a configuration, such as login and password information, you should consider saving the configuration to SQL Server, or use an access control list (ACL) to restrict access to the location or folder where you store the files and allow access only to certain accounts. Typically, you would grant access to the accounts that you permit to run packages, and to the accounts that manage and troubleshoot packages, which may include reviewing the contents of configuration, checkpoint, and log files. SQL Server provides the more secure storage because it offers protection at the server and database levels. To save configurations to SQL Server, you use the SQL Server configuration type. To save to the file system, you use the XML configuration type.

For more information, see [Package Configurations](#), [Create Package Configurations](#), and [Security Considerations for a SQL Server Installation](#).

### Checkpoint Files

Similarly, if the checkpoint file that the package uses includes sensitive information, you should use an access control list (ACL) to secure the location or folder where you store the file. Checkpoint files save current state information on the progress of the package as well as the current values of variables. For example, the package may include a custom variable that contains a telephone number. For more information, see [Restart Packages by Using Checkpoints](#).

### Log Files

Log entries that are written to the file system should also be secured using an access control list (ACL). Log entries can also be stored in SQL Server tables and protected by SQL Server security. Log entries may include sensitive information. For example, if the package contains an Execute SQL task that constructs an SQL statement that refers to a telephone number, the log entry for the SQL statement includes the telephone number. The SQL statement may also reveal private information about table and column names in databases. For more information, see [Integration Services \(SSIS\) Logging](#).

## Access to the Integration Services Service

Package protection levels can limit who is allowed to edit and execute a package. Additional protection is needed to limit who can view the list of packages currently running on a server and who can stop currently executing packages in SQL Server Management Studio.

SQL Server Management Studio uses the SQL Server service to list running packages. Members of the Windows Administrators group can view and stop all currently running packages. Users who are not members of the Administrators group can view and stop only packages that they started.

It is important to restrict access to computers that run an SQL Server service, especially an SQL Server service that can enumerate remote folders. Any authenticated user can request the enumeration of packages. Even if the service does not find the service, the service enumerates folders. These folder names may be useful to a malicious user. If an administrator has configured the service to enumerate folders on a remote machine, users may also be able to see folder names that they would normally not be able to see.

## Related Tasks

The following list contains links to topics that show you how to perform a certain task pertaining to the security.

- [Create a User-Defined Role](#)
- [Assign a Reader and Writer Role to a Package](#)
- [Implement a Signing Policy by Setting a Registry Value](#)
- [Sign a Package by Using a Digital Certificate](#)
- [Set or Change the Protection Level of Packages](#)

# Monitor Running Packages and Other Operations

4/19/2017 • 9 min to read • [Edit Online](#)

You can monitor Integration Services package executions, project validations, and other operations by using one of more of the following tools. Certain tools such as data taps are available only for projects that are deployed to the Integration Services server.

- Logs

For more information, see [Integration Services \(SSIS\) Logging](#).

- Reports

For more information, see [Reports for the Integration Services Server](#).

- Views

For more information, see [Views \(Integration Services Catalog\)](#).

- Performance counters

For more information, see [Performance Counters](#).

- Data taps

## Operation Types

Several different types of operations are monitored in the **SSISDB** catalog, on the Integration Services server. Each operation can have multiple messages associated with it. Each message can be classified into one of several different types. For example, a message can be of type Information, Warning, or Error. For the full list of message types, see the documentation for the Transact-SQL [catalog.operation\\_messages \(SSISDB Database\)](#) view. For a full list of the operations types, see [catalog.operations \(SSISDB Database\)](#).

Nine different status types are used to indicate the status of an operation. For a full list of the status types, see the [catalog.operations \(SSISDB Database\)](#) view.

## Active Operations Dialog Box

Use the **Active Operations** dialog box to view the status of currently running Integration Services operations on the Integration Services server, such as deployment, validation, and package execution. This data is stored in the SSISDB catalog.

For more information about related Transact-SQL views, see [catalog.operations \(SSISDB Database\)](#), [catalog.validations \(SSISDB Database\)](#), and [catalog.executions \(SSISDB Database\)](#).

### Open the Active Operations Dialog Box

1. Open SQL Server Management Studio.
2. Connect Microsoft SQL Server Database Engine
3. In Object Explorer, expand the **Integration Services** node, right-click **SSISDB**, and then click **Active Operations**.

### Configure the Options

#### Type

Specifies the type of operation. The following are the possible values for the **Type** field and the corresponding values in the operations\_type column of the Transact-SQL **catalog.operations** view.

Integration Services initialization	1
Operations cleanup (SQL Agent job)	2
Project versions cleanup (SQL Agent job)	3
Deploy project	101
Restore project	106
Create and start package execution	200
Stop operation (stopping a validation or execution	202
Validate project	300
Validate package	301
Configure catalog	1000

### Stop

Click to stop a currently running operation.

## Viewing and Stopping Packages Running on the Integration Services Server

The **SSISDB** database stores execution history in internal tables that are not visible to users. However it exposes the information that you need through public views that you can query. It also provides stored procedures that you can call to perform common tasks related to packages.

Typically you manage Integration Services objects on the server in SQL Server Management Studio. However you can also query the database views and call the stored procedures directly, or write custom code that calls the managed API. SQL Server Management Studio and the managed API query the views and call the stored procedures to perform many of their tasks. For example, you can view the list of Integration Services packages that are currently running on the server, and request packages to stop if you have to.

### Viewing the List of Running Packages

You can view the list of packages that are currently running on the server in the **Active Operations** dialog box. For more information, see [Active Operations Dialog Box](#).

For information about the other methods that you can use to view the list of running packages, see the following topics.

#### Transact-SQL access

To view the list of packages that are running on the server, query the view, [catalog.executions \(SSISDB Database\)](#) for packages that have a status of 2.

#### Programmatic access through the managed API

See the [Microsoft.SqlServer.Management.IntegrationServices](#) namespace and its classes.

### Stopping a Running Package

You can request a running package to stop in the **Active Operations** dialog box. For more information, see [Active Operations Dialog Box](#).

For information about the other methods that you can use to stop a running package, see the following topics.

Transact-SQL access

To stop a package that is running on the server, call the stored procedure, [catalog.stop\\_operation \(SSISDB Database\)](#).

Programmatic access through the managed API

See the [Microsoft.SqlServer.Management.IntegrationServices](#) namespace and its classes.

### Viewing the History of Packages That Have Run

To view the history of packages that have run in Management Studio, use the **All Executions** report. For more information on the **All Executions** report and other standard reports, see [Reports for the Integration Services Server](#).

For information about the other methods that you can use to view the history of running packages, see the following topics.

Transact-SQL access

To view information about packages that have run, query the view, [catalog.executions \(SSISDB Database\)](#).

Programmatic access through the managed API

See the [Microsoft.SqlServer.Management.IntegrationServices](#) namespace and its classes.

## Reports for the Integration Services Server

In the current release of SQL Server Integration Services, standard reports are available in SQL Server Management Studio to help you monitor Integration Services projects that have been deployed to the Integration Services server. These reports help you to view package status and history, and, if necessary, identify the cause of package execution failures.

At the top of each report page, the back icon takes you to the previous page you viewed, the refresh icon refreshes the information displayed on the page, and the print icon allows you to print the current page.

For information on how to deploy packages to the Integration Services server, see [Deploy Integration Services \(SSIS\) Projects and Packages](#).

### Integration Services Dashboard

The **Integration Services Dashboard** report provides an overview of all the package executions on the SQL Server instance. For each package that has run on the server, the dashboard allows you to "zoom in" to find specific details on package execution errors that may have occurred.

The report displays the following sections of information.

SECTION	DESCRIPTION
<b>Execution Information</b>	Shows the number of executions that are in different states (failed, running, succeeded, others) in the past 24 hours.
<b>Package Information</b>	Shows the total number of packages that have been executed in the past 24 hours.
<b>Connection Information</b>	Shows the connections that have been used in failed executions in the past 24 hours.

SECTION	DESCRIPTION
<b>Package Detailed Information</b>	<p>Shows the details of the completed executions that have occurred in the past 24 hours. For example, this section shows the number of failed executions versus the total number of executions, the duration of an executions (in seconds), and the average duration of executions for over the past three months.</p> <p>You can view additional information for a package by clicking <b>Overview</b>, <b>All Messages</b>, and <b>Execution Performance</b>.</p> <p>The <b>Execution Performance</b> report shows the duration of the last execution instance, as well as the start and end times, and the environment that was applied.</p> <p>The chart and associated table included in the <b>Execution Performance</b> report shows the duration of the past 10 successful executions of the package. The table also shows the average execution duration over a three-month period. Different environments and different literal values may have been applied at runtime for these 10 successful executions of the package.</p> <p>Finally, the <b>Execution Performance</b> report shows the Active Time and Total Time for the package data flow components. The Active Time refers to the total amount of time that component has spent executing in all phases, and the Total Time refers to the total time elapsed for a component. The report only displays this information for package components when the logging level of the last package execution was set to Performance or Verbose.</p> <p>The <b>Overview</b> report shows the state of package tasks. The <b>Messages</b> report shows the event messages and error messages for the package and tasks, such as reporting the start and end times, and the number of rows written.</p> <p>You can also click <b>View Messages</b> in the <b>Overview</b> report to navigate to the <b>Messages</b> report. You can also click <b>View Overview</b> in the <b>Messages</b> report to navigate to the <b>Overview</b> report.</p>

You can filter the table displayed on any page by clicking **Filter** and then selecting criteria in the **Filter Settings** dialog. The filter criteria that are available depend on the data being displayed. You can change the sort order of the report by clicking the sort icon in the **Filter Settings** dialog.

### All Executions Report

The **All Executions Report** displays a summary of all Integration Services executions that have been performed on the server. There can be multiple executions of the sample package. Unlike the **Integration Services Dashboard** report, you can configure the **All Executions** report to show executions that have started during a range of dates. The dates can span multiple days, months, or years.

The report displays the following sections of information.

SECTION	DESCRIPTION
Filter	Shows the current filter applied to the report, such as the Start time range.

SECTION	DESCRIPTION
Execution Information	Shows the start time, end time, and duration for each package execution. You can view a list of the parameter values that were used with a package execution, such as values that were passed to a child package using the Execute Package task. To view the parameter list, click Overview.

For more information about using the Execute Package task to make values available to a child package, see [Execute Package Task](#).

For more information about parameters, see [Integration Services \(SSIS\) Package and Project Parameters](#).

## All Connections

The **All Connections** report provides the following information for connections that have failed, for executions that have occurred on the SQL Server instance.

The report displays the following sections of information.

SECTION	DESCRIPTION
Filter	Shows the current filter applied to the report, such as connections with a specified string and the <b>Last failed time</b> range.  You set the <b>Last failed time</b> range to display only connection failures that occurred during a range of dates. The range can span multiple days, months, or years.
Details	Shows the connection string, number of executions during which a connection failed, and the date when the connection last failed.

## All Operations Report

The **All Operations Report** displays a summary of all Integration Services operations that have been performed on the server, including package deployment, validation, and execution, as well as other administrative operations. As with the Integration Services Dashboard, you can apply a filter to the table to narrow down the information displayed.

## All Validations Report

The **All Validations Report** displays a summary of all Integration Services validations that have been performed on the server. The summary displays information for each validation such as status, start time, and end time. Each summary entry includes a link to messages generated during validation. As with the Integration Services Dashboard, you can apply a filter to the table to narrow down the information displayed.

## Custom Reports

You can add a custom report (.rdl file) to the **SSISDB** catalog node under the **Integration Services Catalogs** node in SQL Server Management Studio. Before adding the report, confirm that you are using a three-part naming convention to fully qualify the objects you reference such as a source table. Otherwise, SQL Server Management Studio will display an error. The naming convention is <database>.<owner>.<object>. An example would be SSISDB.internal.executions.

### NOTE

When you add custom reports to the **SSISDB** node under the **Databases** node, the SSISDB prefix is not necessary.

For instructions on how to create and add a custom report, see [Add a Custom Report to Management Studio](#).

## View Reports for the Integration Services Server

In the current release of SQL Server Integration Services, standard reports are available in SQL Server Management Studio to help you monitor Integration Services projects that have been deployed to the Integration Services server. For more information about the reports, see [Reports for the Integration Services Server](#).

### To view reports for the Integration Services server

1. In SQL Server Management Studio, expand the **Integration Services Catalogs** node in Object Explorer.
2. Right-click **SSISDB**, click **Reports**, and then click **Standard Reports**.
3. Click one more of the following to view a report.
  - **Integration Services Dashboard**
  - **All Executions**
  - **All Validations**
  - **All Operations**
  - **All Connections**

## See Also

[Execution of Projects and Packages](#)

[Troubleshooting Reports for Package Execution](#)



# Troubleshoot Integration Services (SSIS) Packages





3/24/2017 • 1 min to read • [Edit Online](#)

## In this section

- [Troubleshooting Tools for Package Development](#)
- [Troubleshooting Tools for Package Connectivity](#)
- [Troubleshooting Tools for Package Execution](#)
- [Troubleshooting Reports for Package Development](#)
- [Generating Dump Files for Package Execution](#)

# Views (Integration Services Catalog)

3/24/2017 • 2 min to read • [Edit Online](#)

**THIS TOPIC APPLIES TO:**  SQL Server (starting with 2012)  Azure SQL Database  Azure SQL Data Warehouse  Parallel Data Warehouse

This section describes the Transact-SQL views that are available for administering Integration Services projects that have been deployed to an instance of SQL Server.

Query the Integration Services views to inspect objects, settings, and operational data that are stored in the **SSISDB** catalog.

The default name of the catalog is SSISDB. The objects that are stored in the catalog include projects, packages, parameters, environments, and operational history.

You can use the database views and stored procedures directly, or write custom code that calls the managed API. Management Studio and the managed API query the views and call the stored procedures that are described in this section to perform many of their tasks.

## In This Section

[catalog.catalog\\_properties \(SSISDB Database\)](#)

Displays the properties of the Integration Services catalog.

[catalog.effective\\_object\\_permissions \(SSISDB Database\)](#)

Displays the effective permissions for the current principal for all objects in the Integration Services catalog.

[catalog.environment\\_variables \(SSISDB Database\)](#)

Displays the environment variable details for all environments in the Integration Services catalog.

[catalog.environments \(SSISDB Database\)](#)

Displays the environment details for all environments in the Integration Services catalog. Environments contain variables that can be referenced by Integration Services projects.

[catalog.execution\\_parameter\\_values \(SSISDB Database\)](#)

Displays the actual parameter values that are used by Integration Services packages during an instance of execution.

[catalog.executions \(SSISDB Database\)](#)

Displays the instances of package execution in the Integration Services catalog. Packages that are executed with the Execute Package task run in the same instance of execution as the parent package.

[catalog.explicit\\_object\\_permissions \(SSISDB Database\)](#)

Displays only the permissions that have been explicitly assigned to the user.

[catalog.extended\\_operation\\_info \(SSISDB Database\)](#)

Displays extended information for all operations in the Integration Services catalog.

[catalog.folders \(SSISDB Database\)](#)

Displays the folders in the Integration Services catalog.

[catalog.object\\_parameters \(SSISDB Database\)](#)

Displays the parameters for all packages and projects in the Integration Services catalog.

[catalog.object\\_versions \(SSISDB Database\)](#)

Displays the versions of objects in the Integration Services catalog. In this release, only versions of projects are supported in this view.

[catalog.operation\\_messages \(SSISDB Database\)](#)

Displays messages that are logged during operations in the Integration Services catalog.

[catalog.operations \(SSISDB Database\)](#)

Displays the details of all operations in the Integration Services catalog.

[catalog.packages \(SSISDB Database\)](#)

Displays the details for all packages that appear in the Integration Services catalog.

[catalog.environment\\_references \(SSISDB Database\)](#)

Displays the environment references for all projects in the Integration Services catalog.

[catalog.projects \(SSISDB Database\)](#)

Displays the details for all projects that appear in the Integration Services catalog.

[catalog.validations \(SSISDB Database\)](#)

Displays the details of all project and package validations in the Integration Services catalog.

[catalog.master\\_properties \(SSISDB Database\)](#)





Displays the properties of the Integration Services Scale Out Master.

[catalog.worker\\_agents \(SSISDB Database\)](#)

Displays the information of Integration Services Scale Out Worker.

# Stored Procedures (Integration Services Catalog)

3/24/2017 • 4 min to read • [Edit Online](#)

**THIS TOPIC APPLIES TO:**  SQL Server (starting with 2012)  Azure SQL Database  Azure SQL Data Warehouse  Parallel Data Warehouse

This section describes the Transact-SQL stored procedures that are available for administering Integration Services projects that have been deployed to an instance of SQL Server.

Call the Integration Services stored procedures to add, remove, modify, or execute objects that are stored in the **SSISDB** catalog.

The default name of the catalog is SSISDB. The objects that are stored in the catalog include projects, packages, parameters, environments, and operational history.

You can use the database views and stored procedures directly, or write custom code that calls the managed API. Management Studio and the managed API query the views and call the stored procedures that are described in this section to perform many of their tasks.

## In This Section

[catalog.add\\_data\\_tap](#)

Adds a data tap on the output of a component in a package data flow.

[catalog.add\\_data\\_tap\\_by\\_guid](#)

Adds a data tap to a specific data flow path in a package data flow.

[catalog.check\\_schema\\_version](#)

Determines whether the SSISDB catalog schema and the Integration Services binaries (ISServerExec and SQLCLR assembly) are compatible.

[catalog.clear\\_object\\_parameter\\_value \(SSISDB Database\)](#)

Clears the value of a parameter for an existing Integration Services project or package that is stored on the server.

[catalog.configure\\_catalog \(SSISDB Database\)](#)

Configures the Integration Services catalog by setting a catalog property to a specified value.

[catalog.create\\_environment \(SSISDB Database\)](#)

Creates an environment in the Integration Services catalog.

[catalog.create\\_environment\\_reference \(SSISDB Database\)](#)

Creates an environment reference for a project in the Integration Services catalog.

[catalog.create\\_environment\\_variable \(SSISDB Database\)](#)

Create an environment variable in the Integration Services catalog.

[catalog.create\\_execution \(SSISDB Database\)](#)

Creates an instance of execution in the Integration Services catalog.

[catalog.create\\_execution\\_dump](#)

Causes a running package to pause and create a dump file.

[catalog.create\\_folder \(SSISDB Database\)](#)

Creates a folder in the Integration Services catalog.

[catalog.delete\\_environment \(SSISDB Database\)](#)

Deletes an environment from a folder in the Integration Services catalog.

[catalog.delete\\_environment\\_reference \(SSISDB Database\)](#)

Deletes an environment reference from a project in the Integration Services catalog.

[catalog.delete\\_environment\\_variable \(SSISDB Database\)](#)

Deletes an environment variable from an environment in the Integration Services catalog.

[catalog.delete\\_folder \(SSISDB Database\)](#)

Deletes a folder from the Integration Services catalog.

[catalog.delete\\_project \(SSISDB Database\)](#)

Deletes an existing project from a folder in the Integration Services catalog.

[catalog.deny\\_permission \(SSISDB Database\)](#)

Denies a permission on a securable object in the Integration Services catalog.

[catalog.deploy\\_project \(SSISDB Database\)](#)

Deploys a project to a folder in the Integration Services catalog or updates an existing project that has been deployed previously.

[catalog.get\\_parameter\\_values \(SSISDB Database\)](#)

Resolves and retrieves the default parameter values from a project and corresponding packages in the Integration Services catalog.

[catalog.get\\_project \(SSISDB Database\)](#)

Retrieves the properties of an existing project in the Integration Services catalog.

[catalog.grant\\_permission \(SSISDB Database\)](#)

Grants a permission on a securable object in the Integration Services catalog.

[catalog.move\\_environment \(SSISDB Database\)](#)

Moves an environment from one folder to another within the Integration Services catalog.

[\[catalog.move\\_project \(\(SSISDB Database\)\)\]\(../Topic/catalog.move\\_project%20\(\(SSISDB%20Database\).md\)](#)

Moves a project from one folder to another within the Integration Services catalog.

[catalog.remove\\_data\\_tap](#)

Removes a data tap from a component output that is in an execution.

[catalog.rename\\_environment \(SSISDB Database\)](#)

Renames an environment in the Integration Services catalog.

[catalog.rename\\_folder \(SSISDB Database\)](#)

Renames a folder in the Integration Services catalog.

[catalog.restore\\_project \(SSISDB Database\)](#)

Restores a project in the Integration Services catalog to a previous version.

[catalog.revoke\\_permission \(SSISDB Database\)](#)

Revokes a permission on a securable object in the Integration Services catalog.

[catalog.set\\_environment\\_property \(SSISDB Database\)](#)

Sets the property of an environment in the Integration Services catalog.

[catalog.set\\_environment\\_reference\\_type \(SSISDB Database\)](#)

Sets the reference type and environment name associated with an existing environment reference for a project in the Integration Services catalog.

[catalog.set\\_environment\\_variable\\_property \(SSISDB Database\)](#)

Sets the property of an environment variable in the Integration Services catalog.

[catalog.set\\_environment\\_variable\\_protection \(SSISDB Database\)](#)

Sets the sensitivity bit of an environment variable in the Integration Services catalog.

[catalog.set\\_environment\\_variable\\_value \(SSISDB Database\)](#)

Sets the value of an environment variable in the Integration Services catalog.

[catalog.set\\_execution\\_parameter\\_value \(SSISDB Database\)](#)

Sets the value of a parameter for an instance of execution in the Integration Services catalog.

[catalog.set\\_execution\\_property\\_override\\_value](#)

Sets the value of a property for an instance of execution in the Integration Services catalog.

[catalog.set\\_folder\\_description \(SSISDB Database\)](#)

Sets the description of a folder in the Integration Services catalog.

[catalog.set\\_object\\_parameter\\_value \(SSISDB Database\)](#)

Sets the value of a parameter in the Integration Services catalog. Associates the value to an environment variable or assigns a literal value that will be used by default if no other values are assigned.

[catalog.start\\_execution \(SSISDB Database\)](#)

Starts an instance of execution in the Integration Services catalog.

[catalog.startup](#)

Performs maintenance of the state of operations for the SSISDB catalog.

[catalog.stop\\_operation \(SSISDB Database\)](#)

Stops a validation or instance of execution in the Integration Services catalog.

[catalog.validate\\_package \(SSISDB Database\)](#)

Asynchronously validates a package in the Integration Services catalog.

[catalog.validate\\_project \(SSISDB Database\)](#)

Asynchronously validates a project in the Integration Services catalog.

[catalog.add\\_execution\\_worker \(SSISDB Database\)](#)

Adds a Integration Services Scale Out Worker to an instance of execution in Scale Out.

[catalog.enable\\_worker\\_agent \(SSISDB Database\)](#)





Enable a Scale Out Worker for Scale Out Master working with this Integration Services catalog.

[catalog.disable\\_worker\\_agent \(SSISDB Database\)](#)

Disable a Scale Out Worker for Scale Out Master working with this Integration Services catalog.

# Functions - dm\_execution\_performance\_counters

3/24/2017 • 2 min to read • [Edit Online](#)

**THIS TOPIC APPLIES TO:**  SQL Server (starting with 2014)  Azure SQL Database  Azure SQL Data Warehouse  Parallel Data Warehouse

Returns the performance statistics for an execution that is running on the Integration Services server.

## Syntax

```
dm_execution_performance_counters [ @execution_id = ] execution_id
```

## Arguments

[ @execution\_id = ] *execution\_id*

The unique identifier of the execution that contains one or more packages. Packages that are executed with the Execute Package task, run in the same execution as the parent package.

If an execution ID is not specified, performance statistics for multiple executions are returned. If you are a member of the **ssis\_admin** database role, performance statistics for all running executions are returned. If you are not a member of the **ssis\_admin** database role, performance statistics for the running executions for which you have read permissions, are returned. The *execution\_id* is a **BigInt**.

## Remarks

The following table lists the counter name values returned by the dm\_execution\_performance\_counter function.

COUNTER NAME	DESCRIPTION
BLOB bytes read	Number of bytes of binary large object (BLOB) data that the data flow engine reads from all sources.
BLOB bytes written	Number of bytes of BLOB data that the data flow engine writes to all destinations.
BLOB files in use	Number of BLOB files that the data flow engine is using for spooling.
Buffer memory	Amount of memory that is used by the Integration Services buffers, including physical and virtual memory.
Buffers in use	Number of buffer objects, of all types, that all data flow components and the data flow engine are using.
Buffers spooled	Number of buffers written to the disk.
Flat buffer memory	Amount of memory, in bytes, that is used by all flat buffers. Flat buffers are blocks of memory that a component uses to store data.

COUNTER NAME	DESCRIPTION
Flat buffers in use	Number of flat buffers that the data flow engine uses. All flat buffers are private buffers.
Private buffer memory	<p>Amount of memory in use by all private buffers. A private buffer is a buffer that a transformation uses for temporary work.</p> <p>A buffer is not private if the data flow engine creates the buffer to support the data flow.</p>
Private buffers in use	Number of buffers that the transformations use for temporary work.
Rows read	Total number of rows ready the execution.
Rows written	Total number of rows written by the execution.

## Return

The `dm_execution_performance_counters` function returns a table with the following columns, for a running execution. The information returned is for all of the packages contained in the execution. If there are no running executions, an empty table is returned.

COLUMN NAME	COLUMN TYPE	DESCRIPTION	REMARKS
execution_id	<b>BigInt</b>  <b>NULL</b> is not a valid value.	Unique identifier for the execution that contains the package.	
counter_name	<b>nvarchar(128)</b>	The name of the counter.	See the <b>Remarks</b> section of values.
counter_value	<b>BigInt</b>	Value returned by the counter.	

## Example

In the following example, the function returns statistics for a running execution with an ID of 34.

```
select * from [catalog].[dm_execution_performance_counters] (34)
```

## Example

In the following example, the function returns statistics for all the executions running on the Integration Services server, depending on your permissions.

```
select * from [catalog].[dm_execution_performance_counters] (NULL)
```

## Permissions



This function requires one of the following permissions:

- READ and MODIFY permissions on the instance of execution
- Membership to the **ssis\_admin** database role
- Membership to the **sysadmin** server role

## Errors and Warnings

The following list describes conditions that cause the function to fail.

- The user does not have MODIFY permissions for the specified execution.
- The specified execution ID is not valid.

# Errors and Events Reference (Integration Services)

3/24/2017 • 3 min to read • [Edit Online](#)

This section of the documentation contains information about several errors and events related to Integration Services. Cause and resolution information is included for error messages.

For more information about Integration Services error messages, including a list of most Integration Services errors and their descriptions, see [Integration Services Error and Message Reference](#). However, the list currently does not include troubleshooting information.

## IMPORTANT

Many of the error messages that you may see when you are working with Integration Services come from other components. These may include OLE DB providers, other database components such as the Database Engine and Analysis Services , or other services or components such as the file system, the SMTP server, or Microsoft Message Queueing. To find information about these external error messages, see the documentation specific to the component.

## Error Messages

SYMBOLIC NAME OF ERROR	DESCRIPTION
DTS_E_CACHELOADEDFROMFILE	Indicates that the package cannot run because a Cache Transform transformation is trying to write data to the in-memory cache. However, a Cache connection manager has already loaded a cache file into the in-memory cache.
DTS_E_CANNOTACQUIRECONNECTIONFROMCONNECTIONMANAGER	Indicates that the package cannot run because a specified connection failed.
DTS_E_CANNOTCONVERTBETWEENUNICODEANDNONUNICODESTRINGCOLUMN	Indicates that a data flow component is trying to pass Unicode string data to another component that expects non-Unicode string data in the corresponding column, or vice versa.
DTS_E_CANNOTCONVERTBETWEENUNICODEANDNONUNICODESTRINGCOLUMNS	Indicates that a data flow component is trying to pass Unicode string data to another component that expects non-Unicode string data in the corresponding column, or vice versa.
DTS_E_CANTINSERTCOLUMNTYPE	Indicates that the column cannot be added to the database table because the conversion between the Integration Services column data type and the database column data type is not supported.
DTS_E_CONNECTIONNOTFOUND	Indicates that the package cannot run because the specified connection manager cannot be found.
DTS_E_CONNECTIONREQUIREDFORMETADATA	Indicates that SSIS Designer must connect to a data source to retrieve new or updated metadata for a source or destination, and that it is unable to connect to the data source.

SYMBOLIC NAME OF ERROR	DESCRIPTION
DTS_E_MULTIPLECACHEWRITES	Indicates that the package cannot run because a Cache Transform transformation is trying to write data to the in-memory cache. However, another Cache Transform transformation has already written to the in-memory cache.
DTS_E_PRODUCTLEVELTOLOW	Indicates that the package cannot run because the appropriate version of SQL Server Integration Services is not installed.
DTS_E_READNOTFILLEDCACHE	Indicates that a Lookup transformation is trying to read data from the in-memory cache at the same time that a Cache Transform transformation is writing data to the cache.
DTS_E_UNPROTECTXMLFAILED	Indicates that the system did not decrypt a protected XML node.
DTS_E_WRITEWHILECACHEINUSE	Indicates that a Cache Transform transformation is trying to write data to the in-memory cache at the same time that a Lookup transformation is reading data from the in-memory cache.
DTS_W_EXTERNALMETADATACOLUMNSOUTOFSYNC	Indicates that the column metadata in the data source does not match the column metadata in the source or destination component that is connected to the data source.

## Events (SQLISPackage)

For more information, see [Events Logged by an Integration Services Package](#).

EVENT	DESCRIPTION
SQLISPackage_12288	Indicates that a package started.
SQLISPackage_12289	Indicates that a package has finished running successfully.
SQLISPACKAGE_12291	Indicates that a package was unable to finish running and has stopped.
SQLISPackage_12546	Indicates that a task or other executable in a package has finished its work.
SQLISPackage_12549	Indicates that a warning message was raised in a package.
SQLISPackage_12550	Indicates that an error message was raised in a package.
SQLISPackage_12551	Indicates that a package did not finish its work and stopped.
SQLISPackage_12557	Indicates that a package has finished running.

## Events (SQLISService)

For more information, see [Events Logged by the Integration Services Service](#).

EVENT	DESCRIPTION
SQLISService_256	Indicates that the service is about to start.
SQLISService_257	Indicates that the service has started.
SQLISService_258	Indicates that the service is about to stop.
SQLISService_259	Indicates that the service has stopped.
SQLISService_260	Indicates that the service tried to start, but could not.
SQLISService_272	Indicates that the configuration file does not exist at the specified location.
SQLISService_273	Indicates that the configuration file could not be read or is not valid.
SQLISService_274	Indicates that the registry entry that contains the location of the configuration file does not exist or is empty.

## See Also

[Integration Services Error and Message Reference](#)

# Integration Services Error and Message Reference

4/6/2017 • 218 min to read • [Edit Online](#)

The following tables list predefined Integration Services errors, warnings, and informational messages, in ascending numerical order within each category, along with their numeric codes and symbolic names. Each of these errors is defined as a field in the [Microsoft.SqlServer.Dts.Runtime.Hresults](#) class in the [Microsoft.SqlServer.Dts.Runtime](#) namespace.

This list may be useful when you encounter an error code without its description. The list does not include troubleshooting information at this time.

## IMPORTANT

Many of the error messages that you may see while working with Integration Services come from other components. In this topic, you will find all the errors raised by Integration Services components. If you do not see your error in the list, the error was raised by a component outside Integration Services. These may include OLE DB providers, other database components such as the Database Engine and Analysis Services , or other services or components such as the file system, the SMTP server, Message Queuing (also known as MSMQ), and so forth. To find information about these external error messages, see the documentation specific to the component.

This list contains the following groups of messages:

- [Error Messages \(DTS\\_E\\_\\*\)](#)
- [Warning Messages \(DTS\\_W\\_\\*\)](#)
- [Informational Messages \(DTS\\_I\\_\\*\)](#)
- [General and Event Messages \(DTS\\_MSG\\_\\*\)](#)
- [Success Messages \(DTS\\_S\\_\\*\)](#)
- [Data Flow Component Error Messages \(DTSBC\\_E\\_\\*\)](#)

## Error Messages

The symbolic names of Integration Services error messages begin with **DTS\_E\_**.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0x8002F347	-2147290297	DTS_E_STOREDPROCSTASK_OVERWRITINGSPATDESTINATION	Overwriting Stored Procedure "%1" at destination.
0x8020837E	-2145352834	DTS_E_ADOSRCUNKNOWNTYPEMAPPEDTONTEXT	The data type "%1" found on column "%2" is not supported for the %3. This column will be converted to DT_NTEXT.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0x8020838C	-2145352820	DTS_E_XMLSRCSCHEMACOLUMNNOTINEXTERNALMETADATA	The column %1 in table %2 in the XML schema does not have a mapping in the external metadata columns.
0xC0000032	-1073741774	DTS_E_NOTINITIALIZED	An internal object or variable was not initialized. This is an internal product error. This error is returned when a variable should have a valid value but does not.
0xC0000033	-1073741773	DTS_E_EXPIRED	Integration Services evaluation period has expired.
0xC0000034	-1073741772	DTS_E_NEGATIVEVALUESNOTALLOWED	This property cannot be assigned a negative value. This error occurs when a negative value is assigned to a property that can only contain positive values, such as the COUNT property.
0xC0000035	-1073741771	DTS_E_NEGATIVEINDEXNOTALLOWED	Indexes cannot be negative. This error occurs when a negative value is used as an index to a collection.
0xC00060AB	-1073717077	DTS_E_INVALIDSSISSERVERNAME	Invalid server name "%1". SSIS service does not support multi-instance, use just server name instead of "server name\instance".
0xC0008445	-1073707963	DTS_E_SCRIPTMIGRATIONFAILED64BIT	Migration for VSA scripts can not be done on 64 bit platforms due to lack of Visual Tools for Applications designer support. Run the migration under WOW64 on 64 bit platforms.
0xC000931A	-1073704166	DTS_E_COMMANDDESTINATIONADAPTERSTATIC_ERRORINCOMMAND	The command execution generated errors.
0xC000F427	-1073679321	DTS_E_SSISSTANDALONENOTINSTALLED	To run a SSIS package outside of SQL Server Data Tools (SSDT) you must install %1 of Integration Services or higher.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0010001	-1073676287	DTS_E_VARIABLENOTFOUND	The variable cannot be found. This occurs when an attempt is made to retrieve a variable from the Variables collection on a container during execution of the package, and the variable is not there. The variable name may have changed or the variable is not being created.
0xC0010003	-1073676285	DTS_E_VARIABLEREADONLY	Error trying to write to a read-only variable, "%1".
0xC0010004	-1073676284	DTS_E_MANAGEDCOMPONENTSTORENOTFOUND	Unable to find the directories containing Tasks and Data Flow Task components. Check the integrity of your installation.
0xC0010006	-1073676282	DTS_E_PACKAGENAMETOOLONG	Package name is too long. The limit is 128 characters. Shorten the package name.
0xC0010007	-1073676281	DTS_E_PACKAGEDESCRIPTI ONTOO LONG	Package description is too long. The limit is 1024 characters. Shorten the package description.
0xC0010008	-1073676280	DTS_E_VERCOMMENTSTOO LONG	VersionComments property is too long. The limit is 1024 characters. Try shortening the VersionComments.
0xC0010009	-1073676279	DTS_E_ELEMENTNOTFOUND	The element cannot be found in a collection. This error happens when you try to retrieve an element from a collection on a container during execution of the package and the element is not there.
0xC001000A	-1073676278	DTS_E_PACKAGENOTFOUND	The specified package could not be loaded from the SQL Server database.
0xC001000C	-1073676276	DTS_E_INVALIDVARIABLEVALUE	The variable value assignment is not valid. This error happens when the client or a task assigns a runtime object to a variable value.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC001000D	-1073676275	DTS_E_RESERVEDNAMESPACE	Error assigning namespace to the variable. The namespace "System" is reserved for system use. This error happens when a component or task attempts to create a variable with a namespace of "System".
0xC001000E	-1073676274	DTS_E_CONNECTIONNOTFOUND	The connection "%1" is not found. This error is thrown by Connections collection when the specific connection element is not found.
0xC001000F	-1073676273	DTS_E_64BITVARIABLERECAST	The variable "%1" is a 64-bit integer variable, which is not supported on this operating system. The variable has been recast to 32-bit integer.
0xC0010010	-1073676272	DTS_E_CANTCHANGEREADONLYATTRUNTIME	An attempt change to a read-only attribute on variable "%1" occurred. This error happens when a read-only attribute for a variable is being changed at runtime. Read-only attributes can be changed only at design time.
0xC0010011	-1073676271	DTS_E_VARIABLEINVALIDCONTAINERREF	Invalid attempt to set a variable to a container reference. Variables are not allowed to reference containers.
0xC0010013	-1073676269	DTS_E_INVALIDVARVALUE	Assigning invalid value or object to variable "%1". This error happens when a value is not appropriate for variables.
0xC0010014	-1073676268	DTS_E_GENERICERROR	One or more error occurred. There should be more specific errors preceding this one that explains the details of the errors. This message is used as a return value from functions that encounter errors.
0xC0010016	-1073676266	DTS_E_INVALIDARRAYVALUE	Error getting or setting an array value. The type "%1" is not allowed. This occurs when loading an array into a variable.



HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0010017	-1073676265	DTS_E_UNSUPPORTEDARRAYTYPE	Unsupported type in array. This happens when saving an array of unsupported types into a variable.
0xC0010018	-1073676264	DTS_E_PERSISTENCEERROR	Error loading value "%1" from node "%2".
0xC0010019	-1073676263	DTS_E_INVALIDNODE	Node "%1" is not a valid node. This happens when saving fails.
0xC0010020	-1073676256	DTS_E_ERRORLOADINGTASK	Failed to load task "%1", type "%2". The contact information for this task is "%3".
0xC0010021	-1073676255	DTS_E_ERRORELEMENTNOTINCOLL	Element "%1" does not exist in collection "%2".
0xC0010022	-1073676254	DTS_E_MISSINGOBJECTDATA	The ObjectData element is missing in the XML block of a hosted object. This occurs when the XML parser attempts to locate the data element for an object and it cannot be found.
0xC0010023	-1073676253	DTS_E_VARIABLENOTFOUNDINCOLL	The variable "%1" cannot be found. This error occurs when an attempt to retrieve a variable from a variables collection on a container during execution of the package occurs, and the variable is not there. A variable name may have changed or the variable is not being created.
0xC0010025	-1073676251	DTS_E_HASEMPTYTASKHOSTS	The package cannot execute because it contains tasks that failed to load.
0xC0010026	-1073676250	DTS_E_TASKISEMPTY	The task has failed to load. The contact information for this task is "%1".
0xC0010027	-1073676249	DTS_E_ERRORLOADINGTASKNOCONTACT	Error loading task "%1".
0xC0010028	-1073676248	DTS_E_ERRORATLOADTASK	Error loading task. This happens when loading a task from XML fails.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0010200	-1073675776	DTS_E_MULTIPLECACHEWRITES	The %1 cannot write to the cache because %2 has already written to it.
0xC0010201	-1073675775	DTS_E_SETCACHEFORINSERTFAILED	Failed to prepare the cache for new data.
0xC0010202	-1073675774	DTS_E_SETCACHEFORFILLFAILED	Failed to mark the cache as filled with data.
0xC0010203	-1073675773	DTS_E_READUNINITIALIZEDCACHE	The cache is not initialized and cannot be read by %1.
0xC0010204	-1073675772	DTS_E_SETCACHEFORREADFAILED	Failed to prepare the cache for providing data.
0xC0010205	-1073675771	DTS_E_READNOTFILLEDCACHE	The cache is being written to by %1, and cannot be read by %2.
0xC0010206	-1073675770	DTS_E_WRITEWHILECACHEINUSE	The cache is being read from %1 and cannot be written to by %2.
0xC0011001	-1073672191	DTS_E_CANTLOADFROMNODE	The runtime object cannot be loaded from the specified XML node. This happens when trying to load a package or other object from an XML node that is not of the correct type, such as a non-SSIS XML node.
0xC0011002	-1073672190	DTS_E_OPENPACKAGEFILE	Failed to open package file "%1" due to error 0x%2!8.X! "%3". This happens when loading a package and the file cannot be opened or loaded correctly into the XML document. This can be the result of either providing an incorrect file name was specified when calling LoadPackage or the XML file was specified and has an incorrect format.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0011003	-1073672189	DTS_E_LOADPACKAGEXML	Failed to load XML due to error 0x%1!8.8X! "%2". This happens when loading a package and the file cannot be opened or loaded correctly into XML document. This can be the result of either providing an incorrect file name to the LoadPackage method or the XML file specified having an incorrect format.
0xC0011004	-1073672188	DTS_E_LOADPACKAGEXMLFILE	Failed to load XML from package file "%1" due to error 0x%2!8.8X! "%3". This happens when loading a package and the file cannot be opened or loaded correctly into an XML document. This can be the result of either providing an incorrect file name to the LoadPackage method or the XML file specified having an incorrect format.
0xC0011005	-1073672187	DTS_E_OPENFILE	Failed to open package file. This happens when loading a package and the file cannot be opened or loaded correctly into an XML document. This can be the result of either providing an incorrect file name to the LoadPackage method or the XML file specified having an incorrect format.
0xC0011006	-1073672186	DTS_E_UNABLETODECODEBINARYFORMAT	Unable to decode a binary format in the package.
0xC0011007	-1073672185	DTS_E_FUNDAMENTALLOADINGERROR	Unable to load the package as XML because of package does not have a valid XML format. A specific XML parser error will be posted.
0xC0011008	-1073672184	DTS_E_LOADFROMXML	Error loading from XML. No further detailed error information can be specified for this problem because no Events object was passed where detailed error information can be stored.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0011009	-1073672183	DTS_E_XMLDOMERROR	Cannot create an instance of the XML Document Object Model. MSXML may not be registered.
0xC001100D	-1073672179	DTS_E_CANNOTLOADOLDPACKAGES	The package cannot be loaded. This occurs when attempting to load an older version package, or the package file refers to an invalid structured object.
0xC001100E	-1073672178	DTS_E_SAVEFILE	Failed to save package file.
0xC001100F	-1073672177	DTS_E_SAVEPACKAGEFILE	Failed to save package file "%1" with error 0x%2!8.X! "%3".
0xC001200D	-1073668083	DTS_E_IDTSNAMENOTSUPPORTED	The object must inherit from IDTSName100 and does not.
0xC0012018	-1073668072	DTS_E_CONFIGFORMATINVALID_PACKAGEDELIMITER	The configuration entry, "%1", has an incorrect format because it does not begin with package delimiter. There was no "\package" delimiter.
0xC0012019	-1073668071	DTS_E_CONFIGFORMATINVALID	The configuration entry "%1" had an incorrect format. This can occur because of a missing delimiter or formatting errors, like an invalid array delimiter.
0xC001201B	-1073668069	DTS_E_CONFIGFILEFAILEDEXPORT	Failure exporting configuration file.
0xC0012021	-1073668063	DTS_E_PROPERTIESCOLLECTIONREADONLY	Properties collection cannot be modified.
0xC0012022	-1073668062	DTS_E_DTRXMLSAVEFAILURE	Unable to save configuration file. The file may be read only.
0xC0012023	-1073668061	DTS_E_FAILPACKAGEONFAILURENA	FailPackageOnFailure property is not applicable to the package container.
0xC0012024	-1073668060	DTS_E_TASKPRODUCTLEVEL	The task "%1" cannot run on installed %2 of Integration Services. It requires %3 or higher.
0xC0012029	-1073668055	DTS_E_UNABLETOSAVETOFILE	Unable to save xml to "%1". The file may be read only.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0012037	-1073668041	DTS_E_CONFIGTYPECONVERSIONFAILED	Failed to convert a type in the configuration "%1" for the package path "%2". This happens when a configuration value cannot be converted from a string to the appropriate destination type. Check the configuration value to ensure it can be converted to the type of the destination property or variable.
0xC0012049	-1073668023	DTS_E_CONFIGFAILED	Configuration failure. This is a generic warning for all configuration types. Other warnings should precede this with more information.
0xC0012050	-1073668016	DTS_E_REMOTEPACKAGEVALIDATION	Package failed validation from the ExecutePackage task. The package cannot run.
0xC0013001	-1073663999	DTS_E_FAILTOCREATEMUTEX	Failed to create mutex "%1" with error 0x%2!8.X!.
0xC0013002	-1073663998	DTS_E_MutexOWNBYDIFFUSER	Mutex "%1" already exists and is owned by another user.
0xC0013003	-1073663997	DTS_E_WAITFORMutexFAILED	Failed to acquire mutex "%1" with error 0x%2!8.X!.
0xC0013004	-1073663996	DTS_E_FAILTORELEASEMUTEX	Failed to release mutex "%1" with error 0x%2!8.X!.
0xC0014003	-1073659901	DTS_E_INVALIDTASKPOINTER	The wrappers task pointer is not valid. The wrapper has an invalid pointer to a task.
0xC0014004	-1073659900	DTS_E_ALREADYADDED	The executable has been added to the Executables collection of another container. This occurs when a client tries to add an executable to more than one Executables collection. You need to remove the executable from the current Executables collection before attempting to add it.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0014005	-1073659899	DTS_E_UNKNOWNCONNECTIONMANAGERTYPE	The connection type "%1" specified for connection manager "%2" is not recognized as a valid connection manager type. This error is returned when an attempt is made to create a connection manager for an unknown connection type. Check the spelling in the connection type name.
0xC0014006	-1073659898	DTS_E_COLLECTIONCOULDNTADD	An object was created but the attempt to add it to a collection failed. This can occur due to an out-of-memory condition.
0xC0014007	-1073659897	DTS_E_ODBCERRORENV	There was an error creating an Open Database Connectivity (ODBC) environment.
0xC0014008	-1073659896	DTS_E_ODBCERRORDBC	There was an error creating an Open Database Connectivity (ODBC) database connection.
0xC0014009	-1073659895	DTS_E_ODBCERRORCONNECTION	There was an error trying to establish an Open Database Connectivity (ODBC) connection with the database server.
0xC001400A	-1073659894	DTS_E_CONNECTIONMANAGERQUALIFIERALREADYSET	The qualifier is already set on this instance of the connection manager. The qualifier may be set once per instance.
0xC001400B	-1073659893	DTS_E_CONNECTIONMANAGERQUALIFIERNOTSET	The qualifier has not been set on this instance of the connection manager. Setting the qualifier is required to complete initialization.
0xC001400C	-1073659892	DTS_E_CONNECTIONMANAGERQUALIFIERNOTSUPPORTED	This connection manager does not support specification of qualifiers.
0xC001400D	-1073659891	DTS_E_CANNOTCLONECONNECTIONMANAGER	Connection manager "0x%1" cannot be cloned for out-of-process execution.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC001400E	-1073659890	DTS_E_NOSQLPROFILERDLL	The log provider for SQL Server Profiler was unable to load pfcInt.dll. Please check that SQL Server Profiler is installed.
0xC001400F	-1073659889	DTS_E_LOGFAILED	The SSIS logging infrastructure failed with error code 0x%1!8.8X!. This error indicates that this logging error is not attributable to a specific log provider.
0xC0014010	-1073659888	DTS_E_LOGPROVIDERFAILED	The SSIS logging provider "%1" failed with error code 0x%2!8.8X! (%3). This indicates a logging error attributable to the specified log provider.
0xC0014011	-1073659887	DTS_E_SAVETOSQLSERVER_OLEDB	The SaveToSQLServer method has encountered OLE DB error code 0x%1!8.8X! (%2). The SQL statement that was issued has failed.
0xC0014012	-1073659886	DTS_E_LOADFROMSQLSERVER_OLEDB	The LoadFromSQLServer method has encountered OLE DB error code 0x%1!8.8X! (%2). The SQL statement that was issued has failed.
0xC0014013	-1073659885	DTS_E_REMOVEFROMSQLSERVER_OLEDB	The RemoveFromSQLServer method encountered OLE DB error code 0x%1!8.8X! (%2) The SQL statement that was issued has failed.
0xC0014014	-1073659884	DTS_E_EXISTSONSQLSERVER_OLEDB	The ExistsOnSQLServer method has encountered OLE DB error code 0x%1!8.8X! (%2). The SQL statement issued has failed.
0xC0014015	-1073659883	DTS_E_CONNECTIONSTRING	OLE DB has failed making a database connection when using the supplied connection string.
0xC0014016	-1073659882	DTS_E_FROMEXECISNOTCHILD	When adding a precedence constraint, a From executable was specified that is not a child of this container.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0014017	-1073659881	DTS_E_TOEXECISNOTCHILD	When adding a precedence constraint, the To executable specified is not a child of this container.
0xC0014018	-1073659880	DTS_E_ODBCTRANSACTIONENLIST	There was an error trying enlist an ODBC connection in a transaction. The SQLSetConnectAttr failed to set the SQL_ATTR_ENLIST_IN_DTC attribute.
0xC0014019	-1073659879	DTS_E_CONNECTIONOFFLINE	The connection manager "%1" will not acquire a connection because the package OfflineMode property is TRUE. When the OfflineMode is TRUE, connections cannot be acquired.
0xC001401A	-1073659878	DTS_E_BEGINTRANSACTION	The SSIS Runtime has failed to start the distributed transaction due to error 0x%1!8.X! "%2". The DTC transaction failed to start. This could occur because the MSDTC Service is not running.
0xC001401B	-1073659877	DTS_E_SETQUALIFIERDESIGNTIMEONLY	The SetQualifier method cannot be called on a connection manager during package execution. This method is used at design-time only.
0xC001401C	-1073659876	DTS_E_SQLPERSISTENCEVERSION	Storing or modifying packages in SQL Server requires the SSIS runtime and database to be the same version. Storing packages in earlier versions is not supported.
0xC001401D	-1073659875	DTS_E_CONNECTIONVALIDATIONFAILED	Connection "%1" failed validation.
0xC001401E	-1073659874	DTS_E_INVALIDFILENAMEINCONNECTION	The file name "%1" specified in the connection was not valid.



HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC001401F	-1073659873	DTS_E_MULTIPLEFILESONRETAINEDCONNECTION	Multiple file names cannot be specified on a connection when the Retain property is TRUE. Vertical bars were found on the connection string, meaning multiple file names are being specified and, in addition, the Retain property is TRUE.
0xC0014020	-1073659872	DTS_E_ODBCERROR	An ODBC error %1!d! has occurred.
0xC0014021	-1073659871	DTS_E_PRECEDENCECONSTRAINT	There was an error in the precedence constraint between "%1" and "%2".
0xC0014022	-1073659870	DTS_E_FAILEDPOPNATIVEFEED	Failed to populate the ForEachEnumeratorInfos collection with native ForEachEnumerators with the following error code: %1.
0xC0014023	-1073659869	DTS_E_GETENUMERATOR	The GetEnumerator method of the ForEach Enumerator has failed with error 0x%1!8.X! "%2". This occurs when the ForEach Enumerator cannot enumerate.
0xC0014024	-1073659868	DTS_E_CANTGETCERTDATA	The raw certificate data cannot be obtained from the supplied certificate object (error: %1). This occurs when CPackage::put_CertificateObject cannot instantiate the ManagedHelper object, when the ManagedHelper object fails, or when the ManagedHelper object returns a malformed array.
0xC0014025	-1073659867	DTS_E_CANTCREATECERTCONTEXT	Failed to create certificate context (error: %1). This occurs in CPackage::put_CertificateObject or CPackage::LoadFromXML when the corresponding CryptoAPI function fails.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0014026	-1073659866	DTS_E_CANTOPENCERTSTORE	Opening MY certificate store failed with error "%1". This occurs in CPackage::LoadUserCertificateByName and CPackage::LoadUserCertificateByHash.
0xC0014027	-1073659865	DTS_E_CANTFINDCERTBYNAME	The certificate specified by name in MY store cannot be found (error: %1). This occurs in CPackage::LoadUserCertificateByName.
0xC0014028	-1073659864	DTS_E_CANTFINDCERTBYHASH	Unable to find the specified certificate by hash in "MY" store (error: %1). Occurs in CPackage::LoadUserCertificateByHash.
0xC0014029	-1073659863	DTS_E_INVALIDCERTHASHFORMAT	The hash value is not a one-dimensional array of bytes (error: %1). This occurs in CPackage::LoadUserCertificateByHash.
0xC001402A	-1073659862	DTS_E_CANTACCESSARRAYDATA	The data in the array cannot be accessed (error: %1). This error can occur wherever GetDataFromSafeArray is called.
0xC001402B	-1073659861	DTS_E_CREATEMANAGEDHELPERFAILED	The SSIS managed helper object failed during creation with error 0x%1!8.8X! "%2". This occurs whenever CoCreateInstance CLSID_DTSMangedHelper fails.
0xC001402C	-1073659860	DTS_E_OLEDBTRANSACTIONENLIST	The SSIS Runtime has failed to enlist the OLE DB connection in a distributed transaction with error 0x%1!8.8X! "%2".
0xC001402D	-1073659859	DTS_E_SIGNPACKAGEFAILED	Package signing failed with error 0x%1!8.8X! "%2". This occurs when the ManagedHelper.SignDocument method fails.
0xC001402E	-1073659858	DTS_E_CHECKENVELOPEFAILED	Failed to check for XML signature envelope in package XML with error 0x%1!8.8X! "%2". This occurs in CPackage::LoadFromXML.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC001402F	-1073659857	DTS_E_GETXMLSOURCEFAILED	Failed to obtain XML source from XML DOM object with error 0x%1!8.8X! "%2". This occurs when IXMLDOMDocument::get_xml fails.
0xC0014030	-1073659856	DTS_E_PACKAGEVERIFICATIONFAILED	The cryptographic signature of the package failed verification due to error 0x%1!8.8X! "%2". This occurs when the signature verification operation fails.
0xC0014031	-1073659855	DTS_E_GETKEYFROMCERTFAILED	Failed to obtain cryptographic key pair associated with the specified certificate with error 0x%1!8.8X! "%2". Verify that you have the key pair for which the certificate was issued. This error usually occurs when trying to sign a document using a certificate for which the person does not have the private key.
0xC0014032	-1073659854	DTS_E_INVALIDSIGNATURE	The digital signature is not valid. The contents of the package have been modified.
0xC0014033	-1073659853	DTS_E_UNTRUSTEDSIGNATURE	The digital signature is valid; however the signer is not trusted and, therefore, authenticity cannot be guaranteed.
0xC0014034	-1073659852	DTS_E_TRANSACTIONENLISTNOTSUPPORTED	The connection does not support enlisting in distributed transaction.
0xC0014035	-1073659851	DTS_E_PACKAGEPROTECT	Failed to apply package protection with error 0x%1!8.8X! "%2". This error occurs when saving to Xml.
0xC0014036	-1073659850	DTS_E_PACKAGEUNPROTECT	Failed to remove package protection with error 0x%1!8.8X! "%2". This occurs in the CPackage::LoadFromXML method.
0xC0014037	-1073659849	DTS_E_PACKAGEPASSWORD	The package is encrypted with a password. The password was not specified, or is not correct.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0014038	-1073659848	DTS_E_DUPLICATECONSTRAINT	A precedence constraint already exists between the specified executables. More than one precedence constraint is not allowed.
0xC0014039	-1073659847	DTS_E_PACKAGELOADFAILED	The package failed to load due to error 0x%1!8.8X! "%2". This occurs when CPackage::LoadFromXML fails.
0xC001403A	-1073659846	DTS_E_PACKAGEOBJECTNOTENVELOPED	Failed to find package object in signed XML envelope with error 0x%1!8.8X! "%2". This occurs when signed XML does not contain a SSIS package, as expected.
0xC001403B	-1073659845	DTS_E_JAGGEDEVENTINFO	The lengths of parameter names, types, and descriptions arrays are not equal. The lengths must be equal. This occurs when the lengths of the arrays are mismatched. There should be one entry per parameter in each array.
0xC001403C	-1073659844	DTS_E_GETPACKAGEINFOS	An OLE DB error 0x%1!8.8X! (%2) occurred while enumerating packages. A SQL statement was issued and failed.
0xC001403D	-1073659843	DTS_E_UNKNOWNLOGPROVIDERTYPE	The log provider type "%1" specified for log provider "%2" is not recognized as a valid log provider type. This error occurs when an attempt is made to create a log provider for unknown log provider type. Verify the spelling in the log provider type name.
0xC001403E	-1073659842	DTS_E_UNKNOWNLOGPROVIDERTYPENOSUBS	The log provider type is not recognized as a valid log provider type. This error occurs when an attempt is made to create a log provider for unknown log provider type. Verify the spelling in the log provider type name.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC001403F	-1073659841	DTS_E_UNKNOWNCONNECTIONMANAGERTYPE	The connection type specified for connection manager is not a valid connection manager type. This error occurs when an attempt is made to create a connection manager for unknown connection type. Verify the spelling of the connection type name.
0xC0014040	-1073659840	DTS_E_PACKAGEREMOVED	An error was encountered when trying to remove the package "%1" from SQL Server.
0xC0014042	-1073659838	DTS_E_FOLDERADDFAILED	An error was encountered when trying to create a folder on SQL Server named "%1" in folder "%2".
0xC0014043	-1073659837	DTS_E_CREATEFOLDERONSQLSERVER_OLEDB	The CreateFolderOnSQLServer method has encountered OLE DB error code 0x%1!8.8X! (%2) The SQL statement issued has failed.
0xC0014044	-1073659836	DTS_E_FOLDERRENAMEFAILED	An error occurred when renaming folder " %1\\%2" to "%1\\%3" on SQL Server.
0xC0014045	-1073659835	DTS_E_RENAMEFOLDERONSQLSERVER_OLEDB	The RenameFolderOnSQLServer method encountered OLE DB error code 0x%1!8.8X! (%2). The SQL statement issued has failed.
0xC0014046	-1073659834	DTS_E_FOLDERDELETEFAILED	Error deleting SQL Server folder "%1".
0xC0014047	-1073659833	DTS_E_REMOVEFOLDERFROMSQLSERVER_OLEDB	The RemoveFolderOnSQLServer method encountered OLE DB error code 0x%1!8.8X! (%2). The SQL statement issued has failed.
0xC0014048	-1073659832	DTS_E_INVALIDPATHTOPACKAGE	The specified package path does not contain a package name. This occurs when the path does not contain at least one backslash or one forward slash.
0xC0014049	-1073659831	DTS_E_FOLDERNOTFOUND	Cannot find folder "%1".

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC001404A	-1073659830	DTS_E_FINDFOLDERONSQL SERVER_OLEDB	While trying to find a folder on SQL an OLE DB error was encountered with error code 0x%1!8.8X! (%2).
0xC001404B	-1073659829	DTS_E_OPENLOGFAILED	The SSIS logging provider has failed to open the log. Error code: 0x%1!8.8X!.
0xC001404C	-1073659828	DTS_E_GETCONNECTIONIN FOS	Failed to get ConnectionInfos collection with error 0x%1!8.8X! "%2". This error occurs when the call to IDTSApplication100::get_ConnectionInfos fails.
0xC001404D	-1073659827	DTS_E_VARIABLEDEADLOCK	Deadlock detected while trying to lock variables. The locks cannot be acquired after 16 attempts. The locks timed out.
0xC001404E	-1073659826	DTS_E_NOTDISPENSED	The Variables collection has not been returned from the VariableDispenser. An operation was attempted that is only allowed on dispensed collections.
0xC001404F	-1073659825	DTS_E_VARIABLESALREADY UNLOCKED	This Variables collection has already been unlocked. The Unlock method is called only once on a dispensed Variables collection.
0xC0014050	-1073659824	DTS_E_VARIABLEUNLOCKF AILED	One or more variables failed to unlock.
0xC0014051	-1073659823	DTS_E_DISPENSEDREADON LY	The Variables collection was returned the from VariableDispenser and cannot be modified. Items cannot be added to or removed from dispensed collections.
0xC0014052	-1073659822	DTS_E_VARIABLEALREADYO NREADLIST	The variable "%1" is already on the read list. A variable may only be added once to either the read lock list or the write lock list.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0014053	-1073659821	DTS_E_VARIABLEALREADYONWRITELIST	The variable "%1" is already on the write list. A variable may only be added once to either the read lock list or the write lock list.
0xC0014054	-1073659820	DTS_E_LOCKVARIABLEFORREAD	Failed to lock variable "%1" for read access with error 0x%2!8.8X! "%3".
0xC0014055	-1073659819	DTS_E_LOCKVARIABLEFORWRITE	Failed to lock variable "%1" for read/write access with error 0x%2!8.8X! "%3".
0xC0014056	-1073659818	DTS_E_CUSTOMEVENTCONFLICT	The custom event "%1" is already declared with a different parameter list. A task is trying to declare a custom event, which another task has already declared with a different parameter list.
0xC0014057	-1073659817	DTS_E_EVENTHANDLERNOTALLOWED	The task providing the custom event "%1" does not allow this event to be handled in the package. The custom event was declared with AllowEventHandlers = FALSE.
0xC0014059	-1073659815	DTS_E_UNSAFEVARIABLESALREADYSET	The VariableDispenser received an unsafe Variables collection. This operation cannot be repeated.
0xC001405A	-1073659814	DTS_E_INVALIDPARENTPACKAGEPATH	GetPackagePath was called on the GetEnumerator but there was no ForEachLoop package path specified.
0xC001405B	-1073659813	DTS_E_VARIABLEDEADLOCK_READ	A deadlock was detected while trying to lock variable "%1" for read access. A lock could not be acquired after 16 attempts and timed out.
0xC001405C	-1073659812	DTS_E_VARIABLEDEADLOCK_READWRITE	A deadlock was detected while trying to lock variables "%1" for read/write access. A lock cannot be acquired after 16 attempts. The locks timed out.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC001405D	-1073659811	DTS_E_VARIABLEDEADLOCK_BOTH	A deadlock was detected while trying to lock variables "%1" for read access and variables "%2" for read/write access. A lock cannot be acquired after 16 attempts. The locks timed out.
0xC001405E	-1073659810	DTS_E_PACKAGEPASSWORD_EMPTY	The protection level of the package requires a password, but PackagePassword property is empty.
0xC001405F	-1073659809	DTS_E_DECRYPTXML_PASSWORD	Failed to decrypt an encrypted XML node because the password was not specified or not correct. Package load will attempt to continue without the encrypted information.
0xC0014060	-1073659808	DTS_E_DECRYPTPACKAGE_USERKEY	Failed to decrypt a package that is encrypted with a user key. You may not be the user who encrypted this package, or you are not using the same machine that was used to save the package.
0xC0014061	-1073659807	DTS_E_SERVERSTORAGE_DISALLOWED	The protection level, ServerStorage, cannot be used when saving to this destination. The system could not verify that the destination supports secure storage capability.
0xC0014062	-1073659806	DTS_E_LOADFROMSQLSERVER	LoadFromSqlServer method has failed.
0xC0014063	-1073659805	DTS_E_SIGNATURE_POLICYVIOLATION	The package cannot be loaded because the state of the digital signature violates signature policy. Error 0x%1!8.X! "%2"
0xC0014064	-1073659804	DTS_E_SIGNATURE_NOTPRESENT	The package is not signed.
0xC0014065	-1073659803	DTS_E_SQLPROFILERDLL_ONLY_X86	The log provider for SQL Server Profiler was unable to load pfcInt.dll because it is only supported on 32-bit systems.



HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0014100	-1073659648	DTS_E_NAMEALREADYADDED	The object cannot be added because another object with the same name already exists in the collection. Use a different name to resolve this error.
0xC0014101	-1073659647	DTS_E_NAMEALREADYEXISTS	The object name cannot be changed from "%1" to "%2" because another object in the collection already uses that name. Use a different name to resolve this error.
0xC0014103	-1073659645	DTS_E_FAILEDDEPENDENCIES	There was an error enumerating the package dependencies. Check other messages for more information.
0xC0014104	-1073659644	DTS_E_INVALIDCHECKPOINT_TRANSACTION	The current package settings are not supported. Please change the SaveCheckpoints property or the TransactionOption property.
0xC001410E	-1073659634	DTS_E_CONNECTIONMANAGERJOINTRANSACTION	The connection manager failed to defect from the transaction.
0xC0015001	-1073655807	DTS_E_BPDUPLICATE	The specified breakpoint ID already exists. This error occurs when a task calls CreateBreakpoint with the same ID multiple times. It is possible to create a breakpoint with the same ID multiple times if the task calls RemoveBreakpoint on the first creation before creating the second one.
0xC0015002	-1073655806	DTS_E_BPUNKNOWNID	The specified breakpoint ID does not exist. This error occurs when a task references a breakpoint that does not exist.
0xC0015004	-1073655804	DTS_E_CANTWRITETOFILE	The file, "%1", could not be opened for writing. The file could be read-only, or you do not have the correct permissions.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0015005	-1073655803	DTS_E_NOROWSETRETURNE D	No result rowset is associated with the execution of this query. The result is not correctly specified.
0xC0015105	-1073655547	DTS_E_DUMP_FAILED	Debug dump files were not generated correctly. The hresult is 0x%1!8.8X!.
0xC0016001	-1073651711	DTS_E_INVALIDURL	The URL specified is not valid. This can happen when the server or proxy URL is null, or in an incorrect format. A valid URL format is in the form of http://ServerName:Port/ResourcePath or https://ServerName:Port/ResourcePath.
0xC0016002	-1073651710	DTS_E_INVALIDSCHEME	The URL %1 is not valid. This can happen when a scheme other than http or https is specified, or the URL is in an incorrect format. A valid URL format is in the form of http://ServerName:Port/ResourcePath or https://ServerName:Port/ResourcePath.
0xC0016003	-1073651709	DTS_E_WINHTTPCANNOTC ONNECT	Connection to server %1 cannot be established. This error can occur when the server does not exist, or the proxy settings are incorrect.
0xC0016004	-1073651708	DTS_E_CONNECTIONTERMI NATED	The connection with the server has been reset or terminated. Try again later.
0xC0016005	-1073651707	DTS_E_LOGINFAILURE	The login attempt failed for "%1". This error occurs when the login credentials provided are incorrect. Verify the login credentials.
0xC0016006	-1073651706	DTS_E_INVALIDSERVERNAM E	The server name specified in the URL %1 cannot be resolved.
0xC0016007	-1073651705	DTS_E_PROXYAUTH	Proxy authentication failed. This error occurs when login credentials are not provided, or the credentials are incorrect.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0016008	-1073651704	DTS_E_SECUREFAILURE	SSL certificate response obtained from the server was not valid. Cannot process the request.
0xC0016009	-1073651703	DTS_E_TIMEOUT	The request has timed out. This error can occur when the timeout specified was too short, or a connection to the server or proxy cannot be established. Ensure that the server and proxy URL are correct.
0xC001600A	-1073651702	DTS_E_CLIENTAUTH	Client certificate is missing. This error occurs when the server is expecting an SSL client certificate and the user has provided an invalid certificate, or has not provided a certificate. A client certificate must be configured for this connection.
0xC001600B	-1073651701	DTS_E_REDIRECTFAILURE	The specified server, URL %1, has a redirect and the redirect request failed.
0xC001600C	-1073651700	DTS_E_SERVERAUTH	Server authentication failed. This error occurs when login credentials are not provided, or the credentials are incorrect.
0xC001600D	-1073651699	DTS_E_WINHTTPUNKNOWNERROR	Request cannot be processed. Try again later.
0xC001600E	-1073651698	DTS_E_UNKNOWNSTATUSCODE	Server returned status code - %1!u! : %2. This error occurs when the server is experiencing problems.
0xC001600F	-1073651697	DTS_E_WINHTTPNOTSUPPORTED	This platform is not supported by WinHttp services.
0xC0016010	-1073651696	DTS_E_INVALIDTIMEOUT	Timeout value is not valid. Timeout should be in the range of %1!d! to %2!d! (in seconds).
0xC0016011	-1073651695	DTS_E_INVALIDCHUNKSIZE	The chunk size is not valid. The ChunkSize property should be in the range of %1!d! to %2!d! (in KB).

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0016012	-1073651694	DTS_E_CERTERROR	Error processing client certificate. This error can occur when the client certificate provided was not found in the Personal Certificate Store. Verify that the client certificate is valid.
0xC0016013	-1073651693	DTS_E_FORBIDDEN	Server returned error code "403 - Forbidden". This error can occur when the specified resource needs "https" access, but the certificate validity period has expired, the certificate is not valid for the use requested, or the certificate has been revoked or revocation can not be checked.
0xC0016014	-1073651692	DTS_E_WINHTTPOPEN	Error initializing HTTP session with proxy "%1". This error can occur when an invalid proxy was specified. HTTP connection manager only supports CERN-type proxies.
0xC0016015	-1073651691	DTS_E_OPENCERTSTORE	Error opening certificate store.
0xC0016016	-1073651690	DTS_E_UNPROTECTXMLFAILE	Failed to decrypt protected XML node "%1" with error 0x%2!8.X! "%3". You may not be authorized to access this information. This error occurs when there is a cryptographic error. Verify that the correct key is available.
0xC0016017	-1073651689	DTS_E_UNPROTECTCONNECTIONSTRINGFAILE	Failed to decrypt protected connection string for server "%1" with error 0x%2!8.X! "%3". You may not be authorized to access this information. This error occurs when there is a cryptographic error. Verify that the correct key is available.
0xC0016018	-1073651688	DTS_E_NEGATIVEVERSION	The version number cannot be negative. This error occurs when the VersionMajor, VersionMinor, or VersionBuild property of the package is set to a negative value.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0016019	-1073651687	DTS_E_PACKAGEMIGRATED	The package has been migrated to a later version during loading. It must be reloaded to complete the process. This is an internal error code.
0xC0016020	-1073651680	DTS_E_PACKAGEMIGRATIONFAILED	Package migration from version %1!d! to version %2!d! failed with error 0x%3!8X! "%4".
0xC0016021	-1073651679	DTS_E_PACKAGEMIGRATIONMODULELOAD	Package migration module has failed to load.
0xC0016022	-1073651678	DTS_E_PACKAGEMIGRATIONMODULE	Package migration module has failed.
0xC0016023	-1073651677	DTS_E_CANTDETERMINEWHICHPROPTOPERSIST	Unable to persist object using default persistence. This error occurs when the default persistence is unable to determine which objects are on the hosted object.
0xC0016024	-1073651676	DTS_E_CANTADDREMOVEWHILEEXECUTING	Cannot add or remove an element from a package in runtime mode. This error occurs when an attempt is made to add or remove an object from a collection while the package is executing.
0xC0016025	-1073651675	DTS_E_NODENOTFOUND	The "%1" node cannot be found in custom default persistence. This error occurs if the default saved XML of an extensible object was changed in a way that a saved object is no longer found, or if the extensible object itself changed.
0xC0016026	-1073651674	DTS_E_COLLECTIONLOCKED	This collection cannot be modified during package validation or execution.
0xC0016027	-1073651673	DTS_E_COLLOCKED	The "%1" collection cannot be modified during package validation or execution. "%2" cannot be added to the collection.
0xC0016029	-1073651671	DTS_E_FTPNOTCONNECTED	Connection with the FTP server has not been established.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC001602A	-1073651670	DTS_E_FTPERROR	An error occurred in the requested FTP operation. Detailed error description: %1.
0xC001602B	-1073651669	DTS_E_FTPINVALIDRETRIES	The number of retries is not valid. The number of retries should be between %1!d! and %2!d!.
0xC001602C	-1073651668	DTS_E_LOADWININET	The FTP connection manager needs the following DLL to function: %1.
0xC001602D	-1073651667	DTS_E_FTPINVALIDCONNECTIONSTRING	The port specified in the connection string is not valid. The ConnectionString format is ServerName:Port. Port should be an integer value between %1!d! and %2!d!.
0xC001602E	-1073651666	DTS_E_FTPCREATEFOLDER	Creating folder "%1" ... %2.
0xC001602F	-1073651665	DTS_E_FTPDELETEFOLDER	Deleting folder "%1" ... %2.
0xC0016030	-1073651664	DTS_E_FTPCHANGEFOLDER	Changing current directory to "%1". %2.
0xC0016031	-1073651663	DTS_E_FTPFILEEMPTY	No files to transfer. This error can occur when performing a Send or Receive operation and no files are specified for the transfer.
0xC0016032	-1073651662	DTS_E_FTPINVALIDLOCALPATH	Specified local path is not valid. Specify a valid local path. This can occur when the specified local path is null.
0xC0016033	-1073651661	DTS_E_FTPNOFILESTODELETE	No files specified to delete.
0xC0016034	-1073651660	DTS_E_WINHTTPCERTDECODE	Internal error occurred while loading the certificate. This error could occur when the certificate data is invalid.
0xC0016035	-1073651659	DTS_E_WINHTTPCERTENCODE	Internal error occurred while saving the certificate data.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0016049	-1073651639	DTS_E_CHECKPOINTMISMATCH	Checkpoint file "%1" does not match this package. The ID of the package and the ID in the checkpoint file do not match.
0xC001604A	-1073651638	DTS_E_CHECKPOINTFILEALREADY EXISTS	An existing checkpoint file is found with contents that do not appear to be for this package, so the file cannot be overwritten to start saving new checkpoints. Remove the existing checkpoint file and try again. This error occurs when a checkpoint file exists, the package is set to not use a checkpoint file, but to save checkpoints. The existing checkpoint file will not be overwritten.
0xC001604B	-1073651637	DTS_E_CHECKPOINTFILELOCKED	The checkpoint file "%1" is locked by another process. This may occur if another instance of this package is currently executing.
0xC001604C	-1073651636	DTS_E_OPENCHECKPOINTFILE	Checkpoint file "%1" failed to open due to error 0x%2!8.X! "%3".
0xC001604D	-1073651635	DTS_E_CREATECHECKPOINTFILE	Checkpoint file "%1" failed during creation due to error 0x%2!8.X! "%3".
0xC0016050	-1073651632	DTS_E_FTPINVALIDPORT	The FTP Port contains an invalid value. The FTP Port value should be an integer between %1!d! and %2!d!.
0xC00160AA	-1073651542	DTS_E_CONNECTTOSERVER FAILED	Connect to SSIS Service on machine "%1" failed:  %2.
0xC0017002	-1073647614	DTS_E_PROPERTYEXPRESSIONDISABLEDONVARIABLES	The Expression property is not supported on Variable objects. Use the EvaluateAsExpression property instead.
0xC0017003	-1073647613	DTS_E_PROPERTYEXPRESSIONEVAL	The expression "%1" on property "%2" cannot be evaluated. Modify the expression to be valid.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0017004	-1073647612	DTS_E_PROPERTYEXPRESSIONSET	The result of the expression "%1" on property "%2" cannot be written to the property. The expression was evaluated, but cannot be set on the property.
0xC0017005	-1073647611	DTS_E_FORLOOPEVALEXPRESSIONINVALID	The evaluation expression for the loop is not valid. The expression needs to be modified. There should be additional error messages.
0xC0017006	-1073647610	DTS_E_EXPRESSIONNOTBOOLEAN	The expression "%1" must evaluate to True or False. Change the expression to evaluate to a Boolean value.
0xC0017007	-1073647609	DTS_E_FORLOOPHASNOEXPRESSION	There is no expression for the loop to evaluate. This error occurs when the expression on the For Loop is empty. Add an expression.
0xC0017008	-1073647608	DTS_E_FORLOOPASSIGNEXPRESSIONINVALID	The assignment expression for the loop is not valid and needs to be modified. There should be additional error messages.
0xC0017009	-1073647607	DTS_E_FORLOOPINITEXPRESSIONINVALID	The initialization expression for the loop is not valid and needs to be modified. There should be additional error messages.
0xC001700A	-1073647606	DTS_E_INVALIDVERSIONNUMBER	The version number in the package is not valid. The version number cannot be greater than current version number.
0xC001700C	-1073647604	DTS_E_INVALIDVERSIONCANTBENEGATIVE	The version number in the package is not valid. The version number is negative.
0xC001700D	-1073647603	DTS_E_PACKAGEUPDATEDISABLED	The package has an older format version, but automatic package format upgrading is disabled.
0xC001700E	-1073647602	DTS_E_EXPREVALTRUNCATIONASERROR	A truncation occurred during evaluation of the expression.



HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0019001	-1073639423	DTS_E_FAILEDSETEXECVALV ARIABLE	The wrapper was unable to set the value of the variable specified in the ExecutionValueVariable property.
0xC0019004	-1073639420	DTS_E_VARIABLEEXPRESSION ERROR	The expression for variable "%1" failed evaluation. There was an error in the expression.
0xC0019305	-1073638651	DTS_E_UNSUPPORTEDSQLV ERSION	The attempted operation is not supported with this database version.
0xC001A003	-1073635325	DTS_E_TXNSPECINVALID	Transaction cannot be specified when a retained connection is used. This error occurs when Retain is set to TRUE on the connection manager, but AcquireConnection was called with a non-null transaction parameter.
0xC001A004	-1073635324	DTS_E_INCOMPATIBLETRAN SACTIONCONTEXT	Incompatible transaction context was specified for a retained connection. This connection has been established under a different transaction context. Retained connections can be used under exactly one transaction context.
0xC001B001	-1073631231	DTS_E_NOTSUSPENDED	Resume call failed because the package is not suspended. This occurs when the client calls resume, but the package is not suspended.
0xC001B002	-1073631230	DTS_E_ALREADYEXECUTING	Execute call failed because the executable is already executing. This error occurs when the client calls Execute on a container that is still executing from the last Execute call.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC001B003	-1073631229	DTS_E_NOTEXECUTING	Suspend or Resume call failed because the executable is not executing, or is not the top-level executable. This occurs when the client calls Suspend or Resume on an executable that is not currently processing an Execute call.
0xC001C002	-1073627134	DTS_E_INVALIDFILE	The file specified in the For Each File enumerator is not valid. Check that the file specified in the For Each File enumerator exists.
0xC001C010	-1073627120	DTS_E_VALUEINDEXNOTINTEGER	The value index is not an integer . Mapping a For Each Variable number %1!d! to the variable "%2".
0xC001C011	-1073627119	DTS_E_VALUEINDEXNEGATIVE	The value index is negative. The ForEach Variable Mapping number %1!d! to variable "%2".
0xC001C012	-1073627118	DTS_E_FOREACHVARIABLEMAPPING	ForEach Variable Mapping number %1!d! to variable "%2" cannot be applied.
0xC001C013	-1073627117	DTS_E_OBJECTNOTINFOREACHLOOP	Failure when adding an object to a ForEachPropertyMapping that is not a direct child of the ForEachLoop container.
0xC001F001	-1073614847	DTS_E_FAILEDSYSTEMVARIABLEREMOVE	Failed to remove a system variable. This error occurs when removing a variable that is a required variable. Required variables are variables that are created by the runtime for communicating between tasks and the runtime.
0xC001F002	-1073614846	DTS_E_CHANGESYSTEMVARIABLEREADONLYFAILED	Changing the property of a variable failed because it is a system variable. System variables are read-only.
0xC001F003	-1073614845	DTS_E_CHANGESYSTEMVARIABLENAMEFAILED	Changing the name of a variable failed because it is a system variable. System variables are read-only.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC001F004	-1073614844	DTS_E_CHANGESYSTEMVARIABLENAMESPACEFAILED	Changing the namespace of a variable failed because it is a system variable. System variables are read-only.
0xC001F006	-1073614842	DTS_E_EVENTHANDLERNAME_READONLY	Changing the event handler name failed. Event handler names are read-only.
0xC001F008	-1073614840	DTS_E_PATHUNKNOWN	Cannot retrieve path to object. This is a system error.
0xC001F009	-1073614839	DTS_E_RUNTIMEVARIABLETYPECHANGE	The type of the value being assigned to variable "%1" differs from the current variable type. Variables may not change type during execution. Variable types are strict, except for variables of type Object.
0xC001F010	-1073614832	DTS_E_INVALIDSTRING	Invalid characters in string: "%1". This occurs when a string supplied for a property value contains unprintable characters.
0xC001F011	-1073614831	DTS_E_INVALIDOBJECTNAME	SSIS object name is invalid. More specific errors would have been raised explaining the exact naming problem.
0xC001F021	-1073614815	DTS_E_PROPERTY_READONLY	The property "%1" is read only. This occurs when a change to a read-only property is attempted.
0xC001F022	-1073614814	DTS_E_FAILEDGETTYPEINFO	The object does not support type information. This occurs when the runtime attempts to get the type information from an object to populate the Properties collection. The object must support type information.
0xC001F023	-1073614813	DTS_E_FAILEDPROPERTYGET	An error occurred while retrieving the value of property "%1". The error code is 0x%2!8.8X!.
0xC001F024	-1073614812	DTS_E_FAILEDPROPERTYGET_ERRORINFO	An error occurred while retrieving the value of property "%1". The error code is 0x%2!8.8X! "%3".

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC001F025	-1073614811	DTS_E_FAILEDPROPERTYSET	An error occurred while setting the value of property "%1". The error returned is 0x%2!8.X!.
0xC001F026	-1073614810	DTS_E_FAILEDPROPERTYSET_ERRORINFO	An error occurred while setting the value of property "%1". The error returned is 0x%2!8.X! "%3".
0xC001F027	-1073614809	DTS_E_PROPERTYWRITEONLY	The property "%1" is write-only. This error occurs when trying to retrieve the value of a property through a property object, but the property is write-only.
0xC001F028	-1073614808	DTS_E_NODISPATCH	The object does not implement IDispatch. This error occurs when a property object or properties collection attempts to access an IDispatch interface on an object.
0xC001F029	-1073614807	DTS_E_NOCONTAININGTYPELIB	Unable to retrieve the type library of the object. This error occurs when the Properties collection attempts to retrieve the type library for an object through its IDispatch interface.
0xC001F02A	-1073614806	DTS_E_INVALIDTASKMONIKER	Cannot create a task from XML for task "%1!s!", type "%2!s!" due to error 0x%3!8.X! "%4!s!".
0xC001F02C	-1073614804	DTS_E_FAILEDCREATEXMLDOCUMENT	Failed to create an XML document "%1".
0xC001F02D	-1073614803	DTS_E_PMVARPROPTYPESDIFFERENT	An error occurred because there is a property mapping from a variable to a property with a different type. The property type must match the variable type.
0xC001F02E	-1073614802	DTS_E_PMINVALIDPROPMAPTARGET	Attempted to set property mapping to target unsupported object type. This error occurs when passing an unsupported object type to a property mapping.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC001F02F	-1073614801	DTS_E_COULDNOTRESOLVE PACKAGEPATH	Cannot resolve a package path to an object in the package "%1". Verify that the package path is valid.
0xC001F030	-1073614800	DTS_E_PMNODESTPROPERT Y	The destination property for the property map is empty. Set the destination property name.
0xC001F031	-1073614799	DTS_E_INVALIDPROPERTYM APPINGSFOUND	The package failed to restore at least one property mapping.
0xC001F032	-1073614798	DTS_E_AMBIGUOUSVARIAB LENAME	The variable name is ambiguous because multiple variables with this name exist in different namespaces. Specify namespace-qualified name to prevent ambiguity.
0xC001F033	-1073614797	DTS_E_DESTINATIONOBJECT PARENTLESS	The destination object in a property mapping has no parent. The destination object is not a child of any sequence container. It may have been removed from the package.
0xC001F036	-1073614794	DTS_E_INVALIDPROPERTYM APPING	The property mapping is not valid. The mapping is ignored.
0xC001F038	-1073614792	DTS_E_PMFATALALERTREMO VE	Failure when alerting property mappings that a target is being removed.
0xC001F03A	-1073614790	DTS_E_INVALIDFOREACHPR OPERTYMAPPING	An invalid property mapping is found on the For Each Loop. This occurs when the ForEach property mapping fails to restore.
0xC001F040	-1073614784	DTS_E_PMPROPERTYINVALI D	A destination property was specified on a property mapping that is invalid. This occurs when a property is specified on a destination object that is not found on that object.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC001F041	-1073614783	DTS_E_INVALIDTASKMONIKERNOPARAM	Cannot create a task from XML. This occurs when the runtime is unable to resolve the name to create a task. Verify that the name is correct.
0xC001F080	-1073614720	DTS_E_COULDNOTREPLACECHECKPOINTFILE	Cannot replace the existing checkpoint file with the updated checkpoint file. The checkpoint was successfully created in a temporary file, but overwriting the existing file with the new file failed.
0xC001F081	-1073614719	DTS_E_CHECKPOINTFILENOTSPECIFIED	The package is configured to always restart from a checkpoint, but checkpoint file is not specified.
0xC001F082	-1073614718	DTS_E_CHECKPOINTLOADXML	The attempt to load the XML checkpoint file "%1" failed with error 0x%2!8.8X! "%3". Check that the file name specified is correct, and that the file exists.
0xC001F083	-1073614717	DTS_E_LOADCHECKPOINT	The package failed during execution because the checkpoint file cannot be loaded. Further execution of the package requires a checkpoint file. This error usually occurs when the CheckpointUsage property is set to ALWAYS, which specifies that the package always restarts.
0xC001F185	-1073614459	DTS_E_NOEVALEXPRESSION	The evaluation condition expression on the For Loop "%1" is empty. There must be a Boolean evaluation expression in the For Loop.
0xC001F186	-1073614458	DTS_E_EXPREVALASSIGNMENTTYPENISMATCH	The result of the assignment expression "%1" cannot be converted to a type that is compatible with the variable that it was assigned to.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC001F187	-1073614457	DTS_E_EXPREVALASSIGNMENTTOREADONLYVARIABLE	Error using a read-only variable "%1" in an assignment expression. The expression result cannot be assigned to the variable because the variable is read only. Choose a variable that can be written to, or remove the expression from this variable.
0xC001F188	-1073614456	DTS_E_EXPREVALASSIGNMENTVARIABLELOCKFORWRITEFAILED	Cannot evaluate expression "%1" because the variable "%2" does not exist or cannot be accessed for writing. The expression result cannot be assigned to the variable because the variable was not found, or could not be locked for write access.
0xC001F189	-1073614455	DTS_E_EXPREVALRESULTTYPEENOTSUPPORTED	The expression "%1" has a result type of "%2", which cannot be converted to a supported type.
0xC001F18A	-1073614454	DTS_E_EXPREVALRESULTTYPECONVERSIONFAILED	The conversion of the result of the expression "%1" from type "%2" to a supported type failed with error code 0x%3!8.X!. An unexpected error occurred when trying to convert the expression result to a type supported by the runtime engine, even though the type conversion is supported.
0xC001F200	-1073614336	DTS_E_DTSNAME_NOTNULL	The object name is not valid. The name cannot be set to NULL.
0xC001F201	-1073614335	DTS_E_DTSNAME_NOTEMPTY	The object name is not valid. The name cannot be empty.
0xC001F202	-1073614334	DTS_E_DTSNAME_LEGAL	The object name "%1" is not valid. The name cannot contain any of the following characters: / \ : [ ] . =
0xC001F203	-1073614333	DTS_E_DTSNAME_PRINTABLE	Object name "%1" is not valid. The name cannot contain control characters that render it unprintable.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC001F204	-1073614332	DTS_E_DTSNAME_NOLEAD WHITESP	Object name "%1" is not valid. Name cannot begin with a whitespace.
0xC001F205	-1073614331	DTS_E_DTSNAME_NOTRAIL WHITESP	Object name "%1" is not valid. Name cannot end with a whitespace.
0xC001F206	-1073614330	DTS_E_DTSNAME_BEGINSWI THALPHA	Object name "%1" is not valid. Name must begin with an alphabetical character.
0xC001F207	-1073614329	DTS_E_DTSNAME_BEGINSWI THALPHAUNDERBAR	Object name "%1" is not valid. Name must begin with an alphabetical character or underscore "_".
0xC001F208	-1073614328	DTS_E_DTSNAME_ALPHADI GITUNDERBAR	Object name "%1" is not valid. Name must contain only alphanumeric characters or underscores "_".
0xC001F209	-1073614327	DTS_E_DTSNAME_VALIDFILE NAME	Object name "%1" is not valid. The name cannot contain any of the following characters: / \ : ? " < >
0xC001F420	-1073613792	DTS_E_FAILLOADINGPROPE RTY	Failed to load the value property "%1" using default persistence.
0xC001F422	-1073613790	DTS_E_NODELISTENUM_INV ALIDCONNMGRTYPE	Connection manager "%1" is not of type "%2"
0xC001F423	-1073613789	DTS_E_NODELISTENUM_XP ATHISEMPTY	"%1" is empty
0xC001F424	-1073613788	DTS_E_NODELISTENUM_INV ALIDDATANODE	Invalid data node in the nodelist enumerator section
0xC001F425	-1073613787	DTS_E_NODELISTENUM_NO ENUMERATORCREATED	No enumerator can be created
0xC001F427	-1073613785	DTS_E_OPERATIONFAILCAC HEINUSE	The operation failed because the cache is in use.
0xC001F428	-1073613784	DTS_E_PROPERTYCANNOTB EMODIFIED	The property cannot be modified.
0xC001F429	-1073613783	DTS_E_PACKAGEUPGRADEF AILED	The package upgrade has failed.



HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC00220DE	-1073602338	DTS_E_TKEXECPACKAGE_UNABLETOLOADFILE	Error 0x%1!8.8X! while loading package file "%3". %2.
0xC00220DF	-1073602337	DTS_E_TKEXECPACKAGE_UNSPECIFIEDPACKAGE	The package is not specified.
0xC00220E0	-1073602336	DTS_E_TKEXECPACKAGE_UNSPECIFIEDCONNECTION	The connection is not specified.
0xC00220E2	-1073602334	DTS_E_TKEXECPACKAGE_INCORRECTCONNECTIONMANAGERTYPE	The connection manager "%1" has an unsupported type "%2". Only "FILE" and "OLEDB" connection managers are supported.
0xC00220E3	-1073602333	DTS_E_TKEXECPACKAGE_UNABLETOLOADXML	Error 0x%1!8.8X! while loading package file "%3" into an XML document. %2.
0xC00220E4	-1073602332	DTS_E_TKEXECPACKAGE_UNABLETOLOAD	Error 0x%1!8.8X! while preparing to load the package. %2.
0xC0024102	-1073594110	DTS_E_TASKVALIDATIONFAILED	The Validate method on the task failed, and returned error code 0x%1!8.8X! (%2). The Validate method must succeed and indicate the result using an "out" parameter.
0xC0024104	-1073594108	DTS_E_TASKEXECUTEFAILED	The Execute method on the task returned error code 0x%1!8.8X! (%2). The Execute method must succeed, and indicate the result using an "out" parameter.
0xC0024105	-1073594107	DTS_E_RETRIEVINGDEPENDENCIES	A failure occurred on task "%1": 0x%2!8.8X! while retrieving dependencies. The runtime was retrieving dependencies from the task's dependencies collection when the error occurred. The task may have incorrectly implemented one of the dependency interfaces.
0xC0024107	-1073594105	DTS_E_TASKVALIDATIONERROR	There were errors during task validation.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0024108	-1073594104	DTS_E_CONNECTIONSTRINGFORMAT	The connection string format is not valid. It must consist of one or more components of the form X=Y, separated by semicolons. This error occurs when a connection string with zero components is set on database connection manager.
0xC0024109	-1073594103	DTS_E_UNQUOTEDSEMICOLON	The connection string components cannot contain unquoted semicolons. If the value must contain a semicolon, enclose the entire value in quotes. This error occurs when values in the connection string contain unquoted semicolons, such as the InitialCatalog property.
0xC002410A	-1073594102	DTS_E_LOGPROVIDERVALIDATIONFAILED	Validation of one or more log providers failed. The package cannot execute. The package does not execute when a log provider fails validation.
0xC002410B	-1073594101	DTS_E_INVALIDVALUEINARRAY	Invalid value in array.
0xC002410C	-1073594100	DTS_E_ENUMERATIONELEMENTNOTENUMERABLE	An element of the enumerator returned by the ForEach Enumerator does not implement IEnumerator, contradicting the CollectionEnumerator property of the ForEach Enumerator.
0xC002410D	-1073594099	DTS_E_INVALIDENUMERATORINDEX	The enumerator failed to retrieve element at index "%1!d!".
0xC0029100	-1073573632	DTS_E_AXTASK_MISSING_ENTRY_METHOD_NAME	Function not found.
0xC0029101	-1073573631	DTS_E_AXTASK_EMPTY_SCRIPT	Function not found.
0xC0029102	-1073573630	DTS_E_AXTASK_INITIALIZATION_WITH_WRONG_XML_ELEMENT	ActiveX Script Task was initiated with a wrong XML element.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0029105	-1073573627	DTS_E_AXTASK_HANDLER_NOT_FOUND	Handler not found.
0xC0029106	-1073573626	DTS_E_AXTASKUTIL_ENUMERATE_LANGUAGES_FAILED	An error occurred while attempting to retrieve the scripting languages installed on the system.
0xC0029107	-1073573625	DTS_E_AXTASKUTIL_SCRIPT_HOST_CREATE_FAILED	An error occurred while creating the ActiveX script host. Verify that you have the script host installed properly.
0xC0029108	-1073573624	DTS_E_AXTASKUTIL_SCRIPT_HOSTINIT_FAILED	An error occurred while trying to instantiate the script host for the chosen language. Verify that the script language you have chosen is installed on your system.
0xC0029109	-1073573623	DTS_E_AXTASKUTIL_ADDVARIABLES_FAILED	An error occurred while adding the SSIS variables to the script host namespace. This might prevent the task from using SSIS variables in the script.
0xC002910A	-1073573622	DTS_E_AXTASKUTIL_SCRIPT_PARSING_FAILED	A fatal error occurred while trying to parse the script text. Verify that the script engine for the chosen language is installed properly.
0xC002910B	-1073573621	DTS_E_AXTASKUTIL_MSG_BAD_FUNCTION	The function name entered is not valid. Verify that a valid function name has been specified.
0xC002910C	-1073573620	DTS_E_AXTASKUTIL_EXECUTION_FAILED	An error occurred while executing the script. Verify that the script engine for the selected language is installed properly.
0xC002910D	-1073573619	DTS_E_AXTASKUTIL_ADDTYPELIB_FAILED	An error occurred while adding the managed type library to the script host. Verify that the DTS 2000 runtime is installed.
0xC002910E	-1073573618	DTS_E_BITASK_INITIALIZATION_WITH_WRONG_XML_ELEMENT	Bulk Insert Task was initiated with a wrong XML element.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC002910F	-1073573617	DTS_E_BITASK_DATA_FILE_NOT_SPECIFIED	Data file name not specified.
0xC0029110	-1073573616	DTS_E_BITASK_HANDLER_NOT_FOUND	Handler not found.
0xC0029111	-1073573615	DTS_E_BITASK_CANNOT_ACQUIRE_CONNECTION	Failed to acquire the specified connection: "%1".
0xC0029112	-1073573614	DTS_E_BITASK_NO_CONNECTION_MANAGER_SPECIFIED	Attempt to obtain the Connection Manager failed.
0xC0029113	-1073573613	DTS_E_BITASK_INVALID_CONNECTION	The connection is not valid.
0xC0029114	-1073573612	DTS_E_BITASK_NULL_CONNECTION	The connection is null.
0xC0029115	-1073573611	DTS_E_BITASK_EXECUTE_FAILED	Execution failed.
0xC0029116	-1073573610	DTS_E_BITASK_CANNOT_RETRIEVE_TABLES	An error occurred while retrieving the tables from the database.
0xC0029117	-1073573609	DTS_E_BITASK_CANNOT_RETRIEVE_COLUMN_INFO	An error occurred while retrieving the columns of the table.
0xC0029118	-1073573608	DTS_E_BITASK_ERROR_IN_DATABASE_OPERATION	An error occurred in the database operation.
0xC0029119	-1073573607	DTS_E_BITASK_INVALIDSOURCECONNECTIONNAME	The specified connection "%1" is either not valid, or points to an invalid object. To continue, specify a valid connection.
0xC002911A	-1073573606	DTS_E_BITASK_INVALIDDESTINATIONCONNECTIONNAME	The destination connection specified is not valid. Supply a valid connection to continue.
0xC002911B	-1073573605	DTS_E_BITASK_DESTINATION_TABLE_NOT_SPECIFIED	You must specify a table name to continue.
0xC002911C	-1073573604	DTS_E_BITASK_ERROR_IN_LOAD_FROM_XML	Error occurred in LoadFromXML at the tag "%1".
0xC002911D	-1073573603	DTS_E_BITASK_ERROR_IN_SAVE_TO_XML	Error occurred in SaveToXML at the tag "%1".

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC002911E	-1073573602	DTS_E_BITASKUNMANCON NECTION_INVALID_CONNE CTION	The connection is not valid.
0xC002911F	-1073573601	DTS_E_BITASKUNMANCON NECTION_EXECUTE_FAILED	Execution failed.
0xC0029120	-1073573600	DTS_E_BITASKUNMANCON NECTION_CANNOT_RETRIEV E_TABLES	Error occurred while retrieving the tables from the database.
0xC0029121	-1073573599	DTS_E_BITASKUNMANCON NECTION_CANNOT_RETRIEV E_COLUMN_INFO	Error occurred while retrieving the columns of the table.
0xC0029122	-1073573598	DTS_E_BITASKUNMANCON NECTION_CANNOT_OPEN_F ILE	Error occurred while trying to open the data file.
0xC0029123	-1073573597	DTS_E_BITASKUNMANCON NECTION_OEM_CONVERSI ON_FAILED	Cannot convert the input OEM file to the specified format.
0xC0029124	-1073573596	DTS_E_BITASKUNMANCON NECTION_ERROR_IN_DB_OP ERATION	Error in database operation.
0xC0029125	-1073573595	DTS_E_DTSPROCTASK_NOC ONNECTIONSPECIFIED	No connection manager specified.
0xC0029126	-1073573594	DTS_E_DTSPROCTASK_CON NECTIONMANAGERNOTOL AP	Connection "%1" is not an Analysis Services connection.
0xC0029127	-1073573593	DTS_E_DTSPROCTASK_UNA BLETOLOCATECONNECTIO NMANAGER	Unable to locate connection "%1".
0xC0029128	-1073573592	DTS_E_DTSPROCTASK_INVA LIDTASKDATANODEEXE	Analysis Services Execute DDL task received an invalid task data node.
0xC0029129	-1073573591	DTS_E_DTSPROCTASK_INVA LIDTASKDATANODEPROC	Analysis Services Processing task received an invalid task data node.
0xC002912A	-1073573590	DTS_E_DTSPROCTASK_INVA LIDDDL	The DDL is not valid.
0xC002912B	-1073573589	DTS_E_DTSPROCTASK_INVA LIDDDLPROCESSINGCOMM ANDS	The DDL found in ProcessingCommands is not valid.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC002912C	-1073573588	DTS_E_DTSPROCTASK_CAN NOTWRITEINAREADONLYV ARIABLE	The Execution result cannot be saved in a read-only variable.
0xC002912D	-1073573587	DTS_E_DTSPROCTASK_INVA LIDVARIABLE	Variable "%1" it's not defined.
0xC002912E	-1073573586	DTS_E_DTSPROCTASK_CON NECTIONNOTFOUND	Connection Manager "%1" it's not defined.
0xC002912F	-1073573585	DTS_E_DTSPROCTASK_INVA LIDCONNECTION	Connection Manager "%1" it's not a FILE Connection Manager.
0xC0029130	-1073573584	DTS_E_DTSPROCTASK_NON EXISTENTATTRIBUTE	"%1" was not found during deserialization.
0xC0029131	-1073573583	DTS_E_DTSPROCTASK_TRAC EHASBEENSTOPPED	The trace has been stopped due to an exception.
0xC0029132	-1073573582	DTS_E_DTSPROCTASK_DDLE XECUTIONFAILED	Execution of DDL failed.
0xC0029133	-1073573581	DTS_E_DTSPROCTASK_FILED OESNOTEXIST	There is no file associated with connection "%1".
0xC0029134	-1073573580	DTS_E_DTSPROCTASK_VARI ABLENOTDEFINED	Variable "%1" is not defined.
0xC0029135	-1073573579	DTS_E_DTSPROCTASK_FILEC ONNECTIONNOTDEFINED	File connection "%1" is not defined.
0xC0029136	-1073573578	DTS_E_EXEC2000PKGTASK_I NITIALIZATION_WITH_WRO NG_XML_ELEMENT	Execute DTS 2000 Package task is initiated with a wrong XML element.
0xC0029137	-1073573577	DTS_E_EXEC2000PKGTASK_ HANDLER_NOT_FOUND	Handler not found.
0xC0029138	-1073573576	DTS_E_EXEC2000PKGTASK_ PACKAGE_NAME_NOT_SPEC IFIED	Package name is not specified.
0xC0029139	-1073573575	DTS_E_EXEC2000PKGTASK_ PACKAGE_ID_NOT_SPECIFIE D	Package ID is not specified.
0xC002913A	-1073573574	DTS_E_EXEC2000PKGTASK_ PACKAGE_VERSIONGUID_N OT_SPECIFIED	Package version GUID is not specified.
0xC002913B	-1073573573	DTS_E_EXEC2000PKGTASK_S QLSERVER_NOT_SPECIFIED	SQL Server is not specified.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC002913C	-1073573572	DTS_E_EXEC2000PKGTASK_SQL_USERNAME_NOT_SPECIFIED	SQL Server user name not specified.
0xC002913D	-1073573571	DTS_E_EXEC2000PKGTASK_FILE_NAME_NOT_SPECIFIED	Storage file name not specified.
0xC002913E	-1073573570	DTS_E_EXEC2000PKGTASK_DTS2000CANTBEEMPTY	The DTS 2000 package property is empty.
0xC002913F	-1073573569	DTS_E_EXEC2000PKGTASK_ERROR_IN_PACKAGE_EXECUTION	An error occurred while executing the DTS 2000 package.
0xC0029140	-1073573568	DTS_E_EXEC2000PKGTASK_SQLSERVER_NOT_AVAILABLE_NETWORK	Cannot load the available SQL Servers from the network. Check the network connection.
0xC0029141	-1073573567	DTS_E_EXEC2000PKGTASK_DATATYPE_NULL	The data type cannot be null. Please specify the correct data type to use for validating the value.
0xC0029142	-1073573566	DTS_E_EXEC2000PKGTASK_NULL_VALUE	Cannot validate a null against any data type.
0xC0029143	-1073573565	DTS_E_EXEC2000PKGTASK_NULL_VALUE_ARGUMENT	A required argument is null.
0xC0029144	-1073573564	DTS_E_EXEC2000PKGTASK_CLS_NOT_REGISTERED_EXCEPTION	To execute the DTS 2000 Package task, start SQL Server Setup and use the Advanced button from the Components to Install page to select Legacy Components.
0xC0029145	-1073573563	DTS_E_EXEC2000PKGTASK_NOT_PRIMITIVE_TYPE	"%1" is not a value type.
0xC0029146	-1073573562	DTS_E_EXEC2000PKGTASK_CONVERT_FAILED	Could not convert "%1" to "%2".
0xC0029147	-1073573561	DTS_E_EXEC2000PKGTASK_ERROR_IN_VALIDATE	Could not validate "%1" against "%2".
0xC0029148	-1073573560	DTS_E_EXEC2000PKGTASK_ERROR_IN_LOAD_FROM_XML	Error occurred in LoadFromXML at the tag "%1".
0xC0029149	-1073573559	DTS_E_EXEC2000PKGTASK_ERROR_IN_SAVE_TO_XML	Error occurred in SaveToXML at the tag "%1".

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC002914A	-1073573558	DTS_E_EXECPROCTASK_INV ALIDTIMEOUT	The time-out value provided is not valid. Specify the number of seconds that the task allows the process to run. The minimum time-out is 0, which indicates that no time-out value is used and the process runs to completion or until an error occurs. The maximum time-out is 2147483 (((2^31) - 1)/1000).
0xC002914B	-1073573557	DTS_E_EXECPROCTASK_CAN TREDIRECTIO	Cannot redirect streams if the process can continue executing beyond the lifetime of the task.
0xC002914C	-1073573556	DTS_E_EXECPROCTASK_PRO CESSHASTIMEDOUT	The process timed out.
0xC002914D	-1073573555	DTS_E_EXECPROCTASK_EXE CUTABLENOTSPECIFIED	The executable is not specified.
0xC002914E	-1073573554	DTS_E_EXECPROCTASK_STD OUTVARREADONLY	The standard out variable is read-only.
0xC002914F	-1073573553	DTS_E_EXECPROCTASK_STD ERRVARREADONLY	The standard error variable is read-only.
0xC0029150	-1073573552	DTS_E_EXECPROCTASK_REC EIVEDINVALIDTASKDATAN ODE	The Execute Process task received a task data node that is not valid.
0xC0029151	-1073573551	DTS_E_EXECPROCTASK_PRO CESSEXITCODEEXCEEDS	In Executing "%2" "%3" at "%1", The process exit code was "%4" while the expected was "%5".
0xC0029152	-1073573550	DTS_E_EXECPROCTASK_WO RKINGDIRDOESNOTEXIST	The directory "%1" does not exist.
0xC0029153	-1073573549	DTS_E_EXECPROCTASK_FILE DOESNOTEXIST	File/Process "%1" does not exist in directory "%2".
0xC0029154	-1073573548	DTS_E_EXECPROCTASK_FILE NOTINPATH	File/Process "%1" is not in path.
0xC0029156	-1073573546	DTS_E_EXECPROCTASK_WO RKINGDIRECTORYDOESNOT EXIST	Working Directory "%1" does not exist.
0xC0029157	-1073573545	DTS_E_EXECPROCTASK_ERR OREXECUTIONVALUE	The process exited with return code "%1". However, "%2" was expected.



HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0029158	-1073573544	DTS_E_FSTASK_SYNCFAILED	Synchronization object failed.
0xC0029159	-1073573543	DTS_E_FSTASK_INVALIDDATA	The File System task received an invalid task data node.
0xC002915A	-1073573542	DTS_E_FSTASK_DIRECTORYEXISTS	The Directory already exists.
0xC002915B	-1073573541	DTS_E_FSTASK_PATHNOTVALID	"%1" is not valid on operation type "%2".
0xC002915C	-1073573540	DTS_E_FSTASK_DESTINATIONNOTSET	Destination property of operation "%1" not set.
0xC002915D	-1073573539	DTS_E_FSTASK_SOURCENOTSET	Source property of operation "%1" not set.
0xC002915E	-1073573538	DTS_E_FSTASK_CONNECTIONNOTFILE	Type of Connection "%1" is not a file.
0xC002915F	-1073573537	DTS_E_FSTASK_VARIABLEDOESNOTEXIST	Variable "%1" does not exist.
0xC0029160	-1073573536	DTS_E_FSTASK_VARIABLENOTASTRING	Variable "%1" is not a string.
0xC0029163	-1073573533	DTS_E_FSTASK_FILEDOESNOTEXIST	File or directory "%1" represented by connection "%2" does not exist.
0xC0029165	-1073573531	DTS_E_FSTASK_DESTINATIONUSAGEINVALID	The destination file connection manager "%1" has an invalid usage type: "%2".
0xC0029166	-1073573530	DTS_E_FSTASK_SOURCEUSAGEINVALID	The source file connection manager "%1" has an invalid usage type "%2".
0xC0029167	-1073573529	DTS_E_FSTASK_LOGENTRYGETTINGFILEOPERATION	FileSystemOperation
0xC0029168	-1073573528	DTS_E_FSTASK_LOGENTRYGETTINGFILEOPERATIONDESCRIPTION	Provides information regarding File System operations.
0xC0029169	-1073573527	DTS_E_FSTASK_TASKDISPLAYNAME	File System Task
0xC002916A	-1073573526	DTS_E_FSTASK_TASKDESCRIPTION	Perform file system operations, such as copying and deleting files.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC002916B	-1073573525	DTS_E_FTPTASK_SYNCOBJFAILED	Synchronization object failed.
0xC002916C	-1073573524	DTS_E_FTPTASK_UNABLETOOBTAINFILELIST	Unable to obtain the file list.
0xC002916D	-1073573523	DTS_E_FTPTASK_LOCALPATHEMPTY	The local path is empty.
0xC002916E	-1073573522	DTS_E_FTPTASK_REMOTEPATHEMPTY	The remote path is empty.
0xC002916F	-1073573521	DTS_E_FTPTASK_LOCALVARIABLEEMPTY	The local variable is empty.
0xC0029170	-1073573520	DTS_E_FTPTASK_REMOTEVARIABLEEMPTY	The remote variable is empty.
0xC0029171	-1073573519	DTS_E_FTPTASK_FTPRCVDINVALIDTASKDATANODE	The FTP task received an invalid task data node.
0xC0029172	-1073573518	DTS_E_FTPTASK_CONNECTION_NAME_NULL	The connection is empty. Verify that a valid FTP connection is provided.
0xC0029173	-1073573517	DTS_E_FTPTASK_CONNECTION_NOT_FTP	The connection specified is not an FTP connection. Verify that a valid FTP connection is provided.
0xC0029175	-1073573515	DTS_E_FTPTASK_INITIALIZATION_WITH_NULL_XML_ELEMENT	Cannot initialize the task with a null XML element.
0xC0029176	-1073573514	DTS_E_FTPTASK_SAVE_TO_NULL_XML_ELEMENT	Cannot save the task to a null XML document.
0xC0029177	-1073573513	DTS_E_FTPTASK_ERROR_IN_LOAD_FROM_XML	Error occurred in LoadFromXML at the tag "%1".
0xC0029178	-1073573512	DTS_E_FTPTASK_NOFILESATLOCATION	There are no files at "%1".
0xC0029179	-1073573511	DTS_E_FTPTASK_LOCALVARIABLEISEMPTY	The variable "%1" is empty.
0xC002917A	-1073573510	DTS_E_FTPTASK_REMOTEVARIABLEISEMPTY	The variable "%1" is empty.
0xC002917B	-1073573509	DTS_E_FTPTASK_NOFILESINCONNMGR	The File "%1" doesn't contain file path(s).

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC002917C	-1073573508	DTS_E_FTPTASK_NOFILEPAT HSINLOCALVAR	The variable "%1" doesn't contain file path(s).
0xC002917D	-1073573507	DTS_E_FTPTASK_VARIABLEN OTASTRING	Variable "%1" is not a string.
0xC002917E	-1073573506	DTS_E_FTPTASK_VARIABLEN OTFOUND	Variable "%1" does not exist.
0xC002917F	-1073573505	DTS_E_FTPTASK_INVALIDPA THONOPERATION	Invalid path on operation "%1".
0xC0029180	-1073573504	DTS_E_FTPTASK_DIRECTORY EXISTS	"%1" already exists.
0xC0029182	-1073573502	DTS_E_FTPTASK_CONNECTI ONTYPENOTFILE	Type of Connection "%1" is Not a file.
0xC0029183	-1073573501	DTS_E_FTPTASK_FILEDOESN OTEXIST	File represented by "%1" does not exist.
0xC0029184	-1073573500	DTS_E_FTPTASK_INVALIDDDI RECTORY	Directory is not specified in the variable "%1".
0xC0029185	-1073573499	DTS_E_FTPTASK_NOFILESFO UND	No files found in "%1".
0xC0029186	-1073573498	DTS_E_FTPTASK_NODIRECT ORYPATHINCONMGR	Directory is not specified in the file connection manager "%1".
0xC0029187	-1073573497	DTS_E_FTPTASK_UNABLETO DELETELOCALEFILE	Unable to delete local file "%1".
0xC0029188	-1073573496	DTS_E_FTPTASK_UNABLETO REMOVELOCALDIRECTORY	Unable to remove local directory "%1".
0xC0029189	-1073573495	DTS_E_FTPTASK_UNABLETO CREATELOCALDIRECTORY	Unable to create local directory "%1".
0xC002918A	-1073573494	DTS_E_FTPTASK_UNABLETO RECEIVEFILES	Unable to receive files using "%1".
0xC002918B	-1073573493	DTS_E_FTPTASK_UNABLETO SENDFILES	Unable to send files using "%1".
0xC002918C	-1073573492	DTS_E_FTPTASK_UNABLETO MAKEDIRREMOTE	Unable to create remote directory using "%1".
0xC002918D	-1073573491	DTS_E_FTPTASK_UNABLETO REMOVEDIRREMOTE	Unable to remove remote directory using "%1".
0xC002918E	-1073573490	DTS_E_FTPTASK_UNABLETO DELETEREMOTEFILES	Unable to delete remote files using "%1".

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC002918F	-1073573489	DTS_E_FTPTASK_UNABLETO CONNECTTOSERVER	Unable to connect to FTP server using "%1".
0xC0029190	-1073573488	DTS_E_FTPTASK_INVALIDDVA RIABLEVALUE	Variable "%1" doesn't start with "/".
0xC0029191	-1073573487	DTS_E_FTPTASK_INVALIDDRE MOTEPATH	Remote path "%1" doesn't start with "/".
0xC0029192	-1073573486	DTS_E_DTS_E_FTPTASK_CAN NOT_ACQUIRE_CONNECTIO N	There was an error acquiring the FTP connection. Please check if you have specified a valid connection type "%1".
0xC0029193	-1073573485	DTS_E_MSGQTASKUTIL_CER T_OPEN_STORE_FAILED	Opening the certificate store failed.
0xC0029194	-1073573484	DTS_E_MSGQTASKUTIL_CER T_FAILED_GETTING_DISPLA Y_NAME	An error occurred while retrieving the display name of the certificate.
0xC0029195	-1073573483	DTS_E_MSGQTASKUTIL_CER T_FAILED_GETTING_ISSUER_ NAME	An error occurred while retrieving the issuer name of the certificate.
0xC0029196	-1073573482	DTS_E_MSGQTASKUTIL_CER T_FAILED_GETTING_FRIEND LY_NAME	An error occurred while retrieving the friendly name of the certificate.
0xC0029197	-1073573481	DTS_E_MSMQTASK_NO_CO NNECTION	The MSMQ connection name is not set.
0xC0029198	-1073573480	DTS_E_MSMQTASK_INITIALI ZATION_WITH_WRONG_XM L_ELEMENT	Task was initialized with the wrong XML element.
0xC0029199	-1073573479	DTS_E_MSMQTASK_DATA_FI LE_NAME_EMPTY	Data file name is empty.
0xC002919A	-1073573478	DTS_E_MSMQTASK_DATA_FI LE_SAVE_NAME_EMPTY	The name specified for the data file to save is empty.
0xC002919B	-1073573477	DTS_E_MSMQTASK_DATA_FI LE_SIZE_ERROR	File size should be less than 4 MB.
0xC002919C	-1073573476	DTS_E_MSMQTASK_DATA_FI LE_SAVE_FAILED	Saving the data file failed.
0xC002919D	-1073573475	DTS_E_MSMQTASK_STRING_ COMPARE_VALUE_MISSING	String filter value is empty.
0xC002919E	-1073573474	DTS_E_MSMQTASK_INVALI D_QUEUE_PATH	Queue path is not valid.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC002919F	-1073573473	DTS_E_MSMQTASK_NOT_TRANSACTIONAL	The message queue task does not support enlisting in distributed transactions.
0xC00291A0	-1073573472	DTS_E_MSMQTASK_INVALID_MESSAGE_TYPE	The message type is not valid.
0xC00291A1	-1073573471	DTS_E_MSMQTASK_TASK_TIMEOUT	The message queue timed out. No message has been received.
0xC00291A2	-1073573470	DTS_E_MSMQTASK_INVALID_PROPERTY_VALUE	The property specified is not valid. Verify that the argument type is correct.
0xC00291A3	-1073573469	DTS_E_MSMQTASK_MESSAGE_NON_AUTHENTICATED	Message is not authenticated.
0xC00291A4	-1073573468	DTS_E_MSMQTASK_INVALID_ENCRYPTION_ALGORITHM	You are trying to set the value of Encryption Algorithm with an invalid object.
0xC00291A5	-1073573467	DTS_E_MSMQTASK_VARIABLE_TO_RECEIVE_STRING_MESSAGE_EMPTY	The variable to receive string message is empty.
0xC00291A6	-1073573466	DTS_E_MSMQTASK_RECEIVE_VARIABLE_EMPTY	Variable to receive variable message is empty.
0xC00291A7	-1073573465	DTS_E_MSMQTASK_CONNECTIONTYPE_NOT_MSMQ	Connection "%1" is not of type MSMQ.
0xC00291A8	-1073573464	DTS_E_MSMQTASK_DATAFILE_ALREADY_EXISTS	The data file "%1" already exists at the specified location. Cannot overwrite the file as the Overwrite option is set to false.
0xC00291A9	-1073573463	DTS_E_MSMQTASK_STRING_MESSAGE_TO_VARIABLE_NOT_FOUND	The specified variable "%1" to receive string message is not found in the package variable collection.
0xC00291AA	-1073573462	DTS_E_MSMQTASK_CONNECTION_MANAGER_NULL	The connection manager "%1" is empty.
0xC00291AB	-1073573461	DTS_E_MSMQTASK_CONNECTION_MANAGER_DOES_NOT_EXIST	The connection manager "%1" does not exist.
0xC00291AC	-1073573460	DTS_E_SCRIPTTASK_COMPILE_ERROR_MSG	Error "%1": "%2"\\r\\nLine "%3" Column "%4" through "%5".

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC00291AD	-1073573459	DTS_E_SCRIPTTASK_COMPILEERRORMSG2	There was an error compiling the script: "%1".
0xC00291AE	-1073573458	DTS_E_SCRIPTTASK_COMPILEERRORMSG3	Error "%1": "%2"\r\nLine "%3" Columns "%4"-%5"\r\nLine Text: "%6".
0xC00291AF	-1073573457	DTS_E_SCRIPTTASK_SCRIPTREPORTEDFAILURE	User script returned a failure result.
0xC00291B0	-1073573456	DTS_E_SCRIPTTASK_SCRIPTFILESFAILEDTOLOAD	User script files failed to load.
0xC00291B1	-1073573455	DTS_E_SCRIPTTASK_SCRIPTTHREWEXCEPTION	User script threw an exception: "%1".
0xC00291B2	-1073573454	DTS_E_SCRIPTTASK_COULDNOTCREATEENTRYPOINTCLASS	Could not create an instance of entrypoint class "%1".
0xC00291B3	-1073573453	DTS_E_SCRIPTTASK_LOADFROMXML EXCEPTION	There was an exception while loading Script Task from XML: "%1".
0xC00291B4	-1073573452	DTS_E_SCRIPTTASK_SOURCEITEMNOTFOUNDEXCEPTION	Source item "%1" was not found in the package.
0xC00291B5	-1073573451	DTS_E_SCRIPTTASK_BINARYITEMNOTFOUNDEXCEPTION	Binary item "%1" was not found in the package.
0xC00291B6	-1073573450	DTS_E_SCRIPTTASK_UNRECOGNIZEDSCRIPTLANGUAGE EXCEPTION	"%1" was not recognized as a valid script language.
0xC00291B7	-1073573449	DTS_E_SCRIPTTASK_ILLEGALSCRIPTNAME	The script name is not valid. It cannot contain spaces, slashes, special characters, or begin with a number.
0xC00291B8	-1073573448	DTS_E_SCRIPTTASK_INVALIDSCRIPTLANGUAGE	The script language specified is not valid.
0xC00291B9	-1073573447	DTS_E_SCRIPTTASK_CANNOTINITNULLTASK	Cannot initialize to a null task.
0xC00291BA	-1073573446	DTS_E_SCRIPTTASK_MUSTINITWITHRIGHTTASK	The Script Task user interface must initialize to an Script Task.
0xC00291BB	-1073573445	DTS_E_SCRIPTTASK_WASNOTINITED	The Script Task user interface is not initialized.
0xC00291BC	-1073573444	DTS_E_SCRIPTTASK_HOSTNAME_CANNOT_EMPTY	Name cannot be empty.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC00291BD	-1073573443	DTS_E_SCRIPTTASK_INVALID_SCRIPT_NAME	The project name is not valid. It cannot contain spaces, slashes, special characters, or begin with a number.
0xC00291BE	-1073573442	DTS_E_SCRIPTTASK_INVALID_SCRIPT_LANGUAGE	The script language specified is not valid.
0xC00291BF	-1073573441	DTS_E_SCRIPTTASK_INVALID_ENTRY_POINT	Entry point not found.
0xC00291C0	-1073573440	DTS_E_SCRIPTTASK_LANGUAGE_EMPTY	The script language is not specified. Verify that a valid script language is specified.
0xC00291C1	-1073573439	DTS_E_SCRIPTTASK_INITIALIZATION_WITH_NULL_TASK	User interface initialization: The task is null.
0xC00291C2	-1073573438	DTS_E_SCRIPTTASK_UI_INITIALIZATION_WITH_WRONG_TASK	The Script Task user interface is initialized with an incorrect task.
0xC00291C3	-1073573437	DTS_E_SENDMAILTASK_RECIPIENT_EMPTY	No recipient is specified.
0xC00291C4	-1073573436	DTS_E_SENDMAILTASK_SMTP_SERVER_NOT_SPECIFIED	The Simple Mail Transfer Protocol (SMTP) server is not specified. Provide a valid name or IP address of the SMTP server.
0xC00291C5	-1073573435	DTS_E_SENDMAILTASK_TASK_INITIALIZATION_WITH_WRONG_XML_ELEMENT	Send Mail task is initiated with an incorrect XML element.
0xC00291CB	-1073573429	DTS_E_SENDMAILTASK_INVALID_ATTACHMENT	Either the file "%1" does not exist or you do not have permissions to access the file.
0xC00291CD	-1073573427	DTS_E_SENDMAILTASK_CHECK_VALID_SMTP_SERVER	Verify that the Simple Mail Transfer Protocol (SMTP) server specified is valid.
0xC00291CE	-1073573426	DTS_E_SENDMAILTASK_CONNECTION_TYPE_NOT_FILE	Connection "%1" is not of type File.
0xC00291CF	-1073573425	DTS_E_SENDMAILTASK_FILE_DOES_NOT_EXIST	On operation "%1", file "%2" does not exist.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC00291D0	-1073573424	DTS_E_SENDMAILTASK_VARIABLETYPEISNOTSTRING	Variable "%1" is not of type string.
0xC00291D1	-1073573423	DTS_E_SENDMAILTASK_CONNECTIONTYPEISNOTSMTP	Connection "%1" is not of type SMTP.
0xC00291D2	-1073573422	DTS_E_SENDMAILTASK_CONNECTIONISNULL	Connection "%1" is empty.
0xC00291D3	-1073573421	DTS_E_SENDMAILTASK_CONNECTIONMANAGER	The specified connection "%1" does not exist.
0xC00291D4	-1073573420	DTS_E_SQLTASK_NOSTATEMENTSPECIFIED	No Transact-SQL statement specified.
0xC00291D5	-1073573419	DTS_E_SQLTASK_NOXMLSUPPORT	The connection does not support XML result sets.
0xC00291D6	-1073573418	DTS_E_SQLTASK_NOHANDLERFORCONNECTION	Cannot locate a handler for the specified connection type.
0xC00291D7	-1073573417	DTS_E_SQLTASK_NOCONNECTIONMANAGER	No connection manager is specified.
0xC00291D8	-1073573416	DTS_E_SQLTASK_CANNOTACQUIRECONNECTIONMANAGER	Cannot acquire a connection from the connection manager.
0xC00291D9	-1073573415	DTS_E_SQLTASK_NULLPARAMETERNAME	Cannot have a null parameter name.
0xC00291DA	-1073573414	DTS_E_SQLTASK_INVALIDPARAMETERNAME	The parameter name is not valid.
0xC00291DB	-1073573413	DTS_E_SQLTASK_VALIDPARAMETER TYPES	Valid parameter names are of type Int or String.
0xC00291DC	-1073573412	DTS_E_SQLTASK_READONLY VARIABLE	Variable "%1" cannot be used in a result binding because it is read-only.
0xC00291DD	-1073573411	DTS_E_SQLTASK_INDEXNOTIN COLLECTION	The index is not assigned in this collection.
0xC00291DE	-1073573410	DTS_E_SQLTASK_READONLYPARAMETER	The variable "%1" cannot be used as an "out" parameter or return value in a parameter binding because it is read-only.
0xC00291DF	-1073573409	DTS_E_SQLTASK_OBJECTNOTIN COLLECTION	The object does not exist in this collection.



HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC00291E0	-1073573408	DTS_E_SQLTASK_UNABLETOACQUIREMANAGEDCONN	Cannot acquire a managed connection.
0xC00291E1	-1073573407	DTS_E_UNABLETOPOPRESULT	Cannot populate the result columns for a single row result type. The query returned an empty result set.
0xC00291E2	-1073573406	DTS_E_SQLTASK_INVALIDNUMBEROFRESULTBINDINGS	There is an invalid number of result bindings returned for the ResultSetType: "%1".
0xC00291E3	-1073573405	DTS_E_SQLTASK_RESULTBINDINGTYPEFORROWSETXML	The result binding name must be set to zero for full result set and XML results.
0xC00291E4	-1073573404	DTS_E_SQLTASK_INVALIDPARAMDIRECTIONFLAG	The parameter directions flag is not valid.
0xC00291E5	-1073573403	DTS_E_SQLTASK_NOSQLTASKDATAINXMLFRAGMENT	The XML fragment does not contain SQL Task data.
0xC00291E6	-1073573402	DTS_E_SQLTASK_MULTIPLERETURNVALUEPARAM	A parameter with type return value is not the first parameter, or there are more than one parameter of type return value.
0xC00291E7	-1073573401	DTS_E_SQLTASK_CONNECTIONTYPENOTFILE	Connection "%1" is not a file connection manager.
0xC00291E8	-1073573400	DTS_E_SQLTASK_FILEDOESNOTEXIST	File represented by "%1" does not exist.
0xC00291E9	-1073573399	DTS_E_SQLTASK_VARIABLETYPEISNOTSTRING	Type of variable "%1" is not string.
0xC00291EA	-1073573398	DTS_E_SQLTASK_VARIABLENOTFOUND	Variable "%1" does not exist or could not be locked.
0xC00291EB	-1073573397	DTS_E_SQLTASK_CANNOTLOCATECONNMANAGER	Connection manager "%1" does not exist.
0xC00291EC	-1073573396	DTS_E_SQLTASK_FAILEDTOACQUIRECONNECTION	Failed to acquire connection "%1". Connection may not be configured correctly or you may not have the right permissions on this connection.
0xC00291ED	-1073573395	DTS_E_SQLTASK_RESULTBYNAMENOTSUPPORTED	Result binding by name "%1" is not supported for this connection type.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC00291EE	-1073573394	DTS_E_SQLTASKCONN_ERR_NO_ROWS	A result set type of single row is specified, but no rows were returned.
0xC00291EF	-1073573393	DTS_E_SQLTASKCONN_ERR_NO_DISCONNECTED_RS	No disconnected record set is available for the Transact-SQL statement.
0xC00291F0	-1073573392	DTS_E_SQLTASKCONN_ERR_UNSUPPORTED_TYPE	Unsupported type.
0xC00291F1	-1073573391	DTS_E_SQLTASKCONN_ERR_UNKNOWN_TYPE	Unknown type.
0xC00291F2	-1073573390	DTS_E_SQLTASKCONN_ERR_PARAM_DATA_TYPE	Unsupported data type on parameter binding \"%s\".
0xC00291F3	-1073573389	DTS_E_SQLTASKCONN_ERR_PARAM_NAME_MIX	Parameter names cannot be an mix of ordinal and named types.
0xC00291F4	-1073573388	DTS_E_SQLTASKCONN_ERR_PARAM_DIR	The parameter direction on parameter binding \"%s\" is not valid.
0xC00291F5	-1073573387	DTS_E_SQLTASKCONN_ERR_RESULT_DATA_TYPE	The data type on result set binding \"%s\" is not supported.
0xC00291F6	-1073573386	DTS_E_SQLTASKCONN_ERR_RESULT_COL_INDEX	The result column index %d is not valid.
0xC00291F7	-1073573385	DTS_E_SQLTASKCONN_ERR_UNKNOWN_RESULT_COL	Cannot find column \"%s\" in the result set.
0xC00291F9	-1073573383	DTS_E_SQLTASKCONN_ERR_NOROWSET	No result rowset is associated with the execution of this query.
0xC00291FA	-1073573382	DTS_E_SQLTASKCONN_ERR_ODBC_DISCONNECTED	Disconnected recordsets are not available from ODBC connections.
0xC00291FB	-1073573381	DTS_E_SQLTASKCONN_ERR_RESULT_SET_DATA_TYPE	The data type in the result set, column %hd, is not supported.
0xC00291FC	-1073573380	DTS_E_SQLTASKCONN_ERR_CANT_LOAD_XML	Cannot load XML with query result.
0xC00291FD	-1073573379	DTS_E_TTGENTASK_NOCON NORVARIABLE	A connection name or variable name for the package must be specified.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC00291FE	-1073573378	DTS_E_TTGENTASK_FAILEDCREATE	Failed to create the package.
0xC00291FF	-1073573377	DTS_E_TTGENTASK_BADTABLEMETADATA	The TableMetaDataNode is not an XMLNode.
0xC0029200	-1073573376	DTS_E_TTGENTASK_FAILEDCREATEPIPELINE	Failed to create the pipeline.
0xC0029201	-1073573375	DTS_E_TTGENTASK_BADVARIABLETYPE	The variable is not the correct type.
0xC0029202	-1073573374	DTS_E_TTGENTASK_NOTFILECONNECTION	The connection manager specified is not a FILE connection manager.
0xC0029203	-1073573373	DTS_E_TTGENTASK_BADFILENAME	Invalid file name specified on the connection manager "%1".
0xC0029204	-1073573372	DTS_E_WEBSERVICETASK_CONNECTION_NAME_NULL	The connection is empty. Verify that a valid HTTP connection is specified.
0xC0029205	-1073573371	DTS_E_WEBSERVICETASK_CONNECTION_NOT_FOUND	The connection does not exist. Verify that a valid, existing HTTP connection is specified.
0xC0029206	-1073573370	DTS_E_WEBSERVICETASK_CONNECTION_NOT_HTTP	The connection specified is not a HTTP connection. Verify that a valid HTTP connection is specified.
0xC0029207	-1073573369	DTS_E_WEBSERVICETASK_SERVICE_NULL	The Web Service name is empty. Verify that a valid web service name is specified.
0xC0029208	-1073573368	DTS_E_WEBSERVICETASK_METHODNAME_NULL	The web method name is empty. Verify that a valid web method is specified.
0xC0029209	-1073573367	DTS_E_WEBSERVICETASK_WEBMETHODINFO_NULL	The web method is empty or may not exist. Verify that there is an existing web method to specify.
0xC002920A	-1073573366	DTS_E_WEBSERVICETASK_OUTPUTLOC_NULL	The output location is empty. Verify that an existing file connection or variable is specified.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC002920B	-1073573365	DTS_E_WEBSERVICETASK_V ARIABLE_NOT_FOUND	The variable cannot be found. Verify that the variable exists in the package.
0xC002920C	-1073573364	DTS_E_WEBSERVICETASK_V ARIABLE_READONLY	Cannot save the result. Verify that the variable is not read-only.
0xC002920D	-1073573363	DTS_E_WEBSERVICETASK_ER ROR_IN_LOAD_FROM_XML	Error occurred in LoadFromXML at the tag "%1".
0xC002920E	-1073573362	DTS_E_WEBSERVICETASK_ER ROR_IN_SAVE_TO_XML	Error occurred in SaveToXML at the tag "%1".
0xC002920F	-1073573361	DTS_E_WEBSERVICETASK_TA SK_SAVE_TO_NULL_XML_EL EMENT	Cannot save the task to a null XML document.
0xC0029210	-1073573360	DTS_E_WEBSERVICETASK_TA SK_INITIALIZATION_WITH_N ULL_XML_ELEMENT	Cannot initialize the task with a null XML element.
0xC0029211	-1073573359	DTS_E_WEBSERVICETASK_TA SK_INITIALIZATION_WITH_ WRONG_XML_ELEMENT	The Web Service Task is initiated with an incorrect XML element.
0xC0029212	-1073573358	DTS_E_WEBSERVICETASK_U NEXPECTED_XML_ELEMENT	Unexpected XML element found.
0xC0029213	-1073573357	DTS_E_WEBSERVICETASK_C ANNOT_ACQUIRE_CONNEC TION	There was an error acquiring the HTTP connection. Verify that a valid connection type is specified.
0xC0029214	-1073573356	DTS_E_WEBSERVICETASK_FI LE_CONN_NOT_FOUND	Cannot save the result. Verify that there is an existing file connection.
0xC0029215	-1073573355	DTS_E_WEBSERVICETASK_FI LE_NOT_FOUND	Cannot save the result. Verify that the file exists.
0xC0029216	-1073573354	DTS_E_WEBSERVICETASK_FI LE_NULL	Cannot save the result. The file name is empty or the file is in use by another process.
0xC0029217	-1073573353	DTS_E_WEBSERVICETASK_C ANNOT_ACQUIRE_FILE_CO NNECTION	There was an error in acquiring the file connection. Verify that a valid file connection is specified.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0029218	-1073573352	DTS_E_WEBSERVICETASK_D ATATYPE_NOT_SUPPORTED	Only Complex Types with Primitive values, Primitive Arrays, and Enumerations are supported.
0xC0029219	-1073573351	DTS_E_WEBSERVICETASK_PA RAMTYPE_NOT_SUPPORTED	Only Primitive, Enum, Complex, PrimitiveArray, and ComplexArray types are supported.
0xC002921A	-1073573350	DTS_E_WEBSERVICETASK_W SDL_VERSION_NOT_SUPPO RTED	This version of WSDL is not supported.
0xC002921B	-1073573349	DTS_E_WEBSERVICETASK_W RONG_XML_ELEMENT	Initialized with an incorrect XML element.
0xC002921C	-1073573348	DTS_E_WEBSERVICETASK_X ML_ATTRIBUTE_NOT_FOUN D	A mandatory attribute is not found.
0xC002921D	-1073573347	DTS_E_WEBSERVICETASK_EN UM_NO_VALUES	The enum "%1" does not have any values. The WSDL is corrupted.
0xC002921E	-1073573346	DTS_E_WEBSERVICETASK_C ONNECTIONNOTFOUND	The connection cannot be found.
0xC002921F	-1073573345	DTS_E_WEBSERVICETASK_C ONNECTION_ALREADY_EXIS TS	Connection by this name already exists.
0xC0029220	-1073573344	DTS_E_WEBSERVICETASK_N ULL_CONNECTION	Connection cannot be null or empty.
0xC0029221	-1073573343	DTS_E_WEBSERVICETASK_N OT_HTTP_CONNECTION	The connection specified is not a HTTP connection. Verify that a valid HTTP connection is specified.
0xC0029222	-1073573342	DTS_E_WEBSERVICETASK_W SDL_NOT_FOUND	The specified Uniform Resource Identifier (URI) does not contain a valid WSDL.
0xC0029223	-1073573341	DTS_E_WEBSERVICETASK_ER ROR_IN_DOWNLOAD	Could not read the WSDL file. The input WSDL file is not valid. The reader threw the following error: "%1".
0xC0029224	-1073573340	DTS_E_WEBSERVICETASK_SE RVICE_DESC_NULL	Service Description cannot be null.
0xC0029225	-1073573339	DTS_E_WEBSERVICETASK_SE RVICENULL	Service name cannot be null.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0029226	-1073573338	DTS_E_WEBSERVICETASK_W SDL_NULL	URL cannot be null.
0xC0029227	-1073573337	DTS_E_WEBSERVICETASK_SE RVICE_NOT_FOUND	The service is not currently available.
0xC0029228	-1073573336	DTS_E_WEBSERVICETASK_S OAPPORT_NOT_FOUND	The service is not available on the SOAP port.
0xC0029229	-1073573335	DTS_E_WEBSERVICETASK_S OAPBINDING_NOT_FOUND	Failed to parse the Web Services Description Language (WSDL). Cannot find the Binding that corresponds to the SOAP port.
0xC002922A	-1073573334	DTS_E_WEBSERVICETASK_S OAPPORTTYPE_NOT_FOUN D	Failed to parse the Web Services Description Language (WSDL). Cannot find a PortType that corresponds to the SOAP port.
0xC002922B	-1073573333	DTS_E_WEBSERVICETASK_M SG_NOT_FOUND	Cannot find the message that corresponds to the method specified.
0xC002922C	-1073573332	DTS_E_WEBSERVICETASK_C ANNOT_GEN_PROXY	Could not generate the proxy for the given web service. The following errors were encountered while generating the proxy "%1".
0xC002922D	-1073573331	DTS_E_WEBSERVICETASK_C ANNOT_LOAD_PROXY	Could not load the proxy for the given web service. The exact error is as follows: "%1".
0xC002922E	-1073573330	DTS_E_WEBSERVICETASK_IN VALID_SERVICE	Could not find the specified service. The exact error is as follows: "%1".
0xC002922F	-1073573329	DTS_E_WEBSERVICETASK_W EBMETHOD_INVOKE_FAILE D	The Web Service threw the following error during method execution: "%1".
0xC0029230	-1073573328	DTS_E_WEBSERVICETASK_IN VOKE_ERR	Could not execute the web method. The exact error is as follows: "%1".
0xC0029231	-1073573327	DTS_E_WEBSERVICETASK_M ETHODINFO_NULL	MethodInfo cannot be null.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0029232	-1073573326	DTS_E_WEBSERVICETASK_VALUE_NOT_PRIMITIVE	The specified WebMethodInfo is not correct. The ParamValue supplied does not match the ParamType. The DTSParamValue is not of type PrimitiveValue.
0xC0029233	-1073573325	DTS_E_WEBSERVICETASK_VALUE_NOT_ENUM	The WebMethodInfo specified is not correct. The ParamValue supplied does not match the ParamType. The DTSParamValue found is not of type EnumValue.
0xC0029234	-1073573324	DTS_E_VALUE_WEBSERVICE_TASK_NOT_COMPLEX	The WebMethodInfo specified is not correct. The ParamValue supplied does not match the ParamType. The DTSParamValue found is not of type ComplexValue.
0xC0029235	-1073573323	DTS_E_WEBSERVICETASK_VALUE_NOT_ARRAY	The WebMethodInfo specified is not correct. The ParamValue supplied does not match the ParamType. The DTSParamValue found is not of type ArrayValue.
0xC0029236	-1073573322	DTS_E_WEBSERVICETASK_TYPE_NOT_PRIMITIVE	The WebMethodInfo you have specified is wrong. "%1" is not Primitive Type.
0xC0029237	-1073573321	DTS_E_WEBSERVICETASK_ARRAY_VALUE_INVALID	The format of the ArrayValue is not valid. There should be at least one element in the array.
0xC0029238	-1073573320	DTS_E_WEBSERVICETASK_SELECTED_VALUE_NULL	The value of the enumeration cannot be null. Select a default value for the enumeration.
0xC0029239	-1073573319	DTS_E_WEBSERVICETASK_NULL_VALUE	Cannot validate a null against any datatype.
0xC002923A	-1073573318	DTS_E_WEBSERVICETASK_ENUM_VALUE_NOT_FOUND	The enumeration Value is not correct.
0xC002923B	-1073573317	DTS_E_WEBSERVICETASK_PROPERTY_NOT_EXISTS	The class specified does not contain a public property by the name "%1".
0xC002923C	-1073573316	DTS_E_WEBSERVICETASK_CONVERT_FAILED	Could not convert "%1" to "%2".

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC002923D	-1073573315	DTS_E_WEBSERVICETASK_CLEANUP_FAILED	Cleanup failed. The proxy that was created for the web service may not have been deleted.
0xC002923E	-1073573314	DTS_E_WEBSERVICETASK_CREATE_INSTANCE_FAILED	Could not create an object of type "%1". Please check whether the default constructor exists.
0xC002923F	-1073573313	DTS_E_WEBSERVICETASK_NOT_PRIMITIVE_TYPE	"%1" is not a value type.
0xC0029240	-1073573312	DTS_E_WEBSERVICETASK_ERROR_IN_VALIDATE	Could not validate "%1" against "%1".
0xC0029241	-1073573311	DTS_E_WEBSERVICETASK_DATA_TYPE_NULL	The data type cannot be null. Specify the value of the data type to validate.
0xC0029242	-1073573310	DTS_E_WEBSERVICETASK_INDEX_OUT_OF_BOUNDS	The ParamValue cannot be inserted at this position. The index specified might be lesser than zero or greater than the length.
0xC0029243	-1073573309	DTS_E_WEBSERVICETASK_WRONG_WSDL	The input WSDL file is not valid.
0xC0029244	-1073573308	DTS_E_WMIDRTASK_SYNCOBJECTFAILED	Synchronization object failed.
0xC0029245	-1073573307	DTS_E_WMIDRTASK_MISSING_WQLQUERY	The WQL query is missing.
0xC0029246	-1073573306	DTS_E_WMIDRTASK_DESTINATIONMUSTBESET	The destination must be set.
0xC0029247	-1073573305	DTS_E_WMIDRTASK_MISSING_CONNECTION	No WMI connection is set.
0xC0029248	-1073573304	DTS_E_WMIDRTASK_INVALID_DATANODE	WMI Data Reader Task received an invalid task data node.
0xC0029249	-1073573303	DTS_E_WMIDRTASK_FAILED_VALIDATION	The task failed validation.
0xC002924A	-1073573302	DTS_E_WMIDRTASK_FILE_DOES_NOT_EXIST	File "%1" does not exist.
0xC002924B	-1073573301	DTS_E_WMIDRTASK_CONNECTION_MGR_DOES_NOT_EXIST	Connection manager "%1" does not exist.



HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC002924C	-1073573300	DTS_E_WMIDRTASK_VARIABLETYPEISNOTSTRINGOROBJECT	Variable "%1" is not of type string or object.
0xC002924D	-1073573299	DTS_E_WMIDRTASK_CONNECTIONTYPEISNOTFILE	Connection "%1" is not of type "FILE".
0xC002924E	-1073573298	DTS_E_WMIDRTASK_CONNECTIONTYPEISNOTWMI	Connection "%1" is not of type "WMI".
0xC002924F	-1073573297	DTS_E_WMIDRTASK_FILEALREADYEXISTS	File "%1" already exists.
0xC0029250	-1073573296	DTS_E_WMIDRTASK_CONNECTIONMANAGEREMPTY	Connection manager "%1" is empty.
0xC0029251	-1073573295	DTS_E_WMIDRTASK_VARNOTOBJECT	Variable "%1" should be of type object to be assigned a data table.
0xC0029252	-1073573294	DTS_E_WMIDRTASK_TASKFAILURE	Task failed due to invalid WMI query: "%1".
0xC0029253	-1073573293	DTS_E_WMIDRTASK_CANTWRITE TO VARIABLE	Unable to write to variable "%1" since it set to keep its original value.
0xC0029254	-1073573292	DTS_E_WMIEWTASK_SYNCHRONIZATIONOBJECTFAILED	Synchronization object failed.
0xC0029255	-1073573291	DTS_E_WMIEWTASK_MISSINGWQLQUERY	The WQL query is missing.
0xC0029256	-1073573290	DTS_E_WMIEWTASK_MISSINGCONNECTION	The WMI connection is missing.
0xC0029257	-1073573289	DTS_E_WMIEWTASK_QUERYFAILURE	The task failed to execute the WMI query.
0xC0029258	-1073573288	DTS_E_WMIEWTASK_INVALIDDATANODE	The WMI Event Watcher Task received a task data node that is not valid.
0xC0029259	-1073573287	DTS_E_WMIEWTASK_CONNECTIONMANAGERDOESNOTEXIST	Connection manager "%1" does not exist.
0xC002925A	-1073573286	DTS_E_WMIEWTASK_FILEDOESNOTEXIST	File "%1" does not exist.
0xC002925B	-1073573285	DTS_E_WMIEWTASK_VARIABLETYPEISNOTSTRING	Variable "%1" is not of type string.
0xC002925C	-1073573284	DTS_E_WMIEWTASK_CONNECTIONTYPEISNOTFILE	Connection "%1" is not of type "FILE".

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC002925D	-1073573283	DTS_E_WMIEWTASK_CONNECTIONTYPENOTWMI	Connection "%1" is not of type "WMI".
0xC002925E	-1073573282	DTS_E_WMIEWTASK_FILEALREADYEXISTS	File "%1" already exists.
0xC002925F	-1073573281	DTS_E_WMIEWTASK_CONNECTIONMANAGEREMPTY	Connection manager "%1" is empty.
0xC0029260	-1073573280	DTS_E_WMIEWTASK_TIMEOUTOCCURRED	Timeout of "%1" second(s) occurred before event represented by "%2".
0xC0029261	-1073573279	DTS_E_WMIEWTASK_ERRORMESSAGE	Watching for the Wql query caused the following system exception: "%1". Check the query for errors or WMI connection for access rights/permissions.
0xC0029262	-1073573278	DTS_E_XMLTASK_NODEFAULTOPERATION	The Operations specified is not defined.
0xC0029263	-1073573277	DTS_E_XMLTASK_CONNECTIONTYPENOTFILE	The connection type is not File.
0xC0029264	-1073573276	DTS_E_XMLTASK_CANTGETREADERFROMSOURCE	Cannot get an XmlReader from the source XML document.
0xC0029265	-1073573275	DTS_E_XMLTASK_CANTGETREADERFROMDEST	Cannot get an XmlReader from the changed XML document.
0xC0029266	-1073573274	DTS_E_XMLTASK_CANTGETREADERFROMDIFFGRAM	Cannot get the XDL diffgram reader from the XDL diffgram XML.
0xC0029268	-1073573272	DTS_E_XMLTASK_EMPTYNODELIST	The node list is empty.
0xC0029269	-1073573271	DTS_E_XMLTASK_NOELEMENTFOUND	The element was not found.
0xC002926A	-1073573270	DTS_E_XMLTASK_UNDEFINEOPERATION	The Operations specified is not defined.
0xC002926B	-1073573269	DTS_E_XMLTASK_XPATHNAVIGATORERROR	Unexpected content item in XPathNavigator.
0xC002926C	-1073573268	DTS_E_XMLTASK_NOSCHEMAFOUND	No schema found to enforce validation.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC002926D	-1073573267	DTS_E_XMLTASK_VALIDATIONERROR	A validation error occurred when validating the instance document.
0xC002926E	-1073573266	DTS_E_XMLTASK_SYNCOBJECTFAILED	Synchronization object failed.
0xC002926F	-1073573265	DTS_E_XMLTASK_ROOTNODESNOTMATCHED	The root nodes do not match.
0xC0029270	-1073573264	DTS_E_XMLTASK_INVALIDEDITSCRIPT	The Edit Script Operation type in the final Edit Script is not valid.
0xC0029271	-1073573263	DTS_E_XMLTASK_CDATANODESNOTADDED	CDATA nodes should be added with DiffgramAddSubtrees class.
0xC0029272	-1073573262	DTS_E_XMLTASK_COMMENTNODESNOTADDED	Comment nodes should be added with DiffgramAddSubtrees class.
0xC0029273	-1073573261	DTS_E_XMLTASK_TEXTNODESNOTADDED	Text nodes should be added with DiffgramAddSubtrees class.
0xC0029274	-1073573260	DTS_E_XMLTASK_WHITESPACEISSUE	Significant white space nodes should be added with DiffgramAddSubtrees class.
0xC0029275	-1073573259	DTS_E_XMLTASK_DIFFENUMISSUE	Correct the OperationCost array so that it reflects the XmlDiffOperation enumeration.
0xC0029276	-1073573258	DTS_E_XMLTASK_TASKISEMPTY	There are no operations in the task.
0xC0029277	-1073573257	DTS_E_XMLTASK_DOCUMENTALREADYHASDATA	The document already contains data and should not be used again.
0xC0029278	-1073573256	DTS_E_XMLTASK_INVALIDNODETYPE	The node type is not valid.
0xC0029279	-1073573255	DTS_E_XMLTASK_INVALIDDATANODE	The XML Task received a task data node that is not valid.
0xC002927B	-1073573253	DTS_E_XMLTASK_VARIABLETYPEISNOTSTRING	Variable data type is not a String.
0xC002927C	-1073573252	DTS_E_XMLTASK_COULDNOTGETENCODINGFROMDOCUMENT	Cannot get encoding from XML.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC002927D	-1073573251	DTS_E_XMLTASK_MISSINGS OURCE	Source is not specified.
0xC002927E	-1073573250	DTS_E_XMLTASK_MISSINGS ECONDOPERAND	Second operand is not specified.
0xC002927F	-1073573249	DTS_E_XMLTASK_INVALIDP ATHDESCRIPTOR	Invalid XDL diffgram. "%1" is an invalid path descriptor.
0xC0029280	-1073573248	DTS_E_XMLTASK_NOMATC HINGNODE	Invalid XDL diffgram. No node matches the path descriptor "%1".
0xC0029281	-1073573247	DTS_E_XMLTASK_EXPECTIN GDIFFGRAMELEMENT	Invalid XDL diffgram. Expecting xd:xmldiff as a root element with namespace URI "%1".
0xC0029282	-1073573246	DTS_E_XMLTASK_MISSINGS RCDOCATTRIBUTE	The XDL diffgram is not valid. The srcDocHash attribute on the xd:xmldiff element is missing.
0xC0029283	-1073573245	DTS_E_XMLTASK_MISSINGO PTIONSATTRIBUTE	The XDL diffgram is not valid. The options attribute on the xd:xmldiff element is missing.
0xC0029284	-1073573244	DTS_E_XMLTASK_INVALIDSR CDOCATTRIBUTE	The XDL diffgram is not valid. The srcDocHash attribute has an invalid value.
0xC0029285	-1073573243	DTS_E_XMLTASK_INVALIDO PTIONSATTRIBUTE	The XDL diffgram is not valid. The options attribute has an invalid value.
0xC0029286	-1073573242	DTS_E_XMLTASK_SRCDOCM ISMATCH	The XDL diffgram is not applicable to this XML document. The rcDocHash value does not match.
0xC0029287	-1073573241	DTS_E_XMLTASK_MORETHA NONENODEMATCHED	Invalid XDL diffgram; more than one node matches the "%1" path descriptor on the xd:node or xd:change element.
0xC0029288	-1073573240	DTS_E_XMLTASK_XMLDECL MISMATCH	The XDL diffgram is not applicable to this XML document. A new XML declaration cannot be added.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0029289	-1073573239	DTS_E_XMLTASK_INTERNAL ERRORMORETHANONENOD EINLIST	Internal Error. XmlDiffPathSingleNodeList can contain only one node.
0xC002928A	-1073573238	DTS_E_XMLTASK_INTERNAL ERRORMORETHANONENOD ELEFT	Internal Error. "%1" nodes left after patch, expecting 1.
0xC002928B	-1073573237	DTS_E_XMLTASK_XSLTRESUL TFILEISNOTXML	The File/Text Produced by the XSLT is not a valid XmlDocument, thus can not be set as result of operation: "%1".
0xC002928E	-1073573234	DTS_E_XMLTASK_FILEDOES NOTEXIST	There is no file associated with connection "%1".
0xC002928F	-1073573233	DTS_E_XMLTASK_XMLTEXTE MPTY	Property "%1" has no source Xml text; Xml Text is either invalid, null or empty string.
0xC0029290	-1073573232	DTS_E_XMLTASK_FILEALREA DYEXISTS	File "%1" already exists.
0xC0029293	-1073573229	DTS_E_TRANSFERTASKS_SRC CONNECTIONREQUIRED	A source connection must be specified.
0xC0029294	-1073573228	DTS_E_TRANSFERTASKS_DES TCONNECTIONREQUIRED	A destination connection must be specified.
0xC0029295	-1073573227	DTS_E_TRANSFERTASKS_CO NNECTIONNOTFOUND	The connection "%1" could not be found in the package.
0xC0029296	-1073573226	DTS_E_TRANSFERTASKS_SER VERVERSIONNOTALLOWED	The connection "%1" specifies a SQL Server instance with a version that is not supported for transfer. Only versions 7, 2000, and 2005 are supported.
0xC0029297	-1073573225	DTS_E_TRANSFERTASKS_SRC SERVERLESSEQUALDESTSER VER	The source connection "%1" must specify a SQL Server instance with a version earlier than or the same as the destination connection "%2".
0xC0029298	-1073573224	DTS_E_TRANSFERTASKS_SRC DBREQUIRED	A source database must be specified.
0xC0029299	-1073573223	DTS_E_TRANSFERTASKS_SRC DBMUSTEXIST	The source database "%1" must exist on the source server.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC002929A	-1073573222	DTS_E_TRANSFERTASKS_DES TDBREQUIRED	A destination database must be specified.
0xC002929B	-1073573221	DTS_E_TRANSFERTASKS_SRC DBANDDESTDBTHESAME	The source database and the destination database can not be the same.
0xC002929C	-1073573220	DTS_E_TRANSFERDBTASK_FI LENAMEREQUIRED	The transfer file information %1 is missing the filename.
0xC002929D	-1073573219	DTS_E_TRANSFERDBTASK_F OLDERREQUIRED	The transfer file information %1 is missing the folder part.
0xC002929E	-1073573218	DTS_E_TRANSFERTASKS_NET SHAREREQUIRED	The transfer file information %1 is missing the network share part.
0xC002929F	-1073573217	DTS_E_TRANSFERTASKS_FILE LISTSCOUNTMISMATCH	The number of source transfer files and the number of destination transfer files must be the same.
0xC00292A0	-1073573216	DTS_E_DOESNOTSUPPORTT RANSACTIONS	Enlisting in transactions is not supported.
0xC00292A1	-1073573215	DTS_E_TRANSFERDBTASK_O FFLINEERROR	The following exception occurred during an offline database transfer: %1.
0xC00292A2	-1073573214	DTS_E_TRANSFERDBTASK_N ETSHAREDOESNOTEXIST	The network share "%1" could not be found.
0xC00292A3	-1073573213	DTS_E_TRANSFERDBTASK_N ETSHARENOACCESS	The network share "%1" could not be accessed. The error is: %2.
0xC00292A4	-1073573212	DTS_E_TRANSFERDBTASK_U SERMUSTBEDBOORSYSADM IN	The user "%1" must be a DBO or a sysadmin for "%2" in order to perform an online database transfer.
0xC00292A5	-1073573211	DTS_E_TRANSFERDBTASK_U SERMUSTBESYSADMIN	The user "%1" must be a sysadmin on "%2" to perform an offline database transfer.
0xC00292A6	-1073573210	DTS_E_TRANSFERDBTASK_FT CATALOGSOFFLINEYUKON ONLY	Full text catalogs can only be included when performing an offline database transfer between 2 SQL Server 2005 servers.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC00292A7	-1073573209	DTS_E_TRANSFERDBTASK_N OOVERWRITEDB	The database "%1" already exists on the destination server "%2".
0xC00292A8	-1073573208	DTS_E_TRANSFERDBTASK_M USTHAVESOURCEFILES	At least one source file must be specified.
0xC00292A9	-1073573207	DTS_E_TRANSFERDBTASKS_S RCFILENOTFOUND	Could not find the file "%1" in the source database "%2".
0xC00292B3	-1073573197	DTS_E_MSMQTASK_FIPS140 2COMPLIANCE	The operation requested is not allowed in systems compliant with U.S. FIPS 140-2.
0xC002F210	-1073548784	DTS_E_SQLTASK_ERROREXE CUTINGTHEQUERY	Executing the query "%1" failed with the following error: "%2". Possible failure reasons: Problems with the query, "ResultSet" property not set correctly, parameters not set correctly, or connection not established correctly.
0xC002F300	-1073548544	DTS_E_TRANSFERSPTASK_ER RORREADINGSPNAMES	Error reading stored procedure names from the xml file.
0xC002F301	-1073548543	DTS_E_TRANSFERSPTASK_IN VALIDDATANODE	Invalid data node for the Transfer Stored Procedure task.
0xC002F302	-1073548542	DTS_E_TRANSFERTASKS_CO NNECTIONTYPEISNOTSMOS SERVER	Connection "%1" is not of type "SMOServer".
0xC002F303	-1073548541	DTS_E_TRANSFERSPTASK_EX ECUTIONFAILED	Execution failed with the following error "%1".
0xC002F304	-1073548540	DTS_E_ERROROCCURREDWI THFOLLOWINGMESSAGE	An error occurred with the following error message: "%1".
0xC002F305	-1073548539	DTS_E_BITASK_EXECUTION_ FAILED	Bulk insert execution failed.
0xC002F306	-1073548538	DTS_E_FSTASK_INVALIDDES TPATH	Invalid destination path.
0xC002F307	-1073548537	DTS_E_FSTASK_CANTCREATE DIR	Can not create directory. User chose to fail the task if directory exists.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC002F308	-1073548536	DTS_E_SQLTASK_ODBCNOSUPPORTTRANSACTION	The task has a transaction option of "Required" and connection "%1" is of type "ODBC". ODBC connections don't support transactions.
0xC002F309	-1073548535	DTS_E_SQLTASK_ERRORASSIGNINGVALUETOVAR	An error occurred while assigning a value to variable "%1": "%2".
0xC002F30A	-1073548534	DTS_E_FSTASK_SOURCEISEMPTY	The source is empty.
0xC002F30B	-1073548533	DTS_E_FSTASK_DESTINATIONISEMPTY	The destination is empty.
0xC002F30C	-1073548532	DTS_E_FSTASK_FILEDIRNOTFOUND	File or directory "%1" does not exist.
0xC002F30D	-1073548531	DTS_E_FSTASK_VARSRCORDERISTEMPTY	Variable "%1" is used as a source or destination and is empty.
0xC002F30E	-1073548530	DTS_E_FSTASK_FILEDELETED	File or directory "%1" was deleted.
0xC002F30F	-1073548529	DTS_E_FSTASK_DIRECTORYDELETED	Directory "%1" was deleted.
0xC002F310	-1073548528	DTS_E_WMIDRTASK_VARIABLETYPEISNOTOBJECT	The variable "%1" should be of type object to be assigned a data table.
0xC002F311	-1073548527	DTS_E_WMIDRTASK_VARIABLETYPEISNOTSTRING	The variable "%1" does not have a string data type.
0xC002F312	-1073548526	DTS_E_FTPTASK_CANNOTACQUIRECONNECTION	There was an error acquiring the FTP connection. Verify that a valid connection type is specified in "%1".
0xC002F313	-1073548525	DTS_E_FTPTASK_CONNECTIONNOTFOUND	The FTP connection manager "%1" can not be found.
0xC002F314	-1073548524	DTS_E_FTPTASK_FILEUSAGETYPEERROR	File usage type of connection "%1" should be "%2" for operation "%3".
0xC002F315	-1073548523	DTS_E_TRANSFERTASKS_SOURCECANNOTBESAMEASDESTINATION	The source server can not be the same as the destination server.
0xC002F316	-1073548522	DTS_E_ERRMSGTASK_EMPTYSOURCELIST	There are no Error Messages to transfer.



HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC002F317	-1073548521	DTS_E_ERRMSGTASK_DIFFER ENTMESSAGEANDLANGUA GESIZES	The lists of error messages and their corresponding languages are of different sizes.
0xC002F318	-1073548520	DTS_E_ERRMSGTASK_ERROR MESSAGEOUTOFRANGE	The error message id "%1" is out of the allowed range of user defined error messages. User defined error message ids are between 50000 and 2147483647.
0xC002F319	-1073548519	DTS_E_TRANSFERTASKS_NO TRANSACTIONSUPPORT	This task can not participate in a transaction.
0xC002F320	-1073548512	DTS_E_ERRMSGTASK_FAILE DTOTRANSFERERRORMESSA GES	Failed to transfer some or all of the Error Messages.
0xC002F321	-1073548511	DTS_E_ERRMSGTASK_ERROR MESSAGEALREADYEXISTS	The error message "%1" already exists at destination server.
0xC002F324	-1073548508	DTS_E_ERRMSGTASK_ERROR MESSAGECANTBEFOUND	The error message "%1" can not be found at source server.
0xC002F325	-1073548507	DTS_E_TRANSFERTASKS_EXE CUTIONFAILED	Execution failed with the following error: "%1".
0xC002F327	-1073548505	DTS_E_JOBSTASK_FAILEDTO TRANSFERJOBS	Failed to transfer the Job(s).
0xC002F330	-1073548496	DTS_E_JOBSTASK_EMPTYSO URCELIST	There are no Jobs to transfer.
0xC002F331	-1073548495	DTS_E_JOBSTASK_JOBEXISTS ATDEST	The job "%1" already exists at destination server.
0xC002F334	-1073548492	DTS_E_JOBSTASK_JOBCANT BEFOUND	The job "%1" can not be found at source server.
0xC002F337	-1073548489	DTS_E_LOGINSTASK_EMPTY LIST	The list of "Logins" to transfer is empty.
0xC002F338	-1073548488	DTS_E_LOGINSTASK_CANTG ETLOGINSNAMELIST	Can not get the list of "Logins" from source server.
0xC002F340	-1073548480	DTS_E_LOGINSTASK_ERROR LOGINEXISTS	Login "%1" already exists at destination server.
0xC002F342	-1073548478	DTS_E_LOGINSTASK_LOGIN NOTFOUND	Login "%1" does not exist at source.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC002F344	-1073548476	DTS_E_LOGINTASK_FAILED TOTRANSFERLOGINS	Failed to transfer some or all of the logins.
0xC002F345	-1073548475	DTS_E_STOREDPROCSTASK_FAILED TOTRANSFERSPS	Failed to transfer the stored procedure(s). More informative error should have been raised.
0xC002F346	-1073548474	DTS_E_STOREDPROCSTASK_STOREDPROCNOTFOUND	Stored Procedure "%1" is not found at the source.
0xC002F349	-1073548471	DTS_E_STOREDPROCSTASK_ERRORSTOREDPROCEDURE EXISTS	Stored procedure "%1" already exists at destination server.
0xC002F350	-1073548464	DTS_E_STOREDPROCSTASK_EMPTYSOURCELIST	There are no stored procedures to transfer.
0xC002F353	-1073548461	DTS_E_TRANSOBJECTSTASK_FAILED TOTRANSFEROBJECTS	Failed to transfer the object(s).
0xC002F354	-1073548460	DTS_E_TRANSOBJECTSTASK_EMPTYLIST	The list of "Objects" to transfer is empty.
0xC002F355	-1073548459	DTS_E_TRANSOBJECTSTASK_NOSPATSOURCE	Stored procedure "%1" does not exist at the source.
0xC002F356	-1073548458	DTS_E_TRANSOBJECTSTASK_SPALREADYATDEST	Stored procedure "%1" already exists at destination.
0xC002F357	-1073548457	DTS_E_TRANSOBJECTSTASK_ERRORHANDLINGSPS	An error occurred while trying to get set the Stored Procedures list to transfer: "%1".
0xC002F359	-1073548455	DTS_E_TRANSOBJECTSTASK_NORULEATSOURCE	Rule "%1" does not exist at the source.
0xC002F360	-1073548448	DTS_E_TRANSOBJECTSTASK_RULEALREADYATDEST	Rule "%1" already exists at destination.
0xC002F361	-1073548447	DTS_E_TRANSOBJECTSTASK_ERRORHANDLINGRULES	An error occurred while trying to get set the Rules list to transfer: "%1".
0xC002F363	-1073548445	DTS_E_TRANSOBJECTSTASK_NOTABLEATSOURCE	Table "%1" does not exist at the source.
0xC002F364	-1073548444	DTS_E_TRANSOBJECTSTASK_TABLEALREADYATDEST	Table "%1" already exists at destination.
0xC002F365	-1073548443	DTS_E_TRANSOBJECTSTASK_ERRORHANDLINGTABLES	An error occurred while trying to get set the Tables list to transfer: "%1".

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC002F367	-1073548441	DTS_E_TRANSOBJECTTASK_NOVIEWATSOURCE	View "%1" does not exist at the source.
0xC002F368	-1073548440	DTS_E_TRANSOBJECTTASK_VIEWALREADYATDEST	View "%1" already exists at destination.
0xC002F369	-1073548439	DTS_E_TRANSOBJECTTASK_ERRORHANDLINGVIEWS	An error occurred while trying to get set the Views list to transfer: "%1".
0xC002F371	-1073548431	DTS_E_TRANSOBJECTTASK_NOUDFATSOURCE	User Defined Function "%1" does not exist at the source.
0xC002F372	-1073548430	DTS_E_TRANSOBJECTTASK_UDFALREADYATDEST	User Defined Function "%1" already exists at destination.
0xC002F373	-1073548429	DTS_E_TRANSOBJECTTASK_ERRORHANDLINGUDFS	An error occurred while trying to get set the User Defined Functions list to transfer: "%1".
0xC002F375	-1073548427	DTS_E_TRANSOBJECTTASK_NODEFAULTATSOURCE	Default "%1" does not exist at the source.
0xC002F376	-1073548426	DTS_E_TRANSOBJECTTASK_DEFAULTALREADYATDEST	Default "%1" already exists at destination.
0xC002F377	-1073548425	DTS_E_TRANSOBJECTTASK_ERRORHANDLINGDEFAULTS	An error occurred while trying to get set the Defaults list to transfer: "%1".
0xC002F379	-1073548423	DTS_E_TRANSOBJECTTASK_NOUDDTATSOURCE	User Defined Data Type "%1" does not exist at the source.
0xC002F380	-1073548416	DTS_E_TRANSOBJECTTASK_UDDTALREADYATDEST	User Defined Data Type "%1" already exists at destination.
0xC002F381	-1073548415	DTS_E_TRANSOBJECTTASK_ERRORHANDLINGUDDTS	An error occurred while trying to get set the User Defined Data Types list to transfer: "%1".
0xC002F383	-1073548413	DTS_E_TRANSOBJECTTASK_NOPFATSOURCE	Partition Function "%1" does not exist at the source.
0xC002F384	-1073548412	DTS_E_TRANSOBJECTTASK_PFALREADYATDEST	Partition Function "%1" already exists at destination.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC002F385	-1073548411	DTS_E_TRANSOBJECTTASK_ERRORHANDLINGPFS	An error occurred while trying to get set the Partition Functions list to transfer: "%1".
0xC002F387	-1073548409	DTS_E_TRANSOBJECTTASK_NOPSATSOURCE	Partition Scheme "%1" does not exist at the source.
0xC002F388	-1073548408	DTS_E_TRANSOBJECTTASK_PSALREADYATDEST	Partition Scheme "%1" already exists at destination.
0xC002F389	-1073548407	DTS_E_TRANSOBJECTTASK_ERRORHANDLINGPSS	An error occurred while trying to get set the Partition Schemes list to transfer: "%1".
0xC002F391	-1073548399	DTS_E_TRANSOBJECTTASK_NOSCHEMAATSOURCE	Schema "%1" does not exist at the source.
0xC002F392	-1073548398	DTS_E_TRANSOBJECTTASK_SCHEMAALREADYATDEST	Schema "%1" already exists at destination.
0xC002F393	-1073548397	DTS_E_TRANSOBJECTTASK_ERRORHANDLINGSCHEMAS	An error occurred while trying to get set the Schemas list to transfer: "%1".
0xC002F395	-1073548395	DTS_E_TRANSOBJECTTASK_NOSQLASSEMBLYATSOURCE	SqlAssembly "%1" does not exist at the source.
0xC002F396	-1073548394	DTS_E_TRANSOBJECTTASK_SQLASSEMBLYALREADYATDEST	SqlAssembly "%1" already exists at destination.
0xC002F397	-1073548393	DTS_E_TRANSOBJECTTASK_ERRORHANDLINGSQLASSEMBLIES	An error occurred while trying to get set the SqlAssemblies list to transfer: "%1".
0xC002F399	-1073548391	DTS_E_TRANSOBJECTTASK_NOAGGREGATEATSOURCE	User Defined Aggregate "%1" does not exist at the source.
0xC002F400	-1073548288	DTS_E_TRANSOBJECTTASK_AGGREGATEALREADYATDEST	User Defined Aggregate "%1" already exists at destination.
0xC002F401	-1073548287	DTS_E_TRANSOBJECTTASK_ERRORHANDLINGAGGREGATES	An error occurred while trying to get set the User Defined Aggregates list to transfer: "%1".
0xC002F403	-1073548285	DTS_E_TRANSOBJECTTASK_NOTYPEATSOURCE	User Defined Type "%1" does not exist at the source.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC002F404	-1073548284	DTS_E_TRANSOBJECTTASK_ TYPEALREADYATDEST	User Defined Type "%1" already exists at destination.
0xC002F405	-1073548283	DTS_E_TRANSOBJECTTASK_ ERRORHANDLINGTYPES	An error occurred while trying to get set the User Defined Types list to transfer: "%1".
0xC002F407	-1073548281	DTS_E_TRANSOBJECTTASK_ NOXMLSCHEMACOLLECTIO NATSOURCE	XmlSchemaCollection "%1" does not exist at the source.
0xC002F408	-1073548280	DTS_E_TRANSOBJECTTASK_ XMLSCHEMACOLLECTIONA LREADYATDEST	XmlSchemaCollection "%1" already exists at destination.
0xC002F409	-1073548279	DTS_E_TRANSOBJECTTASK_ ERRORHANDLINGXMLSCHE MACOLLECTIONS	An error occurred while trying to get set the XmlSchemaCollections list to transfer: "%1".
0xC002F411	-1073548271	DTS_E_TRANSOBJECTTASK_ SUPPORTEDONLYNONL Y	Objects of type "%1" are only supported between SQL Server 2005 or newer servers.
0xC002F413	-1073548269	DTS_E_LOGINSTASK_EMPTY DATABASELIST	The databases list is empty.
0xC002F414	-1073548268	DTS_E_TRANSOBJECTTASK_ NOLOGINATSOURCE	Login "%1" does not exist at the source.
0xC002F416	-1073548266	DTS_E_TRANSOBJECTTASK_ LOGINALREADYATDEST	Login "%1" already exists at destination.
0xC002F417	-1073548265	DTS_E_TRANSOBJECTTASK_ ERRORHANDLINGLOGINS	An error occurred while trying to get set the Logins list to transfer: "%1".
0xC002F419	-1073548263	DTS_E_TRANSOBJECTTASK_ NOUSERATSOURCE	User "%1" does not exist at the source.
0xC002F41B	-1073548261	DTS_E_TRANSOBJECTTASK_ USERALREADYATDEST	User "%1" already exists at destination.
0xC002F41C	-1073548260	DTS_E_TRANSOBJECTTASK_ ERRORHANDLINGUSERS	An error occurred while trying to get set the Users list to transfer: "%1".
0xC002F41F	-1073548257	DTS_E_BITASK_CANNOTRET AINCONNINTRANSACTION	The task can not have a retained connection manager in a transaction.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC002F421	-1073548255	DTS_E_SQLTASKOUTPUTENCODINGNOTSUPPORTED	Unable to obtain XML data from SQL Server as Unicode because the provider does not support the OUTPUTENCODING property.
0xC002F426	-1073548250	DTS_E_FTPTASK_FILECONNECTIONNOTFOUND	For the FTP operation "%1", the FILE connection manager "%2" can not be found.
0xC002F428	-1073548248	DTS_E_TRANSOBJECTTASK_CANNOTDROPOBJECTS	"Logins" are server level objects and can not be dropped first since the source and destination are the same server. Dropping objects first will remove the logins from the source as well.
0xC002F429	-1073548247	DTS_E_SQLTASK_PARAMSIZEERROR	Parameter "%1" cannot be negative. (-1) is used for the default value.
0xC0040019	-1073479655	DTS_E_UNREGISTEREDPIPELINEXML_LOAD	Data Flow objects cannot be loaded. Check if Microsoft.SqlServer.PipelineXML.dll is properly registered.
0xC0040020	-1073479648	DTS_E_UNREGISTEREDPIPELINEXML_SAVE	Data Flow objects cannot be saved. Check if Microsoft.SqlServer.PipelineXML.dll is properly registered.
0xC0040040	-1073479616	DTS_E_PIPELINE_SAVE	Failed to save Data Flow objects.
0xC0040041	-1073479615	DTS_E_PIPELINE_LOAD	Failed to load Data Flow objects
0xC0040042	-1073479614	DTS_E_SAVE_PERSTFORMAT	Failed to save Data Flow objects. The specified format is not supported.
0xC0040043	-1073479613	DTS_E_LOAD_PERSTFORMAT	Failed to load Data Flow objects. The specified format is not supported.
0xC0040044	-1073479612	DTS_E_SETPERSIST_PROPERTIES	Failed to set the XML persistence events property for the Data Flow objects.
0xC0040045	-1073479611	DTS_E_SETPERSIST_XMLDOM	Failed to set the persistence XML DOM property for the Data Flow objects.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0040046	-1073479610	DTS_E_SETPERSIST_XMLNODE	Failed to set the persistence XML ELEMENT property for the Data Flow objects.
0xC0040047	-1073479609	DTS_E_SETPERSISTPROP_FAILED	Failed to set xml persistence properties for the Data Flow objects.
0xC0040048	-1073479608	DTS_E_NOCUSTOMPROPCOLL	Failed to get custom property collection for Data Flow components.
0xC0047000	-1073451008	DTS_E_CYCLEINEXECUTIONTREE	An execution tree contains a cycle.
0xC0047001	-1073451007	DTS_E_DISCONNECTEDOBJECT	The %1 object "%2" (%3!d!) is disconnected from the layout.
0xC0047002	-1073451006	DTS_E_INVALIDOBJECTID	The ID for the layout object is not valid.
0xC0047003	-1073451005	DTS_E_INPUTWITHOUTPATHS	A required input object is not connected to a path object.
0xC0047005	-1073451003	DTS_E_INVALIDSYNCHRONOUSINPUT	%1 has an invalid synchronous input ID %2!d!.
0xC0047006	-1073451002	DTS_E_INVALIDOUTPUTLINEAGEID	%1 has lineage ID %2!d!, but should have had %3!d!.
0xC0047008	-1073451000	DTS_E_DUPLICATENAMESINCOLLECTION	The package contains two objects with the duplicate name of "%1" and "%2".
0xC0047009	-1073450999	DTS_E_INVALIDEXCLUSIONGROUP	The "%1" and the "%2" are in the same exclusion group, but they do not have the same synchronous input.
0xC004700A	-1073450998	DTS_E_DUPLICATELINEAGEIDSINCOLLECTION	Two objects in the same collection have a duplicate lineage ID of %1!d!. The objects are %2 and %3.
0xC004700B	-1073450997	DTS_E_VALIDATIONFAILEDONLAYOUT	The layout failed validation.
0xC004700C	-1073450996	DTS_E_VALIDATIONFAILEDONCOMPONENTS	One or more component failed validation.
0xC004700D	-1073450995	DTS_E_VALIDATIONFAILED	The layout and one or more components failed validation.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC004700E	-1073450994	DTS_E_THREADSTARTUPFAILED	The Data Flow task engine failed at startup because it cannot create one or more required threads.
0xC004700F	-1073450993	DTS_E_CANTGETMUTEX	A thread failed to create a mutex at initialization.
0xC0047010	-1073450992	DTS_E_CANTGETSEMAPHORE	A thread failed to create a semaphore at initialization.
0xC0047011	-1073450991	DTS_E_BUFFERFAILUREDETAILS	The system reports %1!d! percent memory load. There are %2 bytes of physical memory with %3 bytes free. There are %4 bytes of virtual memory with %5 bytes free. The paging file has %6 bytes with %7 bytes free.
0xC0047012	-1073450990	DTS_E_BUFFERALLOCFAILED	A buffer failed while allocating %1!d! bytes.
0xC0047013	-1073450989	DTS_E_CANTCREATEBUFFERMANAGER	The Buffer Manager could not be created.
0xC0047015	-1073450987	DTS_E_BUFFERBADSIZE	Buffer Type %1!d! had a size of %2!64d! bytes.
0xC0047016	-1073450986	DTS_E_DANGLINGWITHPATH	%1 is marked as dangling, but has a path attached to it.
0xC0047017	-1073450985	DTS_E_INDIVIDUALVALIDATIONFAILED	%1 failed validation and returned error code 0x%2!8.8X!.
0xC0047018	-1073450984	DTS_E_INDIVIDUALPOSTEXECUTEFAILED	%1 failed the post-execute phase and returned error code 0x%2!8.8X!.
0xC0047019	-1073450983	DTS_E_INDIVIDUALPREPAREFAILED	%1 failed the prepare phase and returned error code 0x%2!8.8X!.
0xC004701A	-1073450982	DTS_E_INDIVIDUALPREEXECUTEFAILED	%1 failed the pre-execute phase and returned error code 0x%2!8.8X!.
0xC004701B	-1073450981	DTS_E_INDIVIDUALCLEANUPFAILED	%1 failed the cleanup phase and returned error code 0x%2!8.8X!.



HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC004701C	-1073450980	DTS_E_INVALIDINPUTLINEAGEID	%1 has lineage ID %2!d! that was not previously used in the Data Flow task.
0xC004701E	-1073450978	DTS_E_EXECUTIONTREECYCLE	Cannot connect %1 to %2 because a cycle would be created.
0xC004701F	-1073450977	DTS_E_CANTCOMPARE	The data type "%1" cannot be compared. Comparison of that data type is not supported, so it cannot be sorted or used as a key.
0xC0047020	-1073450976	DTS_E_REFUSEDFORSHUTDOWN	This thread has shut down and is not accepting buffers for input.
0xC0047021	-1073450975	DTS_E_THREADFAILED	SSIS Error Code DTS_E_THREADFAILED. Thread "%1" has exited with error code 0x%2!8.X!. There may be error messages posted before this with more information on why the thread has exited.
0xC0047022	-1073450974	DTS_E_PROCESSINPUTFAILED	SSIS Error Code DTS_E_PROCESSINPUTFAILED. The ProcessInput method on component "%1" (%2!d!) failed with error code 0x%3!8.X! while processing input "%4" (%5!d!). The identified component returned an error from the ProcessInput method. The error is specific to the component, but the error is fatal and will cause the Data Flow task to stop running. There may be error messages posted before this with more information about the failure.
0xC0047023	-1073450973	DTS_E_CANTREALIZEVIRTUALBUFFERS	A set of virtual buffers cannot be realized.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0047024	-1073450972	DTS_E_PIPELINETOOCOMPLEX	The number of threads required for this pipeline is %1!d!, which is more than the system limit of %2!d!. The pipeline requires too many threads as configured. There are either too many asynchronous outputs, or EngineThreads property is set too high. Split the pipeline into multiple packages, or reduce the value of the EngineThreads property.
0xC0047028	-1073450968	DTS_E_SCHEDULERCOULDNOTCOUNTSOURCES	The Data Flow engine scheduler cannot obtain a count of the sources in the layout.
0xC0047029	-1073450967	DTS_E_SCHEDULERCOULDNOTCOUNTDESTINATIONS	The Data Flow engine scheduler cannot obtain a count of the destinations in the layout.
0xC004702A	-1073450966	DTS_E_COMPONENTVIEWISUNAVAILABLE	The component view is unavailable. Make sure the component view has been created.
0xC004702B	-1073450965	DTS_E_INCORRECTCOMPONENTVIEWID	The component view ID is incorrect. The component view may be out of synchronization. Try releasing the component view and recreating it.
0xC004702C	-1073450964	DTS_E_BUFFERNOTLOCKED	This buffer is not locked and cannot be manipulated.
0xC004702D	-1073450963	DTS_E_CANTBUILDBUFFERTYPE	The Data Flow task cannot allocate memory to build a buffer definition. The buffer definition had %1!d! columns.
0xC004702E	-1073450962	DTS_E_CANTREGISTERBUFFERTYPE	The Data Flow task cannot register a buffer type. The type had %1!d! columns and was for execution tree %2!d!.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC004702F	-1073450961	DTS_E_INVALIDUSESDISPOSITIONSVALUE	The UsesDispositions property cannot be changed from its initial value. This occurs when the XML is edited and the UsesDispositions value is modified. This value is set by the component when it is added to the package and is not allowed to change.
0xC0047030	-1073450960	DTS_E_THREADFAILEDINITIALIZE	The Data Flow task failed to initialize a required thread and cannot begin execution. The thread previously reported a specific error.
0xC0047031	-1073450959	DTS_E_THREADFAILEDCREATE	The Data Flow task failed to create a required thread and cannot begin running. The usually occurs when there is an out-of-memory state.
0xC0047032	-1073450958	DTS_E_EXECUTIONTREECYCLEADDITIONSYNCHRONOUSINPUT	The synchronous input of "%1" cannot be set to "%2" because a cycle would be created.
0xC0047033	-1073450957	DTS_E_INVALIDCUSTOMPROPERTYNAME	A custom property named "%1" is invalid because there is a stock property with that name. A custom property cannot have the same name as a stock property on the same object.
0xC0047035	-1073450955	DTS_E_BUFFERLOCKUNDERFLOW	The buffer was already unlocked.
0xC0047036	-1073450954	DTS_E_INDIVIDUALCACHEINTERFACESFAILED	%1 failed initialization and returned error code 0x%2!8.X!.
0xC0047037	-1073450953	DTS_E_INDIVIDUALRELEASEINTERFACESFAILED	%1 failed during shut down and returned error code 0x%2!8.X!. A component failed to release its interfaces.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0047038	-1073450952	DTS_E_PRIMEOUTPUTFAILED	SSIS Error Code DTS_E_PRIMEOUTPUTFAILED. The PrimeOutput method on %1 returned error code 0x%2!8.X!. The component returned a failure code when the pipeline engine called PrimeOutput(). The meaning of the failure code is defined by the component, but the error is fatal and the pipeline stopped executing. There may be error messages posted before this with more information about the failure.
0xC0047039	-1073450951	DTS_E_THREADCANCELLED	SSIS Error Code DTS_E_THREADCANCELLED. Thread "%1" received a shutdown signal and is terminating. The user requested a shutdown, or an error in another thread is causing the pipeline to shutdown. There may be error messages posted before this with more information on why the thread was cancelled.
0xC004703A	-1073450950	DTS_E_DISTRIBUTORCANTSETPROPERTY	Distributor for thread "%1" failed to initialize property "%2" on component "%3" because of error 0x%8.8X. The distributor could not initialize the component's property and cannot continue running.
0xC004703B	-1073450949	DTS_E_CANTREGISTERVIEWBUFFERTYPE	The Data Flow task cannot register a view buffer type. The type had %1!d! columns and was for input ID %2!d!.
0xC004703F	-1073450945	DTS_E_CANTCREATEEXECUTIONTREE	There is not enough memory to create an execution tree.
0xC0047040	-1073450944	DTS_E_CANTINSERTINTOHASHTABLE	There is not enough memory to insert an object into the hash table.
0xC0047041	-1073450943	DTS_E_OBJECTNOTINHASHTABLE	The object is not in the hash table.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0047043	-1073450941	DTS_E_CANTCREATECOMPONENTVIEW	Cannot create a component view because another one already exists. Only one component view can exist at a time.
0xC0047046	-1073450938	DTS_E_LAYOUTCANTSETUSAGETYPE	At input "%1" (%2!d!), the virtual input column collection does not contain a virtual input column with lineage ID %3!d!.
0xC0047047	-1073450937	DTS_E_WRONGOBJECTTYPE	The requested object has the incorrect object type.
0xC0047048	-1073450936	DTS_E_CANTCREATESPOOLFILE	The buffer manager cannot create a temporary storage file on any path in the BufferTempStoragePath property. There is an incorrect file name or no permission or the paths have been full.
0xC0047049	-1073450935	DTS_E_SEEKFAILED	The buffer manager could not seek to offset %1!d! in file "%2". The file is damaged.
0xC004704A	-1073450934	DTS_E_EXTENDFAILED	The buffer manager cannot extend the file "%1" to length %2!lu! bytes. There was insufficient disk space.
0xC004704B	-1073450933	DTS_E_FILEWRITEFAILED	The buffer manager cannot write %1!d! bytes to file "%2". There was insufficient disk space or quota.
0xC004704C	-1073450932	DTS_E_FILEREADFAILED	The buffer manager cannot read %1!d! bytes from file "%2". The file is damaged.
0xC004704D	-1073450931	DTS_E_VIRTUALNOTSEQUENTIAL	Buffer ID %1!d! supports other virtual buffers and cannot be placed into sequential mode. IDTSBuffer100.SetSequentialMode was called on a buffer that supports virtual buffers.
0xC004704E	-1073450930	DTS_E_BUFFERISREADONLY	This operation could not be performed because buffer is in read-only mode. A read-only buffer cannot be modified.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC004704F	-1073450929	DTS_E_EXECUTIONTREECYCLESETTINGID	ID %1 cannot be set to %2!d! because a cycle would be created.
0xC0047050	-1073450928	DTS_E_NOMOREBUFFERTYPES	The buffer manager ran out of memory while trying to extend the table of buffer types. This is caused by an out-of-memory condition.
0xC0047051	-1073450927	DTS_E_CANTCREATENEWTYPE	The buffer manager failed to create a new buffer type.
0xC0047053	-1073450925	DTS_E_SCHEDULERBADTREE	The Data Flow engine scheduler failed to retrieve the execution tree with index %1!d! from the layout. The scheduler received a count containing more execution trees than actually exist.
0xC0047056	-1073450922	DTS_E_CANTCREATEPRIMEOUTPUTBUFFER	The Data Flow task failed to create a buffer to call PrimeOutput for output "%3" (%4!d!) on component "%1" (%2!d!). This error usually occurs due to an out-of-memory condition.
0xC0047057	-1073450921	DTS_E_SCHEDULERTHREADMEMORY	The Data Flow engine scheduler failed to create a thread object because not enough memory is available. This is caused by an out-of-memory condition.
0xC004705A	-1073450918	DTS_E_SCHEDULEROBJECT	The Data Flow engine scheduler cannot retrieve object with ID %1!d! from the layout. The Data Flow engine scheduler previously located an object that is now no longer available.
0xC004705B	-1073450917	DTS_E_PREPARETREENODEFAILED	The Data Flow task failed to prepare buffers for the execution tree node beginning at output "%1" (%2!d!).
0xC004705C	-1073450916	DTS_E_CANTCREATEVIRTUALBUFFER	The Data Flow task cannot create a virtual buffer to prepare for execution.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC004705E	-1073450914	DTS_E_NOMOREIDS	The maximum ID has been reached. There are no more IDs available to assign to objects.
0xC004705F	-1073450913	DTS_E_ALREADYATTACHED	The %1 is already attached and cannot be attached again. Detach it and try again.
0xC0047060	-1073450912	DTS_E_OUTPUTCOLUMNNAMECONFLICT	Column name "%1" on output "%2" cannot be used because it conflicts with a column of the same name on synchronous input "%3".
0xC0047061	-1073450911	DTS_E_EOFANNOUNCEMENTFAILED	The Data Flow task cannot to create a buffer to mark the end of the rowset.
0xC0047062	-1073450910	DTS_E_USERCOMPONENTEXCEPTION	A managed user component has thrown exception "%1".
0xC0047063	-1073450909	DTS_E_SCHEDULERMEMORY	The Data Flow engine scheduler cannot allocate enough memory for the execution structures. The system was low on memory before execution started.
0xC0047064	-1073450908	DTS_E_BUFFERNOOBJECTMEMORY	An out-of-memory condition prevented the creation of the buffer object.
0xC0047065	-1073450907	DTS_E_BUFFERNOMAPMEMORY	An out-of-memory condition prevents the mapping of a buffer's lineage IDs to DTP_HCOL indexes.
0xC0047066	-1073450906	DTS_E_INDIVIDUALPUTVARIABLESFAILED	The "%1!" cannot cache the Variables collection and returned error code 0x%2!8.X.
0xC0047067	-1073450905	DTS_E_INDIVIDUALPUTCOMPONENTMETADATAFAILD	The "%1" failed to cache the component metadata object and returned error code 0x%2!8.X!.
0xC0047068	-1073450904	DTS_E_SORTEDOUTPUTHASINVALIDSORTKEYPOSITION	"%1" has a non-zero SortKeyPosition, but its value (%2!d!) is too large. It must be less than or equal to the number of columns.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC004706A	-1073450902	DTS_E_SORTEDOUTPUTHASI NVALIDSORTKEYPOSITIONS	The IsSorted property of %1 is set to TRUE, but the absolute values of the non-zero output column SortKeyPositions do not form a monotonically increasing sequence, starting at one.
0xC004706B	-1073450901	DTS_E_INDIVIDUALVALIDAT IONSTATUSFAILED	"%1" failed validation and returned validation status "%2".
0xC004706C	-1073450900	DTS_E_CANTCREATECOMP ONENT	Component "%1!" could not be created and returned error code 0x%2!8.X! "%3!". Make sure that the component is registered correctly.
0xC004706D	-1073450899	DTS_E_COMPONENTNOTRE GISTERED	The module containing "%1" is not registered or installed correctly.
0xC004706E	-1073450898	DTS_E_COMPONENTNOTFO UND	The module containing "%1" cannot be located, even though it is registered.
0xC004706F	-1073450897	DTS_E_BINARYCODENOTFO UND	The script component is configured to pre-compile the script, but binary code is not found. Please visit the IDE in Script Component Editor by clicking Design Script button to cause binary code to be generated.
0xC0047070	-1073450896	DTS_E_CANTCREATEBLOBF ILE	The buffer manager cannot create a file to spool a long object on the directories named in the BLOBTempStoragePath property. Either an incorrect file name was provided, or there are no permissions, or the paths have been full.
0xC0047071	-1073450895	DTS_E_SYNCHRONOUSIDMI SMATCH	The SynchronousInputID property on "%1" was %2!d!, and %3!d! was expected.
0xC0047072	-1073450894	DTS_E_OBJECTIDNOTFOUN D	No object exists with the ID %1!d!.
0xC0047073	-1073450893	DTS_E_OBJECTIDLOOKUPFA ILED	Unable to locate an object with ID %1!d! because of the error code 0x%2!8.X!.



HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0047074	-1073450892	DTS_E_INVALIDCODEPAGE	The code page %1!d! specified on output column "%2" (%3!d!) is not valid. Select a different code page for output column "%2".
0xC0047075	-1073450891	DTS_E_INDIVIDUALPUTEVENTINFOSFAILED	The EventInfos collection could not be cached by "%1" and returned error code 0x%2!8.8X!.
0xC0047077	-1073450889	DTS_E_DUPLICATEOUTPUTCOLUMNNAMES	The name for "%1" is a duplicate. All names must be unique.
0xC0047078	-1073450888	DTS_E_NOOUTPUTCOLUMNFORINPUTCOLUMN	There is no output column associated with input column "%1" (%2!d!).
0xC0047079	-1073450887	DTS_E_EXCLGRPNOSYNCP	"%1" has a virtual buffer extending from a root source. There is an exclusion group that is not zero with a synchronous input that is zero.
0xC004707A	-1073450886	DTS_E_ERROROUTCANTBEONSYNCHNONEXCLUSIVEOUTPUT	"%1" cannot be an error output because error outputs cannot be placed on synchronous, non-exclusive outputs.
0xC004707B	-1073450885	DTS_E_EXPREVALDIVBYZERO	A divide-by-zero error occurred. The right side operand evaluates to zero in the expression "%1".
0xC004707C	-1073450884	DTS_E_EXPREVALLITERALOVERFLOW	The literal "%1" is too large to fit into type %2. The magnitude of the literal overflows the type.
0xC004707D	-1073450883	DTS_E_EXPREVALBINARYOPERATIONNUMERICOVERFLOW	The result of the binary operation "%1" on data types %2 and %3 exceeds the maximum size for numeric types. The operand types could not be implicitly cast into a numeric (DT_NUMERIC) result without loss of precision or scale. To perform this operation, one or both operands need to be explicitly cast with a cast operator.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC004707E	-1073450882	DTS_E_EXPREVALBINARYOP OVERFLOW	The result of the binary operation "%1" exceeds the maximum size for result data type "%2". The magnitude of the result of the operation overflows the type of the result.
0xC004707F	-1073450881	DTS_E_EXPREVALFUNCTION OVERFLOW	The result of the function call "%1" is too large to fit in type "%2". The magnitude of the result of the function call overflows the type of the operand. An explicit cast to a larger type may be required.
0xC0047080	-1073450880	DTS_E_EXPREVALBINARYTYP EMISMATCH	The data types "%1" and "%2" are incompatible for binary operator "%3". The operand types could not be implicitly cast into compatible types for the operation. To perform this operation, one or both operands need to be explicitly cast with a cast operator.
0xC0047081	-1073450879	DTS_E_EXPREVALUNSUPPO RTEDBINARYTYPE	The data type "%1" cannot be used with binary operator "%2". The type of one or both of the operands is not supported for the operation. To perform this operation, one or both operands need to be explicitly cast with a cast operator.
0xC0047082	-1073450878	DTS_E_EXPREVALBINARYSIG NMISMATCH	There is a sign mismatch for the bitwise binary operator "%1" in operation "%2". Both operands for this operator must be positive or negative.
0xC0047083	-1073450877	DTS_E_EXPREVALBINARYOP ERATIONFAILED	The binary operation "%1" failed with error code 0x%2!8.X!. An internal error occurred, or an out-of-memory condition exists.
0xC0047084	-1073450876	DTS_E_EXPREVALBINARYOP ERATIONSETTYPEFAILED	Attempt to set the result type of binary operation "%1" failed with error code 0x%2!8.X!.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0047085	-1073450875	DTS_E_EXPREVALSTRINGCOMPARISONFAILED	Comparing "%1" to string "%2" failed.
0xC0047086	-1073450874	DTS_E_EXPREVALUNSUPPORTEDUNNARYTYPE	The data type "%1" cannot be used with unary operator "%2". This operand type is not supported for the operation. To perform this operation, the operand needs to be explicitly cast with a cast operator.
0xC0047087	-1073450873	DTS_E_EXPREVALUNARYOPERATIONFAILED	The unary operation "%1" failed with error code 0x%2!8.8X!. An internal error occurred, or there is an out-of-memory condition.
0xC0047088	-1073450872	DTS_E_EXPREVALUNARYOPERATIONSETTYPEFAILED	Attempt to set the result type of unary operation "%1" failed with error code 0x%2!8.8X!.
0xC0047089	-1073450871	DTS_E_EXPREVALPARAMTYPEMISMATCH	The function "%1" does not support the data type "%2" for parameter number %3!d!. The type of the parameter could not be implicitly cast into a compatible type for the function. To perform this operation, the operand needs to be explicitly cast with a cast operator.
0xC004708A	-1073450870	DTS_E_EXPREVALINVALIDFUNCTION	The function "%1" was not recognized. Either the function name is incorrect or does not exist.
0xC004708B	-1073450869	DTS_E_EXPREVALFNSTRINGSUBSTRINGINVALIDLENGTH	The length %1!d! is not valid for function "%2". The length parameter cannot be negative. Change the length parameter to zero or a positive value.
0xC004708C	-1073450868	DTS_E_EXPREVALFNSTRINGSUBSTRINGINVALIDSTARTINDEX	The start index %1!d! is not valid for function "%2". The start index value must be an integer greater than 0. Start index is one-based, not zero-based.
0xC004708E	-1073450866	DTS_E_EXPREVALCHARMAPPINGFAILED	The function "%1" cannot perform the character mapping on string "%2".

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC004708F	-1073450865	DTS_E_EXPREVALINVALIDD ATEPART	"%1" is not a valid date part for function "%2".
0xC0047090	-1073450864	DTS_E_EXPREVALINVALIDN ULLPARAM	Parameter number %1!d! of the function NULL with data type "%2" is not valid. The parameters of NULL() must be static, and cannot contain dynamic elements such as input columns.
0xC0047091	-1073450863	DTS_E_EXPREVALINVALIDN ULLPARAMTYPE	Parameter number %1!d! of the function NULL with data type "%2" is not an integer. A parameter of NULL() must be an integer or a type that can be converted to an integer.
0xC0047092	-1073450862	DTS_E_EXPREVALFUNCTION PARAMNOTSTATIC	Parameter number %1!d! of the function "%2" is not static. This parameter must be static, and cannot contain dynamic elements such as input columns.
0xC0047093	-1073450861	DTS_E_EXPREVALINVALIDC ASTPARAM	Parameter number %1!d! of the cast to data type "%2" is not valid. The parameters of cast operators must be static, and cannot contain dynamic elements such as input columns.
0xC0047094	-1073450860	DTS_E_EXPREVALINVALIDC ASTPARAMTYPE	Parameter number %1!d! of the cast to data type "%2" is not an integer. A parameter of a cast operator must be an integer or a type that can be converted to an integer.
0xC0047095	-1073450859	DTS_E_EXPREVALINVALIDC AST	Cannot cast expression "%1" from data type "%2" to data type "%3". The requested cast is not supported.
0xC0047096	-1073450858	DTS_E_EXPREVALINVALIDT OKEN	Attempt to parse the expression "%1" failed. The token "%2" at line number "%3", character number "%4" was not recognized. The expression cannot be parsed because it contains invalid elements at the location specified.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0047097	-1073450857	DTS_E_EXPREVALUNEXPECT EDPARSEERROR	An error occurred when parsing the expression "%1". The expression failed to parse for an unknown reason.
0xC0047098	-1073450856	DTS_E_EXPREVALFAILEDTOP ARSEXPRESSIONWITHHR	Attempt to parse the expression "%1" failed and returned error code 0x%2!8.8X!. The expression cannot be parsed. It might contain invalid elements or it might not be well-formed. There may also be an out-of-memory error.
0xC0047099	-1073450855	DTS_E_EXPREVALFAILEDTOP ARSEXPRESSION	The expression "%1" is not valid and cannot be parsed. The expression may contain invalid elements or it may not be well-formed.
0xC004709A	-1073450854	DTS_E_EXPREVALEXPRESSIO NEMPTY	There was no expression to compute. An attempt was made to compute or get the string of an empty expression.
0xC004709B	-1073450853	DTS_E_EXPREVALCOMPUTE FAILED	Attempt to compute the expression "%1" failed with error code 0x%2!8.8X!.
0xC004709C	-1073450852	DTS_E_EXPREVALBUILDSTRI NGFAILED	Attempt to generate a string representation of the expression failed with error code 0x%1!8.8X!. Failed when attempting to generate a displayable string that represents the expression.
0xC004709D	-1073450851	DTS_E_EXPREVALCANNOTC ONVERTRESULT	Cannot convert the expression result data type "%1" to the column data type "%2". The result of the expression should be written to an input/output column, but the data type of the expression cannot be converted to the data type of the column.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC004709E	-1073450850	DTS_E_EXPREVALCONDITIONALOPINVALIDCONDITIONTYPE	The conditional expression "%1" of the conditional operator has an invalid data type of "%2". The conditional expression of the conditional operator must return a Boolean, which is type DT_BOOL.
0xC004709F	-1073450849	DTS_E_EXPREVALCONDITIONALOPTYPEMISMATCH	The data types "%1" and "%2" are incompatible for the conditional operator. The operand types cannot be implicitly cast into compatible types for the conditional operation. To perform this operation, one or both operands need to be explicitly cast with a cast operator.
0xC00470A0	-1073450848	DTS_E_EXPREVALCONDITIONALOPSETTYPEFAILED	Attempt to set the result type of conditional operation "%1" failed with error code 0x%2!8.8X!.
0xC00470A1	-1073450847	DTS_E_BUFFERORPHANED	This buffer has been orphaned. The buffer manager has shut down, leaving an outstanding buffer and no cleanup will occur for the buffer. There is a potential for memory leaks and other problems.
0xC00470A2	-1073450846	DTS_E_EXPREVALINPUTCOLUMNNAMENOTFOUND	Attempt to find the input column named "%1" failed with error code 0x%2!8.8X!. The input column specified was not found in the input column collection.
0xC00470A3	-1073450845	DTS_E_EXPREVALINPUTCOLUMNIDNOTFOUND	Attempt to find the input column with lineage ID %1!d! failed with error code 0x%2!8.8X!. The input column was not found in the input column collection.
0xC00470A4	-1073450844	DTS_E_EXPREVALNOINPUTCOLUMNCOLLECTIONFORCOLUMNNAME	The expression contains unrecognized token "%1". If "%1" is a variable, it should be expressed as "@%1". The specified token is not valid. If the token is intended to be a variable name, it should be prefixed with the @ symbol.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC00470A5	-1073450843	DTS_E_EXPREVALNOINPUTC OLUMNCOLLECTIONFORC OLUMNID	The expression contains unrecognized token "%1!d!".
0xC00470A6	-1073450842	DTS_E_EXPREVALVARIABLE NOTFOUND	The variable "%1" was not found in the Variables collection. The variable might not exist in the correct scope.
0xC00470A7	-1073450841	DTS_E_EXPREVALINVALIDT OKENSTATE	Attempt to parse the expression "%1" failed. The expression might contain an invalid token, an incomplete token, or an invalid element. It might not be well-formed, or might be missing part of a required element such as a parenthesis.
0xC00470A8	-1073450840	DTS_E_BLANKOUTPUTCOLU MNNAME	The name for "%1" is blank, and names cannot be blank.
0xC00470A9	-1073450839	DTS_E_HASSIDEEFFECTSWIT HSYNCP	The "%1" has the HasSideEffects property set to TRUE, but "%1" is synchronous and cannot have side effects. Set the HasSideEffects property to FALSE.
0xC00470AA	-1073450838	DTS_E_EXPREVALINVALIDC ASTCODEPAGE	The value, %1!d!, specified for the code page parameter of the cast to data type "%2", is not valid. The code page is not installed on the machine.
0xC00470AB	-1073450837	DTS_E_EXPREVALINVALIDC ASTPRECISION	The value %1!d! specified for the precision parameter of the cast to data type "%2" is not valid. Precision must be in the range %3!d! to %4!d! and the precision value is out of range for the type cast.
0xC00470AC	-1073450836	DTS_E_EXPREVALINVALIDC ASTSCALE	The value %1!d! specified for the scale parameter of the cast to data type "%2" is not valid. The scale must be in the range %3!d! to %4!d! and the scale value is out of range for the type cast. Scale must not exceed precision and must be positive.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC00470AD	-1073450835	DTS_E_NONSORTEDOUTPUT THASSORTKEYPOSITIONS	The IsSorted property for "%1" is false, but %2!lu! of its output columns' SortKeyPositions are non-zero.
0xC00470AF	-1073450833	DTS_E_EXPREVALCONDITIONAL OPCODEPAGEMISMATCH	The code pages must match for operands of conditional operation "%1" for type %2. The code page of the left operand does not match the code page of the right operand. For the conditional operator on the specified type, the code pages must be the same.
0xC00470B1	-1073450831	DTS_E_REFERENCEDMETADATA BADCOUNT	Input "%1" (%2!d!) references input "%3" (%4!d!), but they do not have the same number of columns. Input %5!d! has %6!d! columns, while input %7!d! has %8!d! columns.
0xC00470B2	-1073450830	DTS_E_OBJECTLINEAGEID NOTFOUND	No object exists with a lineage ID of %1!d!.
0xC00470B3	-1073450829	DTS_E_FILENAMEOUTPUTC OLUMNNOTFOUND	The output column for the file name cannot be found.
0xC00470B4	-1073450828	DTS_E_FILENAMEOUTPUTC OLUMNINVALIDDATATYPE	The output column for the file name is not a null-terminated Unicode character string, which is data type DT_WSTR.
0xC00470B5	-1073450827	DTS_E_DISTRIBUTORADDFAI LED	A distributor failed to give a buffer to thread "%1" because of error 0x%2!8.X!. The target thread is probably shutting down.
0xC00470B6	-1073450826	DTS_E_LOCALENOTINSTALL ED	The LocaleID %1!d! is not installed on this system.
0xC00470B7	-1073450825	DTS_E_EXPREVALILLEGALHE XESCAPEINSTRINGLITERAL	The string literal "%1" contains an illegal hexadecimal escape sequence of "\x%2". The escape sequence is not supported in string literals in the expression evaluator. The hexadecimal escape sequences must be of the form \xhhhh where h is a valid hexadecimal digit.



HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC00470B8	-1073450824	DTS_E_EXPREVALILLEGALES CAPEINSTRINGLITERAL	The string literal "%1" contains an illegal escape sequence of "%2!c!". The escape sequence is not supported in string literals in the expression evaluator. If a backslash is needed in the string, use a double backslash, "\\".
0xC00470B9	-1073450823	DTS_E_NOOUTPUTCOLUMNS	"%1" contains no output columns. An asynchronous output must contain output columns.
0xC00470BA	-1073450822	DTS_E_LOBDATATYPENOTSUPPORTED	The "%1" has a long object data type of DT_TEXT, DT_NTEXT, or DT_IMAGE, which is not supported.
0xC00470BB	-1073450821	DTS_E_OUTPUTWITHMULTIPLEERRORS	Output ID %1!d! was given multiple error output configurations. First %2!d! and %3!d!, then %4!d! and %5!d!.
0xC00470BC	-1073450820	DTS_E_FAILEDDURINGOLEDBDATATYPECONVERSIONCHECK	The OLE DB provider failed during the data type conversion verification for "%1".
0xC00470BD	-1073450819	DTS_E_BUFFERISEOR	This buffer represents the end of the rowset and its row count cannot be altered. An attempt was made to call AddRow or RemoveRow on a buffer that has the end of rowset flag.
0xC00470BE	-1073450818	DTS_E_EXPREVALUNSUPPORTEDTYPE	The data type "%1" is not supported in an expression. The specified type is not supported or is not valid.
0xC00470BF	-1073450817	DTS_E_PRIMEOUTPUTNOERROR	The PrimeOutput method on "%1" returned success, but did not report an end of the rowset. There is an error in the component. It should have reported an end-of-row. The pipeline will shut down execution to avoid unpredictable results.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC00470C0	-1073450816	DTS_E_EXPREVALDATACON VERSIONOVERFLOW	An overflow occurred while converting from data type "%1" to data type "%2". The source type is too large for the destination type.
0xC00470C1	-1073450815	DTS_E_EXPREVALDATACON VERSIONNOTSUPPORTED	Conversion from data type "%1" to data type "%2" is unsupported. The source type cannot be converted to the destination type.
0xC00470C2	-1073450814	DTS_E_EXPREVALDATACON VERSIONFAILED	Error code 0x%1!8.X! occurred attempting to convert from data type %2 to data type %3.
0xC00470C3	-1073450813	DTS_E_EXPREVALCONDITIO NALOPERATIONFAILED	The conditional operation "%1" failed with error code 0x%2!8.X!. There was an internal error or an out-of-memory error.
0xC00470C4	-1073450812	DTS_E_EXPREVALCASTFAILE D	Casting expression "%1" from data type "%2" to data type "%3" failed with error code 0x%4!8.X!.
0xC00470C5	-1073450811	DTS_E_EXPREVALFUNCTION COMPUTEFAILED	Evaluating function "%1" failed with error code 0x%2!8.X!.
0xC00470C6	-1073450810	DTS_E_EXPREVALFUNCTION CONVERTPARAMTOMEMBE RFAILED	Parameter number %1!d! of the function "%2" cannot be converted to a static value.
0xC00470C7	-1073450809	DTS_E_REDIRECTROWUNAV AILABLEWITHFASTLOADAN DZEROMAXINSERTCOMMIT SIZE	The error row disposition on "%1" cannot be set to redirect the row when the fast load option is turned on, and the maximum insert commit size is set to zero.
0xC00470CE	-1073450802	DTS_E_EXPREVALBINARYOP ERATORCODEPAGEMISMAT CH	The code pages for operands of binary operator "%1" for type "%2" must match. Currently, the code page of the left operand does not match the code page of the right operand. For the specified binary operator on the specified type, the code pages must be the same.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC00470CF	-1073450801	DTS_E_EXPREVALVARIABLEC OMPUTEFAILED	Retrieving the value of Variable "%1" failed with error code 0x%2!8.8X!.
0xC00470D0	-1073450800	DTS_E_EXPREVALVARIABLET YPENOTSUPPORTED	The data type of variable "%1" is not supported in an expression.
0xC00470D1	-1073450799	DTS_E_EXPREVALCASTCODE PAGEMISMATCH	Unable to cast expression "%1" from data type "%2" to data type "%3" because the code page of the value being cast (%4!d!) does not match the requested result code page (%5!d!). The code page of the source must match the code page requested for the destination.
0xC00470D2	-1073450798	DTS_E_BUFFERSIZEOUTOFR ANGE	The default buffer size must be between %1!d! and %2!d! bytes. An attempt was made to set the DefaultBufferSize property to a value that is too small or too large.
0xC00470D3	-1073450797	DTS_E_BUFFERMAXROWSIZ EOUTOFRANGE	The default buffer maximum rows must be larger than %1!d! rows. An attempt was made to set the DefaultBufferMaxRows property to a value that is too small.
0xC00470D4	-1073450796	DTS_E_EXTERNALCOLUMN METADATACODEPAGEMIS MATCH	The code page on %1 is %2!d! and is required to be %3!d!.
0xC00470D5	-1073450795	DTS_E_THREADCOUNTOUT OFRANGE	Failed to assign %3!d! to the EngineThreads property of the Data Flow task. The value must be between %1!d! and %2!d!.
0xC00470D6	-1073450794	DTS_E_EXPREVALINVALIDT OKENSINGLEQUOTE	Parsing the expression "%1" failed. The single quotation mark at line number "%2", character number "%3", was not expected.
0xC00470D7	-1073450793	DTS_E_EXPREVALINVALIDT OKENSINGLEEQUAL	Parsing the expression "%1" failed. The equal sign (=) at line number "%2", character number "%3", was not expected. A double equals sign (==) may be required at the location specified.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC00470DA	-1073450790	DTS_E_INDIVIDUALPUTRETRACTERFAILED	Component "%1" failed to cache the runtime object reference tracker collection and returned error code 0x%2!8.X!.
0xC00470DB	-1073450789	DTS_E_EXPREVALAMBIGUOUSINPUTCOLUMNNAME	There are multiple input columns with the name "%1". The desired input column must be specified uniquely as [Component Name].[%2] or referenced by lineage ID. Currently, the input column specified exists on more than one component.
0xC00470DC	-1073450788	DTS_E_EXPREVALDOTTEDINPUTCOLUMNNAME NOT FOUND	Locating the input column named "[%1].[%2]" failed with error code 0x%3!8.X!. The input column was not found in the input column collection.
0xC00470DD	-1073450787	DTS_E_EXPREVALAMBIGUOUSVARIABLENAME	There are multiple variables with the name "%1". The desired variable must be specified uniquely as @[Namespace::%2]. The variable exists in more than one namespace.
0xC00470DE	-1073450786	DTS_E_REDUCTIONFAILED	The Data Flow engine scheduler failed to reduce the execution plan for the pipeline. Set the OptimizedMode property to false.
0xC00470DF	-1073450785	DTS_E_EXPREVALSQRTINVALIDPARAM	The function SQRT cannot operate on negative values, and a negative value was passed to the SQRT function.
0xC00470E0	-1073450784	DTS_E_EXPREVALLNINVALIDPARAM	The function LN cannot operate on zero or negative values, and a zero or negative value was passed to the LN function.
0xC00470E1	-1073450783	DTS_E_EXPREVALLOGINVALIDPARAM	The function LOG cannot operate on zero or negative values, and a zero or negative value was passed to the LOG function.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC00470E2	-1073450782	DTS_E_EXPREVALPOWERINV ALIDPARAM	The parameters passed to the function POWER cannot be evaluated and yield an indeterminate result.
0xC00470E3	-1073450781	DTS_E_NOCANCELEVENT	The runtime cannot provide a cancel event because of error 0x%1!8.8X!.
0xC00470E4	-1073450780	DTS_E_CANCELRECEIVED	The pipeline received a request to cancel and is shutting down.
0xC00470E5	-1073450779	DTS_E_EXPREVALUNARYOP OVERFLOW	The result of the unary minus (negation) operation "%1" exceeds the maximum size for result data type "%2". The magnitude of the result of the operation overflows the type of the result.
0xC00470E6	-1073450778	DTS_E_EXPREVALPLACEHOL DERINEXPRESSION	The placeholder "%1" was found in an expression. This must be replaced with an actual parameter or operand.
0xC00470E7	-1073450777	DTS_E_EXPREVALFNRIGHTIN VALIDLENGTH	The length %1!d! specified for function "%2" is negative, and is not valid. The length parameter must be positive.
0xC00470E8	-1073450776	DTS_E_EXPREVALFNREPLICA TEINVALIDREPEATCOUNT	The repeat count %1!d! is negative and is not valid for function "%2". The repeat count parameter cannot be negative.
0xC00470EA	-1073450774	DTS_E_EXPREVALVARIABLEC OULDNOTBEREAD	Reading the variable "%1" failed with error code 0x%2!8.8X!.
0xC00470EC	-1073450772	DTS_E_EXPREVALBINARYOP DTSTRNOTSUPPORTED	For operands of a binary operation, the data type DT_STR is supported only for input columns and cast operations. The expression "%1" has a DT_STR operand that is not an input column or the result of a cast, and cannot be used in a binary operation. To perform this operation, the operand needs to be explicitly cast with a cast operator.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC00470ED	-1073450771	DTS_E_EXPREVALCONDITIO NALOPDTSTRNOTSUPPORT ED	For operands of the conditional operator, the data type DT_STR is supported only for input columns and cast operations. The expression "%1" has a DT_STR operand that is not an input column or the result of a cast, and cannot be used with the conditional operation. To perform this operation, the operand needs to be explicitly cast with a cast operator.
0xC00470EE	-1073450770	DTS_E_EXPREVALFNFINDSTR INGINVALIDOCCURRENCEC OUNT	The occurrence count %1!d! is not valid for function "%2". This parameter must be greater than zero.
0xC00470EF	-1073450769	DTS_E_INDIVIDUALPUTLOG ENTRYINFOS	"%1" failed to cache the LogEntryInfos collection and returned error code 0x%2!8.X!.
0xC00470F0	-1073450768	DTS_E_EXPREVALINVALIDD ATEPARTNODE	The date part parameter specified for function "%1" is not valid. It must be a static string. The date part parameter cannot contain dynamic elements, such as input columns, and must be of type DT_WSTR.
0xC00470F1	-1073450767	DTS_E_EXPREVALINVALIDC ASTLENGTH	The value %1!d! specified for the length parameter of the cast to data type %2 is negative and not valid. The length must be positive.
0xC00470F2	-1073450766	DTS_E_EXPREVALINVALIDN ULLCODEPAGE	The value %1!d! specified for the code page parameter of the NULL function with data type "%2" is not valid. The code page is not installed on the computer.
0xC00470F3	-1073450765	DTS_E_EXPREVALINVALIDN ULLPRECISION	The value %1!d! specified for the precision parameter of the NULL function with data type "%2" is out of range. Precision must be in the range %3!d! to %4!d!.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC00470F4	-1073450764	DTS_E_EXPREVALINVALIDN ULLSCALE	The value %1!d! specified for the scale parameter of the NULL function with data type %2 is out of range. Scale must be in the range %3!d! to %4!d!. Scale must not exceed precision and must not be negative.
0xC00470F5	-1073450763	DTS_E_EXPREVALINVALIDN ULLLENGTH	The value %1!d! specified for the length parameter of the "NULL" function with data type %2 is negative and not valid. The length must be positive.
0xC00470F6	-1073450762	DTS_E_NEGATIVESNOTALLO WED	The %1 can't be assigned a negative value.
0xC00470F7	-1073450761	DTS_E_FASTPARSENOTALLO WED	The "%1" custom property for "%2" cannot be set to true. The column data type must be one of the following: DT_I1, DT_I2, DT_I4, DT_I8, DT_UI1, DT_UI2, DT_UI4, DT_UI8, DT_DBTIMESTAMP, DT_DBTIMESTAMP2, DT_DBTIMESTAMPOFFSET, DT_DATE, DT_DBDATE, DT_DBTIME, DT_DBTIME2, or DT_FILETIME.
0xC00470F8	-1073450760	DTS_E_CANNOTREATTACHP ATH	The "%1" cannot be reattached. Delete the path, add a new one, and attach it.
0xC00470F9	-1073450759	DTS_E_EXPREVALINVALIDN UMBEROFPARAMSPLURALS INGULAR	The function "%1" requires %2!d! parameters, not %3!d! parameter. The function name was recognized, but the number of parameters is not valid.
0xC00470FA	-1073450758	DTS_E_EXPREVALINVALIDN UMBEROFPARAMSSINGULA RPLURAL	The function "%1" requires %2!d! parameter, not %3!d! parameters. The function name was recognized, but the number of parameters is not valid.
0xC00470FB	-1073450757	DTS_E_EXPREVALINVALIDN UMBEROFPARAMSPLURALP LURAL	The function "%1" requires %2!d! parameters, not %3!d! parameters. The function name was recognized, but the number of parameters is not valid.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC00470FC	-1073450756	DTS_E_EXPREVALFAILEDTOP ARSEXPRESSIIONOUTOFME MORY	Attempt to parse the expression "%1" failed because there was an out-of-memory error.
0xC00470FD	-1073450755	DTS_E_INDIVIDUALCHECKP RODUCTLEVELFAILED	The %1 failed to be able to perform its required product level check and returned error code 0x%2!8.8X!.
0xC00470FE	-1073450754	DTS_E_PRODUCTLEVELTOL OW	SSIS Error Code DTS_E_PRODUCTLEVELTOL OW. The %1 cannot run on installed %2 of Integration Services. It requires %3 or higher.
0xC00470FF	-1073450753	DTS_E_EXPREVALSTRINGLIT ERALTOOLONG	A string literal in the expression exceeds the maximum allowed length of %1!d! characters.
0xC0047100	-1073450752	DTS_E_EXPREVALSTRINGVA RIABLETOOLONG	The variable %1 contains a string that exceeds the maximum allowed length of %2!d! characters.
0xC0047101	-1073450751	DTS_E_COMPONENT_NOIN TERFACE	The %1 was found, but it does not support a required Integration Services interface (IDTSRuntimeComponent100). Obtain an updated version of this component from the component provider.
0xC0048000	-1073446912	DTS_E_CANNOTOPENREGIS TRYKEY	The registry key "%1" cannot be opened.
0xC0048001	-1073446911	DTS_E_INVALIDCOMPONEN TFILENAME	Cannot get the file name for the component with a CLSID of "%1". Verify that the component is registered properly or that the CLSID provided is correct.
0xC0048002	-1073446910	DTS_E_UNKNOWNCOMPO NENTHASINVALIDCLSID	The CLSID for one of the components is not valid. Verify that all the components in the pipeline have valid CLSIDs.
0xC0048003	-1073446909	DTS_E_COMPONENTHASIN VALIDCLSID	The CLSID for one of the components with ID %1!d! is not valid.
0xC0048004	-1073446908	DTS_E_INVALIDINDEX	The index is not valid.



HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0048005	-1073446907	DTS_E_CANNOTACCESSDTSAPPLICATIONOBJECT	The Application object cannot be accessed. Verify that SSIS is correctly installed.
0xC0048006	-1073446906	DTS_E_ERROROCCURREDWHILERETRIEVINGFILENAME	Retrieving the file name for a component failed with error code 0x%1!8.8X!.
0xC0048007	-1073446905	DTS_E_CANNOTRETRIEVEPROPERTYFORCOMPONENT	Cannot retrieve property "%1" from component with ID %2!d!.
0xC0048008	-1073446904	DTS_E_DUPLICATEIDFOUND	Attempting to use ID %1!d! more than once in the Data Flow Task.
0xC0048009	-1073446903	DTS_E_CANNOTRETRIEVEBYLINEAGE	Cannot retrieve an item by lineage ID from a collection that does not contain columns.
0xC004800B	-1073446901	DTS_E_CANNOTMAPRUNTIMECONNECTIONMANAGER	Cannot find the connection manager with ID "%1" in the connection manager collection due to error code 0x%2!8.8X!. That connection manager is needed by "%3" in the connection manager collection of "%4". Verify that a connection manager in the connection manager collection, Connections, has been created with that ID.
0xC004800E	-1073446898	DTS_E_INPUTNOTKNOWN	Thread "%1" received a buffer for input %2!d!, but this thread is not responsible for that input. An error occurred, causing the Data Flow engine scheduler to build a bad execution plan.
0xC004800F	-1073446897	DTS_E_GETRTINTERFACEFAILED	The component "%1" (%2!d!) cannot provide an IDTSRuntimeComponent100 interface.
0xC0048011	-1073446895	DTS_E_CANTGIVEAWAYBUFFER	The Data Flow task engine attempted to copy a buffer to assign another thread, but failed.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0048012	-1073446894	DTS_E_CANTCREATEVIEWBUFFER	The Data Flow task engine failed to create a view buffer of type %1!d! over type %2!d! for buffer %3!d.
0xC0048013	-1073446893	DTS_E_UNUSABLETEMPORARYPATH	The buffer manager could not create a temporary file on the path "%1". The path will not be considered for temporary storage again.
0xC0048014	-1073446892	DTS_E_DIRECTTONONERROWOUTPUT	The buffer manager attempted to push an error row to an output that was not registered as an error output. There was a call to DirectErrorRow on an output that does not have the IsErrorOut property set to TRUE.
0xC0048015	-1073446891	DTS_E_BUFFERISPRIVATE	A call was made to a buffer method on a private buffer and private buffers do not support this operation.
0xC0048016	-1073446890	DTS_E_BUFFERISFLAT	Private mode buffers do not support this operation.
0xC0048017	-1073446889	DTS_E_BUFFERISPRIMEOUTPUT	This operation cannot be called on a buffer passed to PrimeOutput. A call was made to a buffer method during PrimeOutput, but that call is not allowed during PrimeOutput.
0xC0048018	-1073446888	DTS_E_BUFFERISPROCESSINPUT	This operation cannot be called on a buffer passed to ProcessInput. A call was made to a buffer method during ProcessInput, but that call is not allowed during ProcessInput.
0xC0048019	-1073446887	DTS_E_BUFFERGETTEMPFILENAME	The buffer manager could not get a temporary file name.
0xC004801A	-1073446886	DTS_E_REFERENCECOLUMNTOOWIDE	The code encountered a column that was too wide.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC004801B	-1073446885	DTS_E_CANNOTGETRUNTIMECONNECTIONMANAGERID	Cannot get the ID of the runtime connection manager specified by "%1" in the connection manager collection, Connections, of "%2" due to error code 0x%3!8.X!. Verify that the ConnectionManager.ID property of the runtime connection object has been set for the component.
0xC004801C	-1073446884	DTS_E_EMPTYRUNTIMECONNECTIONMANAGERID	The "%1" in the connection manager collection, Connections, of "%2" does not have a value for the ID property. Verify that the ConnectionManager.ID property of the runtime connection object has been set for the component.
0xC004801D	-1073446883	DTS_E_METADATAREADONLY	Metadata cannot be changed during execution.
0xC004801F	-1073446881	DTS_E_UPGRADEFAILED	The component metadata for "%1" could not be upgraded to the newer version of the component. The PerformUpgrade method failed.
0xC0048020	-1073446880	DTS_E_COMPONENTVERSIONMISMATCH	The version of %1 is not compatible with this version of the DataFlow.
0xC0048021	-1073446879	DTS_E_ERRORCOMPONENT	The component is missing, not registered, not upgradeable, or missing required interfaces. The contact information for this component is "%1".
0xC0048022	-1073446878	DTS_E_BUFFERISNOTPRIMEOUTPUT	The method was called on the wrong buffer. Buffers that are not used for component output do not support this operation.
0xC0049014	-1073442796	DTS_E_EXPREVALSTATIC_COMPUTATIONFAILED	An error occurred during computation of the expression.
0xC0049030	-1073442768	DTS_E_EXPREVALSTATIC_DIVBYZERO	Division by zero occurred in the expression.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0049031	-1073442767	DTS_E_EXPREVALSTATIC_LITERALOVERFLOW	The magnitude of the literal value was too big to fit in the type requested.
0xC0049032	-1073442766	DTS_E_EXPREVALSTATIC_BINARYOPNUMERICOVERFLOW	The result of a binary operation was too big for the maximum size for numeric types. The operand types could not be implicitly cast into a numeric (DT_NUMERIC) result without loss of precision or scale. To perform this operation, one or both operands need to be explicitly cast with a cast operator.
0xC0049033	-1073442765	DTS_E_EXPREVALSTATIC_BINARYOPOVERFLOW	The magnitude of the result of a binary operation overflows the maximum size for result data type.
0xC0049034	-1073442764	DTS_E_EXPREVALSTATIC_FUNCTIONOVERFLOW	The magnitude of the result of a function call was too big to fit in the result type, and overflowed the type of the operand. An explicit cast to a larger type may be required.
0xC0049035	-1073442763	DTS_E_EXPREVALSTATIC_BINARYTYPEMISMATCH	Incompatible data types were used with a binary operator. The operand types could not be implicitly cast into compatible types for the operation. To perform this operation, one or both operands need to be explicitly cast with a cast operator.
0xC0049036	-1073442762	DTS_E_EXPREVALSTATIC_UNSUPPORTEDBINARYTYPE	An unsupported data type was used with a binary operator. The type of one, or both, of the operands is not supported for the operation. To perform this operation, one or both operands need to be explicitly cast with a cast operator.
0xC0049037	-1073442761	DTS_E_EXPREVALSTATIC_BINARYSIGNMISMATCH	There is a sign mismatch for the bitwise binary operator. The operands for this operator must be both positive or both negative.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0049038	-1073442760	DTS_E_EXPREVALSTATIC_BINARYOPERATIONFAILED	A binary operation failed. There was an out-of-memory condition, or an internal error occurred.
0xC0049039	-1073442759	DTS_E_EXPREVALSTATIC_BINARYOPERATIONSETTYPEFAILED	Setting the result type of a binary operation failed.
0xC004903A	-1073442758	DTS_E_EXPREVALSTATIC_STRINGCOMPARISONFAILED	Cannot compare two strings.
0xC004903B	-1073442757	DTS_E_EXPREVALSTATIC_UNSUPPORTEDUNNARYTYPE	An unsupported data type is used with a unary operator. The operand type is not supported for the operation. To perform this operation, the operand needs to be explicitly cast with a cast operator.
0xC004903C	-1073442756	DTS_E_EXPREVALSTATIC_UNARYOPERATIONFAILED	A unary operation failed. An out-of-memory condition occurred, or there was an internal error.
0xC004903D	-1073442755	DTS_E_EXPREVALSTATIC_UNARYOPERATIONSETTYPEFAILED	Setting the result type of a unary operation failed.
0xC004903E	-1073442754	DTS_E_EXPREVALSTATIC_PARAMETERYPEMISMATCH	A function has a parameter with an unsupported data type. The type of the parameter cannot be implicitly cast into a compatible type for the function. To perform this operation, the operand needs to be explicitly cast with a cast operator.
0xC004903F	-1073442753	DTS_E_EXPREVALSTATIC_INVALIDFUNCTION	An invalid function name appeared in the expression. Verify that the function name is correct and does exist.
0xC0049040	-1073442752	DTS_E_EXPREVALSTATIC_FUNCTIONSUBSTRINGINVALIDLENGTH	The length parameter was not valid for function SUBSTRING. The length parameter cannot be negative.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0049041	-1073442751	DTS_E_EXPREVALSTATIC_FN SUBSTRINGINVALIDSTARTIN DEX	The start index was not valid for function SUBSTRING. The start index value must be an integer greater than zero. The start index is 1-based, not 0-based.
0xC0049042	-1073442750	DTS_E_EXPREVALSTATIC_IN VALIDNUMBEROFPARAMS	An incorrect number of parameters was given to a function. The function name was recognized, but the number of parameters was not correct.
0xC0049043	-1073442749	DTS_E_EXPREVALSTATIC_CH ARMAPPINGFAILED	A character mapping function failed.
0xC0049044	-1073442748	DTS_E_EXPREVALSTATIC_IN VALIDDATEPART	An unrecognized date part parameter was specified for a date function.
0xC0049045	-1073442747	DTS_E_EXPREVALSTATIC_IN VALIDNULLPARAM	An invalid parameter was given for function NULL. The parameters of NULL must be static, and cannot contain dynamic elements such as input columns.
0xC0049046	-1073442746	DTS_E_EXPREVALSTATIC_IN VALIDNULLPARAMTYPE	An invalid parameter was given for function NULL. A parameter of NULL must be an integer, or a type that can be converted to an integer.
0xC0049047	-1073442745	DTS_E_EXPREVALSTATIC_FU NCTIONPARAMNOTSTATIC	An invalid parameter was given for a function. This parameter must be static and cannot contain dynamic elements such as input columns.
0xC0049048	-1073442744	DTS_E_EXPREVALSTATIC_IN VALIDCASTPARAM	An invalid parameter was given for a cast operation. Parameters of cast operators must be static, and cannot contain dynamic elements such as input columns.
0xC0049049	-1073442743	DTS_E_EXPREVALSTATIC_IN VALIDCASTPARAMTYPE	An invalid parameter was given for a cast operation. A parameter of a cast operator must be an integer, or a type that can be converted to an integer.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC004904A	-1073442742	DTS_E_EXPREVALSTATIC_IN VALIDCAST	The expression contained an unsupported type cast.
0xC004904B	-1073442741	DTS_E_EXPREVALSTATIC_IN VALIDTOKEN	The expression contained a token that was not recognized. The expression could not be parsed because it contains invalid elements.
0xC004904C	-1073442740	DTS_E_EXPREVALSTATIC_FAI LEDTOPARSEEXPRESSION	The expression is not valid and could not be parsed. It might contain invalid elements, or it might not be well-formed.
0xC004904D	-1073442739	DTS_E_EXPREVALSTATIC_UN ARYOPOVERFLOW	The result of a unary minus (negation) operation overflowed the maximum size for result data type. The magnitude of the result of the operation overflows the type of the result.
0xC004904E	-1073442738	DTS_E_EXPREVALSTATIC_CO MPUTEFAILED	Attempt to compute the expression failed.
0xC004904F	-1073442737	DTS_E_EXPREVALSTATIC_BU ILDSTRINGFAILED	Attempt to generate a string representation of the expression failed.
0xC0049050	-1073442736	DTS_E_EXPREVALSTATIC_CA NNOTCONVERTRESULT	Cannot convert the expression result data type to the column data type. The result of the expression should be written to an input/output column, but the data type of the expression cannot be converted to the data type of the column.
0xC0049051	-1073442735	DTS_E_EXPREVALSTATIC_CO NDITIONALOPINVALIDCON DITIONTYPE	The conditional expression of the conditional operator has invalid data type. The conditional expression must be of type DT_BOOL.
0xC0049052	-1073442734	DTS_E_EXPREVALSTATIC_CO NDITIONALOPTYPEMISMAT CH	The data types of the operands of the conditional operator were incompatible. The operand types could not be implicitly cast into compatible types for the conditional operation. To perform this operation, one or both operands need to be explicitly cast with a cast operator.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0049053	-1073442733	DTS_E_EXPREVALSTATIC_CONDITIONALOPSETTYPEFAILED	Setting the result type of a conditional operation failed.
0xC0049054	-1073442732	DTS_E_EXPREVALSTATIC_INPUTCOLUMNNAME NOT FOUND	The input column specified was not found in the input column collection.
0xC0049055	-1073442731	DTS_E_EXPREVALSTATIC_INPUTCOLUMNID NOT FOUND	Attempt to find an input column by lineage ID failed. The input column was not found in the input column collection.
0xC0049056	-1073442730	DTS_E_EXPREVALSTATIC_NO_INPUTCOLUMN COLLECTION	The expression contains an unrecognized token that appears to be an input column reference, but the input column collection is not available to process input columns. The input column collection has not been provided to the expression evaluator, but an input column was included in the expression.
0xC0049057	-1073442729	DTS_E_EXPREVALSTATIC_VARIABLE NOT FOUND	A variable specified was not found in the collection. It might not exist in the correct scope. Verify that the variable exists and that the scope is correct.
0xC0049058	-1073442728	DTS_E_EXPREVALSTATIC_INVALIDTOKEN STATE	Attempt to parse the expression failed. The expression contains an invalid or incomplete token. It may contain invalid elements, be missing part of a required element such as closing parentheses, or may not be well formed.
0xC0049059	-1073442727	DTS_E_EXPREVALSTATIC_INVALIDCASTCODEPAGE	The value specified for the code page parameter of the cast to data type DT_STR or DT_TEXT is not valid. The specified code page is not installed on the computer.
0xC004905A	-1073442726	DTS_E_EXPREVALSTATIC_INVALIDCASTPRECISION	The value specified for the precision parameter of a cast operation is out of range for the type cast.



HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC004905B	-1073442725	DTS_E_EXPREVALSTATIC_INVALIDCASTSCALE	The value specified for the scale parameter of a cast operation is out of range for the type cast. Scale must not exceed precision and must not be negative.
0xC004905C	-1073442724	DTS_E_EXPREVALSTATIC_CONDITIONALOPCODEPAGE_MISMATCH	The code pages do not match in a conditional operation. The code page of the left operand does not match the code page of the right operand. For the conditional operator of that type, the code pages must be the same.
0xC004905D	-1073442723	DTS_E_EXPREVALSTATIC_ILLEGALHEXESCAPEINSTRING_LITERAL	A string literal contains an illegal hexadecimal escape sequence. The escape sequence is not supported in string literals in the expression evaluator. Hexadecimal escape sequences must be of the form \xhhh where h is a valid hexadecimal digit.
0xC004905E	-1073442722	DTS_E_EXPREVALSTATIC_ILLEGALESCAPEINSTRING_LITERAL	The string literal contains an illegal escape sequence. The escape sequence is not supported in string literals in the expression evaluator. If a backslash is needed in the string, format it as a double backslash, "\\".
0xC004905F	-1073442721	DTS_E_EXPREVALSTATIC_UNSUPPORTEDTYPE	An unsupported or unrecognized data type was used in the expression.
0xC0049060	-1073442720	DTS_E_EXPREVALSTATIC_DATA_CONVERSION_OVERFLOW	An overflow occurred while converting between data types. The source type is too large to fit in the destination type.
0xC0049061	-1073442719	DTS_E_EXPREVALSTATIC_DATA_CONVERSION_NOT_SUPPORTED	The expression contains an unsupported data type conversion. The source type cannot be converted to the destination type.
0xC0049062	-1073442718	DTS_E_EXPREVALSTATIC_DATA_CONVERSION_FAILED	An error occurred while attempting to perform data conversion. The source type could not be converted to the destination type.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0049063	-1073442717	DTS_E_EXPREVALSTATIC_CONDITIONALOPERATIONFAILED	The conditional operation failed.
0xC0049064	-1073442716	DTS_E_EXPREVALSTATIC_CASTFAILED	An error occurred while attempting to perform a type cast.
0xC0049065	-1073442715	DTS_E_EXPREVALFAILEDTOCONVERTSTRCOLUMNNTOWSTR	Converting "%1" from type DT_STR to type DT_WSTR failed with error code 0x%2!8.X!. An error occurred while performing the implicit conversion on the input column.
0xC0049066	-1073442714	DTS_E_EXPREVALSTATIC_FAILEDTOCONVERTSTRCOLUMNNTOWSTR	Converting an input column from type DT_STR to type DT_WSTR failed. An error occurred while performing the implicit conversion on the input column.
0xC0049067	-1073442713	DTS_E_EXPREVALSTATIC_FUNCTIONCOMPUTEFAILED	An error occurred while evaluating the function.
0xC0049068	-1073442712	DTS_E_EXPREVALSTATIC_FUNCTIONCONVERTPARAMTOMEMBERFAILED	A function parameter cannot be converted to a static value. The parameter must be static and cannot contain dynamic elements such as input columns.
0xC0049088	-1073442680	DTS_E_EXPREVALSTATIC_FUNCTIONRIGHTINVALIDLENGTH	The length parameter is not valid for function RIGHT. The length parameter cannot be negative.
0xC0049089	-1073442679	DTS_E_EXPREVALSTATIC_FUNCTIONREPLICATEINVALIDREPEATCOUNT	The repeat count parameter is not valid for function REPLICATE. This parameter cannot be negative.
0xC0049096	-1073442666	DTS_E_EXPREVALSTATIC_BINARYOPERATORCODEPAGE MISMATCH	The code pages do not match in a binary operation. The code page of the left operand does not match the code page of the right operand. For this binary operation, the code pages must be the same.
0xC0049097	-1073442665	DTS_E_EXPREVALSTATIC_VARIABLECOMPUTEFAILED	Retrieving the value for a variable failed.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0049098	-1073442664	DTS_E_EXPREVALSTATIC_VARIABLETYPENOTSUPPORTED	The expression contains a variable with an unsupported data type.
0xC004909B	-1073442661	DTS_E_EXPREVALSTATIC_CASTCODEPAGEMISMATCH	Unable to cast the expression because the code page of the value being cast does not match the requested result code page. The code page of the source must match the code page requested for the destination.
0xC004909C	-1073442660	DTS_E_EXPREVALSTATIC_INVALIDTOKENSINGLEQUOTE	The expression contains an unexpected single quotation mark. A double quotation mark may be required.
0xC004909D	-1073442659	DTS_E_EXPREVALSTATIC_INVALIDTOKENSINGLEEQUAL	The expression contains an unexpected equal sign (=). This error usually occurs when a double equals sign (==) is needed.
0xC00490AA	-1073442646	DTS_E_EXPREVALSTATIC_AMBIGUOUSINPUTCOLUMNNAME	An ambiguous input column name was specified. The column must be qualified as [Component Name].[Column Name] or referenced by lineage ID. This error occurs when the input column exists on more than one component, and must be differentiated by the addition of component name or by using the lineage ID.
0xC00490AB	-1073442645	DTS_E_EXPREVALSTATIC_PLACEHOLDERINEXPRESSION	A placeholder function parameter or operand was found in an expression. This should be replaced with an actual parameter or operand.
0xC00490AC	-1073442644	DTS_E_EXPREVALSTATIC_AMBIGUOUSVARIABLENAME	An ambiguous variable name was specified. The desired variable must be qualified as @[Namespace::Variable]. This error occurs when the variable exists in more than one namespace.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC00490D3	-1073442605	DTS_E_EXPREVALSTATIC_BINARYOPDTSTRNOTSUPPORTED	For operands of binary operation, the data type DT_STR is only supported for input columns and cast operations. A DT_STR operand that is not an input column or the result of a cast cannot be used with a binary operation. To perform this operation, the operand needs to be explicitly cast with a cast operator.
0xC00490D4	-1073442604	DTS_E_EXPREVALSTATIC_CONDITIONALOPDTSTRNOTSUPPORTED	For operands of the conditional operator, the data type DT_STR is only supported for input columns and cast operations. A DT_STR operand that is not an input column or the result of a cast cannot be used with the conditional operation. To perform this operation, the operand needs to be explicitly cast with a cast operator.
0xC00490D5	-1073442603	DTS_E_EXPREVALSTATIC_FINDSTRINGINVALIDOCCURRENCECOUNT	The occurrence count parameter is not valid for function FINDSTRING. This parameter must be greater than zero.
0xC00490DD	-1073442595	DTS_E_EXPREVALSTATIC_INVALIDDATEPARTNODE	The "date part" parameter specified for a date function is not valid. "Date part" parameters must be static strings, and cannot contain dynamic elements such as input columns. They must be of type DT_WSTR.
0xC00490DE	-1073442594	DTS_E_EXPREVALSTATIC_INVALIDCASTLENGTH	The value specified for the length parameter of a cast operation is not valid. The length must be positive. The length specified for the type cast is negative. Change to a positive value.
0xC00490DF	-1073442593	DTS_E_EXPREVALSTATIC_INVALIDNULLLENGTH	The value specified for the length parameter of a NULL function is not valid. The length must be positive. The length specified for the NULL function is negative. Change to a positive value.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC00490E0	-1073442592	DTS_E_EXPREVALSTATIC_IN VALIDNULLCODEPAGE	The value specified for the code page parameter of the NULL function with data type DT_STR or DT_TEXT is not valid. The code page specified is not installed on the computer. Either change the code page that is specified, or install the code page on the computer.
0xC00490E1	-1073442591	DTS_E_EXPREVALSTATIC_IN VALIDNULLPRECISION	The value specified for the precision parameter of a NULL function is not valid. The precision that was specified is out of range for the NULL function.
0xC00490E2	-1073442590	DTS_E_EXPREVALSTATIC_IN VALIDNULLSCALE	The value specified for the scale parameter of a NULL function is not valid. The scale that was specified is out of range for the NULL function. Scale must not exceed precision and must be positive.
0xC00490E8	-1073442584	DTS_E_XMLSRCERRORSETTI NGERROROUTPUTCOLUMN DATA	The %1 failed attempting to write data to %2 on %3. %4
0xC00490F5	-1073442571	DTS_E_TXLOOKUP_CANCEL_ REQUESTED	Lookup transform has received a cancel request from the user.
0xC00490F6	-1073442570	DTS_E_LOBLENGTHLIMITEX CEEDED	Processing of character or binary large object (LOB) data has stopped because the 4-GB limit was reached.
0xC00490F7	-1073442569	DTS_E_CANNOTLOADCOM PONENT	The managed pipeline component "%1" could not be loaded. The exception was: %2.
0xC00F9304	-1072721148	DTS_E_OLEDB_EXCEL_NOT_ SUPPORTED	SSIS Error Code DTS_E_OLEDB_EXCEL_NOT_ SUPPORTED: The Excel Connection Manager is not supported in the 64-bit version of SSIS, as no OLE DB provider is available.
0xC00F9310	-1072721136	DTS_E_CACHEBADHEADER	The cache file is damaged, or the file was not created by using the Cache connection manager. Provide a valid cache file.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0202001	-1071636479	DTS_E_MISSINGSQLCOMM AND	The SQL command has not been set correctly. Check SqlCommand property.
0xC0202002	-1071636478	DTS_E_COMERROR	COM error object information is available. Source: "%1" error code: 0x%2!8.X! Description: "%3".
0xC0202003	-1071636477	DTS_E_ACQUIREDCONNECTIONUNAVAILABLE	Unable to access the acquired connections.
0xC0202004	-1071636476	DTS_E_INCORRECTCOLUMNCOUNT	The number of columns is incorrect.
0xC0202005	-1071636475	DTS_E_COLUMNNOTFOUND	Column "%1" cannot be found at the datasource.
0xC0202007	-1071636473	DTS_E_OLEDBRECORD	An OLE DB record is available. Source: "%1" Hresult: 0x%2!8.X! Description: "%3".
0xC0202009	-1071636471	DTS_E_OLEDBERROR	SSIS Error Code DTS_E_OLEDBERROR. An OLE DB error has occurred. Error code: 0x%1!8.X!.
0xC020200A	-1071636470	DTS_E_ALREADYCONNECTED	Component is already connected. The component needs to be disconnected before attempting to connect it.
0xC020200B	-1071636469	DTS_E_INCORRECTSTOCKPROPERTYVALUE	The value of the property "%1" is incorrect.
0xC020200E	-1071636466	DTS_E_CANNOTOPENDATAFILE	Cannot open the datafile "%1".
0xC0202010	-1071636464	DTS_E_DESTINATIONFLATFILEREQUIRED	No destination flat file name was provided. Make sure the flat file connection manager is configured with a connection string. If the flat file connection manager is used by multiple components, ensure that the connection string contains enough file names.
0xC0202011	-1071636463	DTS_E_TEXTQUALIFIERNOTFOUND	The text qualifier for column "%1" cannot be found.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0202014	-1071636460	DTS_E_CANNOTCONVERTTYPES	Conversion from "%1" to "%2" is not supported.
0xC0202015	-1071636459	DTS_E_PROBLEMDetectingTypeCompatibility	The error code 0x%1!8X! was returned when validating type conversion from %2 to %3.
0xC0202016	-1071636458	DTS_E_CANNOTMAPINPUTCOLUMNTOOUTPUTCOLUMN	Cannot find input column with lineage ID "%1!d!" which is needed by "%2". Check SourceInputColumnLineageID custom property of the output column.
0xC0202017	-1071636457	DTS_E_INCORRECTMINIMUMNUMBEROFOUTPUTS	The number of outputs is incorrect. There must be at least %1!d! outputs.
0xC0202018	-1071636456	DTS_E_INCORRECTEXACTNUMBEROFOUTPUTS	The number of outputs is incorrect. There must be exactly %1!d! output(s).
0xC0202019	-1071636455	DTS_E_STRINGCONVERSIONTOOLONG	A string was too long to be converted.
0xC020201A	-1071636454	DTS_E_INCORRECTEXACTNUMBEROFINPUTS	The number of inputs is incorrect. There must be exactly %1!d! inputs.
0xC020201B	-1071636453	DTS_E_CANNOTHAVEZEROINPUTCOLUMNS	The number of input columns for %1 cannot be zero.
0xC020201C	-1071636452	DTS_E_CANNOTHAVEINPUTS	This component has %1!d! inputs. No input is allowed on this component.
0xC020201D	-1071636451	DTS_E_PROCESSINPUTCALLEDWITHINVALIDINPUTID	ProcessInput was called with an invalid input ID of %1!d!.
0xC020201F	-1071636449	DTS_E_INCORRECTCUSTOMPROPERTYTYPE	The custom property "%1" needs to be of type %2.
0xC0202020	-1071636448	DTS_E_INVALIDBUFFERFERTYPE	The buffer type is not valid. Make sure the Pipeline layout and all components pass validation.
0xC0202021	-1071636447	DTS_E_INCORRECTCUSTOMPROPERTYVALUE	The value for custom property "%1" is incorrect.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0202022	-1071636446	DTS_E_CONNECTIONREQUI REDFORMETADATA	An error occurred due to no connection. A connection is required when requesting metadata. If you are working offline, uncheck Work Offline on the SSIS menu to enable the connection.
0xC0202023	-1071636445	DTS_E_CANTCREATECUSTO MPROPERTY	The custom property "%1" cannot be created.
0xC0202024	-1071636444	DTS_E_CANTGETCUSTOMPR OPERTYCOLLECTION	The custom property collection cannot be retrieved for initialization.
0xC0202025	-1071636443	DTS_E_CANNOTCREATEACC ESSOR	Cannot create an OLE DB accessor. Verify that the column metadata is valid.
0xC0202026	-1071636442	DTS_E_PRIMEOUTPUTCALLED WITHINVALIDOUTPUTID	PrimeOutput was called with an invalid output ID of %1!d!.
0xC0202027	-1071636441	DTS_E_INCORRECTSTOCKPR OPERTY	The value for property "%1" on "%2" is not valid.
0xC0202028	-1071636440	DTS_E_CONNECTIONREQUI REDFORREAD	A connection is required to read data.
0xC020202C	-1071636436	DTS_E_ERRORWHILEREADIN GHEADERROWS	An error occurred while reading header rows.
0xC020202D	-1071636435	DTS_E_DUPLICATECOLUMN NAME	Duplicate column name "%1".
0xC0202030	-1071636432	DTS_E_CANNOTGETCOLUM NNAME	Cannot get the name of the column with ID %1!d!.
0xC0202031	-1071636431	DTS_E_CANTDIRECTROW	Direct row to output "%1" (%2!d!) failed.
0xC020203A	-1071636422	DTS_E_CANNOTCREATEBUL KINSERTTHREAD	Cannot create the bulk insert thread due to error "%1".
0xC020203B	-1071636421	DTS_E_BULKINSERTTHREADI NITIALIZATIONFAILED	The thread for the SSIS Bulk Insert task failed initialization.
0xC020203E	-1071636418	DTS_E_BULKINSERTTHRE ADALREADYRUNNING	The thread for the SSIS Bulk Insert task is already running.



HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC020203F	-1071636417	DTS_E_BULKINSERTTHREAD ABNORMALCOMPLETION	The thread for the SSIS Bulk Insert task terminated with errors or warnings.
0xC0202040	-1071636416	DTS_E_CANNOTGETIROWSE TFASTLOAD	Failed to open a fastload rowset for "%1". Check that the object exists in the database.
0xC0202041	-1071636415	DTS_E_CONNECTREQUIRED FORMETADATAVALIDATIO N	Error due to no connection. A connection is required before metadata validation can proceed.
0xC0202042	-1071636414	DTS_E_DESTINATIONTABLE NAMENOTPROVIDED	A destination table name has not been provided.
0xC0202043	-1071636413	DTS_E_ICONVERTTYPEUNAV AILABLE	The OLE DB provider used by the OLE DB adapter does not support IConvertType. Set the adapter's ValidateColumnMetaData property to FALSE.
0xC0202044	-1071636412	DTS_E_OLEDBPROVIDERDA TATYPECONVERSIONUNSU PPORTED	The OLE DB provider used by the OLE DB adapter cannot convert between types "%1" and "%2" for "%3".
0xC0202045	-1071636411	DTS_E_VALIDATECOLUMN METADATAFAILED	Column metadata validation failed.
0xC0202047	-1071636409	DTS_E_ATTEMPTINGTOINSE RTINTOAROWIDCOLUMN	"%1" is a row ID column and cannot be included in a data insertion operation.
0xC0202048	-1071636408	DTS_E_ATTEMPTINGTOINSE RTINTOAROWVERCOLUMN	Attempting insertion into the row version column "%1". Cannot insert into a row version column.
0xC0202049	-1071636407	DTS_E_ATTEMPTINGTOINSE RTINTOAREADONLYCOLUM N	Failure inserting into the read-only column "%1".
0xC020204A	-1071636406	DTS_E_UNABLETORETRIEVE COLUMNINFO	Unable to retrieve column information from the data source. Make sure your target table in the database is available.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC020204B	-1071636405	DTS_E_CANTLOCKBUFFER	A buffer could not be locked. The system is out of memory or the buffer manager has reached its quota.
0xC020204C	-1071636404	DTS_E_INVALIDCOMPARISONFLAGS	The %1 has a ComparisonFlags property that includes extra flags with the value %2!d!.
0xC020204D	-1071636403	DTS_E_COLUMNMETADATAUNAVAILABLEFORVALIDATION	The column metadata was unavailable for validation.
0xC0202053	-1071636397	DTS_E_CANNOTWRITETODATFILE	Cannot write to the data file.
0xC0202055	-1071636395	DTS_E_COLUMNDELIMITERNOTFOUND	The column delimiter for column "%1" was not found.
0xC0202058	-1071636392	DTS_E_COLUMNPARSEFAILED	Failed to parse the column "%1" in the data file.
0xC020205A	-1071636390	DTS_E_RAWFILENAMEREQUIRED	The file name is not properly specified. Supply the path and name to the raw file either directly in the FileName property or by specifying a variable in the FileNameVariable property.
0xC020205B	-1071636389	DTS_E_RAWFILECANTOPEN	File "%1" cannot be opened for writing. Error may occur when there are no file privileges or the disk is full.
0xC020205C	-1071636388	DTS_E_RAWFILECANTBUFFER	An I/O buffer cannot be created for the output file. Error may occur when there are no file privileges or the disk is full.
0xC020205D	-1071636387	DTS_E_RAWCANTWRITE	Cannot write %1!d! bytes to file "%2". See previous error messages for details.
0xC020205E	-1071636386	DTS_E_RAWBADHEADER	Encountered bad metadata in file header. The file is damaged or not a SSIS-produced raw data file.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC020205F	-1071636385	DTS_E_RAWEXISTSCREATEO NCE	Error occurred because the output file already exists and the WriteOption is set to Create Once. Set the WriteOption property to Create Always, or delete the file.
0xC0202060	-1071636384	DTS_E_RAWCANTAPPENDTR UNCATE	Error caused by conflicting property settings. Both the AllowAppend property and the ForceTruncate property are set to TRUE. Both properties cannot be set to TRUE. Set one of the two properties to FALSE.
0xC0202061	-1071636383	DTS_E_RAWBADVERSION	The file had bad version and flags information. The file is damaged or not a SSIS-produced raw data file.
0xC0202062	-1071636382	DTS_E_RAWVERSIONINCOM PATIBLEAPPEND	The output file was written by an incompatible version and cannot be appended. The file may be an older file format that is no longer useable.
0xC0202064	-1071636380	DTS_E_RAWMETADATAMIS MATCH	Cannot append output file because no column in the existing file matches column "%1" from the input. Old file does not match in metadata.
0xC0202065	-1071636379	DTS_E_RAWMETADATACOU NTMISMATCH	Cannot append output file because the number of columns in the output file does not match the number of columns in this destination. The old file does not match in metadata.
0xC0202067	-1071636377	DTS_E_ERRORRETRIEVINGC OLUMNCODEPAGE	There was an error retrieving column code page information.
0xC0202068	-1071636376	DTS_E_RAWCANTREAD	Cannot read %1!d! bytes from file "%2". The cause of the failure should have been previously reported.
0xC0202069	-1071636375	DTS_E_RAWUNEXPECTEDEO F	Unexpected end-of-file encountered while reading %1!d! bytes from file "%2". The file ended prematurely because of an invalid file format.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC020206A	-1071636374	DTS_E_RAWNOLONGTYPES	The column %1 cannot be used. The raw adapters do not support image, text, or ntext data.
0xC020206B	-1071636373	DTS_E_RAWUNEXPECTEDTYPE	The adapter encountered an unrecognized data type of %1!d!. This could be caused by a damaged input file (source) or by an invalid buffer type (destination).
0xC020206C	-1071636372	DTS_E_RAWSTRINGTOOLONG	String too long. The adapter read a string that was %1!d! bytes long, and expected a string no longer than %2!d! bytes, at offset %3!d!. This could indicate a damaged input file. The file shows a string length that is too large for the buffer column.
0xC020206E	-1071636370	DTS_E_RAWSKIPFAILED	The raw adapter attempted to skip %1!d! bytes in the input file for unreferenced column "%2" with lineage ID %3!d!, but there was an error. The error returned from the operating system should have been previously reported.
0xC020206F	-1071636369	DTS_E_RAWREADFAILED	The raw adapter attempted to read %1!d! bytes in the input file for column "%2" with lineage ID %3!d!, but there was an error. The error returned from the operating system should have been previously reported.
0xC0202070	-1071636368	DTS_E_RAWFILENAMEINVALID	The file name property is not valid. The file name is a device or contains invalid characters.
0xC0202071	-1071636367	DTS_E_BULKINSERTAPIPREPARATIONFAILED	Unable to prepare the SSIS bulk insert for data insertion.
0xC0202072	-1071636366	DTS_E_INVALIDDATABASEOBJECTNAME	Database object name "%1" is not valid.
0xC0202073	-1071636365	DTS_E_INVALIDORDERCLAUSE	Order clause is not valid.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0202074	-1071636364	DTS_E_RAWFILECANTOPEN READ	File "%1" cannot be opened for reading. Error may occur when there are no privileges or the file is not found. Exact cause is reported in previous error message.
0xC0202075	-1071636363	DTS_E_TIMEGENCANTCREAT E	Unable to create the Microsoft.AnalysisServices.TimeDimGenerator.TimeDimGenerator.
0xC0202076	-1071636362	DTS_E_TIMEGENCANTCONF IGURE	Unable to configure the Microsoft.AnalysisServices.TimeDimGenerator.
0xC0202077	-1071636361	DTS_E_TIMEGENCANTCONV ERT	Unsupported datatype for column %1!d!.
0xC0202079	-1071636359	DTS_E_TIMEGENCANTREAD	The attempt to read from the Microsoft.AnalysisServices.TimeDimGenerator failed with error code 0x%1!8.8X!.
0xC020207A	-1071636358	DTS_E_TIMEGENCANTREAD COLUMN	The attempt to read column "%2!d!" data from the Microsoft.AnalysisServices.TimeDimGenerator failed with error code 0x%2!8.8X!.
0xC020207B	-1071636357	DTS_E_RSTDESTBADVARIAB LENAME	The VariableName property is not set to the name of a valid variable. Need a runtime variable name to write to.
0xC020207C	-1071636356	DTS_E_RSTDESTRSTCONFIG PROBLEM	Unable to create or configure the ADODB.Recordset object.
0xC020207D	-1071636355	DTS_E_RSTDESTRSTWRITEPR OBLEM	Error writing to the ADODB.Recordset object.
0xC020207E	-1071636354	DTS_E_FILENAMEINVALID	The file name is not valid. The file name is a device or contains invalid characters.
0xC020207F	-1071636353	DTS_E_FILENAMEINVALIDW ITHPARAM	The file name "%1" is not valid. The file name is a device or contains invalid characters.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0202080	-1071636352	DTS_E_CMDDESTNOPARAMS	Unable to retrieve destination column descriptions from the parameters of the SQL command.
0xC0202081	-1071636351	DTS_E_CMDDESTNOTBOUND	Parameters are not bound. All parameters in the SQL command must be bound to input columns.
0xC0202082	-1071636350	DTS_E_TXPIVOTBADUSAGE	The PivotUsage value for the input column "%1" (%2!d!) is not valid.
0xC0202083	-1071636349	DTS_E_TXPIVOTTOOMANYPIVOTKEYS	Too many Pivot Keys found. Only one input column can be used as the Pivot Key.
0xC0202084	-1071636348	DTS_E_TXPIVOTNOPIVOTKEY	No Pivot Key found. One input column must be used as the Pivot Key.
0xC0202085	-1071636347	DTS_E_TXPIVOTINPUTALREADYMAPPED	More than one output column (such as "%1" (%2!d!)) is mapped to input column "%3" (%4!d!).
0xC0202086	-1071636346	DTS_E_TXPIVOTCANTMAPPIVOTKEY	Output column "%1" (%2!d!) cannot be mapped to PivotKey input column.
0xC0202087	-1071636345	DTS_E_TXPIVOTCANTMAPPIVOTKEYNOTFOUND	Output column "%1" (%2!d!) has a SourceColumn %3!d! that is not a valid input column lineage ID.
0xC0202088	-1071636344	DTS_E_TXPIVOTEMPTYPIVOTKEYVALUE	Output column "%1" (%2!d!) is mapped to a Pivoted Value input column, but its PivotKeyValue property value is missing.
0xC0202089	-1071636343	DTS_E_TXPIVOTDUPLICATEPIVOTKEYVALUE	Output column "%1" (%2!d!) is mapped to a Pivoted Value input column with a non-unique PivotKeyValue property value.
0xC020208A	-1071636342	DTS_E_TXPIVOTOUTPUTNOTMAPPED	Input column "%1" (%2!d!) is not mapped to any output column.
0xC020208B	-1071636341	DTS_E_TXPIVOTCANTCOMPARESETKEYS	Failure occurred while comparing values for the set keys.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC020208D	-1071636339	DTS_E_TXPIVOTNOBLOB	The Input column "%1" (%2!d!) cannot be used as a Set Key, Pivot Key, or Pivot Value because it contains long data.
0xC020208E	-1071636338	DTS_E_TXPIVOTBADOUTPUTTYPE	Incorrect output type. The output column "%1" (%2!d!) must have the same data type and metadata as the input column to which it is mapped.
0xC020208F	-1071636337	DTS_E_TXPIVOTPROCESSERERROR	Failure when trying to pivot the source records.
0xC0202090	-1071636336	DTS_E_TXPIVOTBADPIVOTKEYVALUE	The pivot key value "%1" is not valid.
0xC0202091	-1071636335	DTS_E_ERRORWHILESKIPINGDATAROWS	An error occurred while skipping data rows.
0xC0202092	-1071636334	DTS_E_ERRORWHILEREADINGDATAROWS	An error occurred while processing file "%1" on data row %2!l64d!.
0xC0202093	-1071636333	DTS_E_FAILEDTOINITIALIZEFLATFILEPARSER	An error occurred while initializing the flat file parser.
0xC0202094	-1071636332	DTS_E_UNABLETORETRIEVECOLUMNINFOFROMFLATFILECONNECTIONMANAGER	Unable to retrieve column information from the flat file connection manager.
0xC0202095	-1071636331	DTS_E_FAILEDTOWRITEOUTCOLUMNNAME	Failed to write out column name for column "%1".
0xC0202096	-1071636330	DTS_E_INVALIDFLATFILECOLUMNNTYPE	The column type for column "%1" is incorrect. It is type "%2". It can only be either "%3" or "%4".
0xC0202097	-1071636329	DTS_E_DISKIOBUFFEROVERFLOW	The attempt to write data of %1!d! bytes into the disk I/O failed. The disk I/O buffer has %2!d! free bytes.
0xC0202098	-1071636328	DTS_E_FAILEDTOWRITEOUTHEADER	An error occurred while writing out the file header.
0xC0202099	-1071636327	DTS_E_FAILEDTOGETFILESIZE	An error occurred while getting the file size for file "%1".

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC020209A	-1071636326	DTS_E_FAILEDTOSETFILEPOINTER	An error occurred while setting the file pointer for file "%1".
0xC020209B	-1071636325	DTS_E_UNABLETOSETUPDISKIOBUFFER	An error occurred while setting up the disk I/O buffer.
0xC020209C	-1071636324	DTS_E_COLUMNDATAOVERFLOWDISKIOBUFFER	The column data for column "%1" overflowed the disk I/O buffer.
0xC020209D	-1071636323	DTS_E_DISKIOFAILED	An unexpected disk I/O error occurred while reading the file.
0xC020209E	-1071636322	DTS_E_DISKIOTIMEDOUT	An disk I/O time out occurred while reading the file.
0xC020209F	-1071636321	DTS_E_INPUTSNOTREADONLY	The Usage Type specified for the input columns to this transform cannot be read/write. Change the Usage Type to be read-only.
0xC02020A0	-1071636320	DTS_E_CANNOTCOPYORCONVERTFLATFILEDATA	Cannot copy or convert flat file data for column "%1".
0xC02020A1	-1071636319	DTS_E_FAILEDCOLUMNDATAACONVERSIONSTATUS	Data conversion failed. The data conversion for column "%1" returned status value %2!d! and status text "%3".
0xC02020A2	-1071636318	DTS_E_VARIABLESCOLLECTIONUNAVAILABLE	The Variables collection is not available.
0xC02020A3	-1071636317	DTS_E_TXUNPIVOTDUPLICATEPIVOTKEYVALUE	Duplicate PivotKeyValue. Input column "%1" (%2!d!) is mapped to a Pivoted Value output column and has a non-unique PivotKeyValue.
0xC02020A4	-1071636316	DTS_E_TXUNPIVOTNOUNPIVOTDESTINATION	No unpivot destination found. At least one input column must be mapped with a PivotKeyValue to an DestinationColumn in the output.



HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC02020A5	-1071636315	DTS_E_TXUNPIVOTBADKEYLIST	PivotKeyValue is not valid. In an UnPivot transform with more than one unpivoted DestinationColumn, the set of PivotKeyValues per destination must match exactly.
0xC02020A6	-1071636314	DTS_E_TXUNPIVOTBADUNPIVOTMETADATA	Incorrect UnPivot metadata. In an UnPivot transform, all input columns with a PivotKeyValue that is set, and are pointing to the same DestinationColumn, must have metadata that exactly matches the DestinationColumn.
0xC02020A7	-1071636313	DTS_E_TXPIVOTBADPIVOTKEYCONVERT	Cannot convert the pivot key value "%1" to the data type of the pivot key column.
0xC02020A8	-1071636312	DTS_E_TXUNPIVOTTOOMANYPIVOTKEYS	Too many Pivot Keys specified. Only one output column can be used as the Pivot Key.
0xC02020A9	-1071636311	DTS_E_TXUNPIVOTUNMAPPEDEOUTPUT	Output column "%1" (%2!d!) is not mapped by any input column's DestinationColumn property.
0xC02020AA	-1071636310	DTS_E_TXUNPIVOTNOPIVOT	No output column is marked as the PivotKey.
0xC02020AB	-1071636309	DTS_E_TXUNPIVOTNOTINPUTMAP	Input column "%1" (%2!d!) has a DestinationColumn property value that does not refer to a valid output column LineageID.
0xC02020AC	-1071636308	DTS_E_TXUNPIVOTDUPLICATEDESTINATION	Duplicate destination error. More than one non-pivoted input column is mapped to the same destination output column.
0xC02020AD	-1071636307	DTS_E_TOTALINPUTCOLSCANNOTBEZERO	No input columns found. At least one input column must be mapped to an output column.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC02020AE	-1071636306	DTS_E_TXMERGEJOINMUST HAVESAMENUMBEROFINP UTANDOUTPUTCOLS	The number of input and output columns are not equal. The total number of input columns on all inputs must be the same as the total number of output columns.
0xC02020AF	-1071636305	DTS_E_INPUTMUSTBESORTE D	The input is not sorted. The "%1" must be sorted.
0xC02020B0	-1071636304	DTS_E_TXMERGEJOININVALI DJOINTYPE	The JoinType custom property for the %1 contains a value of %2!d!, which is not valid. Valid values are 0 (full), 1 (left), or 2 (inner).
0xC02020B1	-1071636303	DTS_E_TXMERGEJOININVALI DNUMKEYCOLS	The NumKeyColumns value is not valid. In the %1, the value for the NumKeyColumns custom property must be between 1 and %2!lu!.
0xC02020B2	-1071636302	DTS_E_NOKEYCOLS	No key columns are found. The %1 must have at least one column with a SortKeyPosition that is non-zero.
0xC02020B3	-1071636301	DTS_E_TXMERGEJOINNOTE NOUGHKEYCOLS	Not enough key columns. The %1 must have at least %2!d! columns with non-zero SortKeyPosition values.
0xC02020B4	-1071636300	DTS_E_TXMERGEJOINDATAT YPEMISMATCH	Datatype mismatch occurred. The datatypes for the columns with SortKeyPosition value %1!d! do not match.
0xC02020B5	-1071636299	DTS_E_TXMERGEJOININVALI DSORTKEYPOS	The column with the SortKeyPosition value of %1!d! is not valid. It should be %2!d!.
0xC02020B6	-1071636298	DTS_E_TXMERGEJOINSORTD IRECTIONMISMATCH	Sort direction mismatch. The sort directions for the columns with SortKeyPosition value %1!d! do not match.
0xC02020B7	-1071636297	DTS_E_TXMERGEJOINOUTP UTCOLMUSTHAVEASSOCIA TEDINPUTCOL	Missing column. The %1 must have an associated input column.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC02020B8	-1071636296	DTS_E_TXMERGEJOINREAD ONLYINPUTCOLSWITHNOO UTPUTCOL	Input columns must have output columns. There are input columns with a usage type of read-only that do not have associated output columns.
0xC02020B9	-1071636295	DTS_E_TXMERGEJOINNONSTR INGCOMPARISONFLAGS NOTZERO	The comparison flags are not zero. The comparison flags for non-string columns must be zero.
0xC02020BA	-1071636294	DTS_E_TXMERGEJOINCOMPAR ISONFLAGSMISMATCH	The comparison flags for the columns with SortKeyPosition value %1!d! do not match.
0xC02020BB	-1071636293	DTS_E_TXPIVOTBADPIVOTKEY VALUEONSTRING	Unrecognized pivot key value.
0xC02020BC	-1071636292	DTS_E_TXLINEAGEINVALIDLI NEAGEITEM	Lineage item value %1!d! is not valid. The valid range is between %2!d! and %3!d!.
0xC02020BD	-1071636291	DTS_E_CANNOTHAVEANYIN PUTCOLUMNS	Input columns not allowed. The number of input columns must be zero.
0xC02020BE	-1071636290	DTS_E_TXLINEAGEDATATYPE MISMATCH	The datatype for "%1" is not valid for the specified lineage item.
0xC02020BF	-1071636289	DTS_E_TXLINEAGEINVALIDL ENGTH	The length for "%1" is not valid for the specified lineage item.
0xC02020C1	-1071636287	DTS_E_METADATAMISMATCH WITHOUTOUTPUTCOLUMN	The metadata for "%1" does not match the metadata for the associated output column.
0xC02020C3	-1071636285	DTS_E_TXMERGESORTKEYPOS ITIONMISMATCH	There are output columns that have SortKeyPosition values that don't match the associated input columns' SortKeyPosition.
0xC02020C4	-1071636284	DTS_E_ADDROWTOBUFFERF AILED	The attempt to add a row to the Data Flow task buffer failed with error code 0x%1!8.X!.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC02020C5	-1071636283	DTS_E_DATACONVERSIONFAILED	Data conversion failed while converting column "%1" (%2!d!) to column "%3" (%4!d!). The conversion returned status value %5!d! and status text "%6".
0xC02020C6	-1071636282	DTS_E_FAILEDTOALLOCATEROWHANDLEBUFFER	The attempt to allocate a row handle buffer failed with error code 0x%1!8.8X!.
0xC02020C7	-1071636281	DTS_E_FAILEDTOSENDROWTOSQLSERVER	The attempt to send a row to SQL Server failed with error code 0x%1!8.8X!.
0xC02020C8	-1071636280	DTS_E_FAILEDTOPREPAREBUFFERSTATUS	The attempt to prepare the buffer status failed with error code 0x%1!8.8X!.
0xC02020C9	-1071636279	DTS_E_FAILEDTOBUFFERROWSTARTS	The attempt to retrieve the start of the buffer row failed with error code 0x%1!8.8X!.
0xC02020CA	-1071636278	DTS_E_BULKINSERTTHREADTERMINATED	The thread for the SSIS Bulk Insert is no longer running. No more rows can be inserted. Try increasing the bulk insert thread timeout.
0xC02020CB	-1071636277	DTS_E_RAWTOOMANYCOLUMNS	The source file is not valid. The source file is returning a count of more than 131,072 columns. This usually occurs when the source file is not produced by the raw file destination.
0xC02020CC	-1071636276	DTS_E_TXUNIONALL_EXTRADANGLINGINPUT	The %1 is an extra unattached input and will be removed.
0xC02020CD	-1071636275	DTS_E_TXUNIONALL_NONDANGLINGUNATTACHEDINPUT	The %1 is not attached but is not marked as dangling. It will be marked as dangling.
0xC02020CF	-1071636273	DTS_E_TXPIVOTRUNTIMEDUPLICATEPIVOTKEYVALUE	Duplicate pivot key value "%1".
0xC02020D0	-1071636272	DTS_E_TXPIVOTRUNTIMEDUPLICATEPIVOTKEYVALUEONSTRING	Duplicate pivot key value.
0xC02020D1	-1071636271	DTS_E_FAILEDTOGETCOMPONENTLOCALEID	Failure retrieving component locale ID. Error code 0x%1!8.8X!.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC02020D2	-1071636270	DTS_E_MISMATCHCOMPONENTCONNECTIONMANAGERLOCALEID	Mismatched locale IDs. The component locale ID (%1!d!) does not match the connection manager locale ID (%2!d!).
0xC02020D3	-1071636269	DTS_E_LOCALEIDNOTSET	The component locale ID has not been set. Flat file adapters need to have the locale ID on the flat file connection manager set.
0xC02020D4	-1071636268	DTS_E_RAWBYTESTOOLONG	The binary field is too large. The adapter attempted to read a binary field that was %1!d! bytes long, but expected a field no longer than %2!d! bytes at offset %3!d!. This usually occurs when the input file is not valid. The file contains a string length that is too large for the buffer column.
0xC02020D5	-1071636267	DTS_E_TXSAMPLINGINVALIDPCT	The percentage, %2!d!, is not valid for the "%1" property. It must be between 0 and 100.
0xC02020D6	-1071636266	DTS_E_TXSAMPLINGINVALIDROWS	The number of rows, %2!d!, is not valid for the "%1" property. It must be greater than 0.
0xC02020D7	-1071636265	DTS_E_RAWSTRINGINPUTTOOLONG	The adapter was asked to write a string that was %1!64d! bytes long, but all data must be less than 4294967295 bytes in length.
0xC02020D9	-1071636263	DTS_E_ATLEASTONEINPUTMUSTBEMAPPEDTOOUTPUT	No inputs were mapped to an output. The "%1" must have at least one input column mapped to an output column.
0xC02020DB	-1071636261	DTS_E_CANNOTCONVERTDATATYPESWITHDIFFERENTCODEPAGES	Conversion from "%1" with code page %2!d! to "%3" with code page %4!d! is not supported.
0xC02020DC	-1071636260	DTS_E_COLUMNNOTMAPPEDEXTERNALMETADATACOLUMN	The external metadata column mapping for %1 is not valid. The external metadata column ID cannot be zero.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC02020DD	-1071636259	DTS_E_COLUMNMAPPEDTO NONEXISTENTEXTERNALME TADATACOLUMN	The %1 is mapped to an external metadata column that does not exist.
0xC02020E5	-1071636251	DTS_E_UNABLETOWRITELO BDATATOBUFFER	Writing long object data of type DT_TEXT, DT_NTEXT, or DT_IMAGE to Data Flow task buffer failed for column "%1".
0xC02020E8	-1071636248	DTS_E_CANNOTGETIROWSE T	Opening a rowset for "%1" failed. Check that the object exists in the database.
0xC02020E9	-1071636247	DTS_E_VARIABLEACCESSFAI LED	Accessing variable "%1" failed with error code 0x%2!8.8X!.
0xC02020EA	-1071636246	DTS_E_CONNECTIONMANA GERNOTFOUND	The connection manager "%1" is not found. A component failed to find the connection manager in the Connections collection.
0xC02020EB	-1071636245	DTS_E_VERSIONUPGRADEF AILED	The upgrade from version "%1" to version %2!d! failed.
0xC02020EC	-1071636244	DTS_E_RSTDESTBIGBLOB	A value in an input column is too large to be stored in the ADODB.Recordset object.
0xC02020ED	-1071636243	DTS_E_CANNOTCONVERTBE TWEENUNICODEANDNONU NICODESTRINGCOLUMNS	Columns "%1" and "%2" cannot convert between unicode and non-unicode string data types.
0xC02020EE	-1071636242	DTS_E_ROWCOUNTBADVAR IABLENAME	The variable "%1" specified by VariableName property is not a valid variable. Need a valid variable name to write to.
0xC02020EF	-1071636241	DTS_E_ROWCOUNTBADVAR IABLETYPE	The variable "%1" specified by VariableName property is not an integer. Change the variable to be of type VT_I4, VT_UI4, VT_I8, or VT_UI8.
0xC02020F0	-1071636240	DTS_E_NOCOLUMNADVAN CETHROUGHFILE	No column was specified to allow the component to advance through the file.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC02020F1	-1071636239	DTS_E_MERGEJOINSORTED OUTPUTHASNOSORTKEYPO SITIONS	The "%1" has IsSorted set to TRUE, but the SortKeyPosition on all output columns are zero. Either change the IsSorted to FALSE, or select at least one output column to contain a non-zero SortKeyPosition.
0xC02020F2	-1071636238	DTS_E_METADATAMISMATCH WITHINPUTCOLUMN	The "%1" metadata does not match the metadata of the input column.
0xC02020F3	-1071636237	DTS_E_RSTDESTBADVARIABLE	The value of the specified variable cannot be located, locked, or set.
0xC02020F4	-1071636236	DTS_E_CANTPROCESSCOLUMN TYPECODEPAGE	The column "%1" cannot be processed because more than one code page (%2!d! and %3!d!) are specified for it.
0xC02020F5	-1071636235	DTS_E_CANTINSERTCOLUMN TYPE	The column "%1" can't be inserted because the conversion between types %2 and %3 is not supported.
0xC02020F6	-1071636234	DTS_E_CANNOTCONVERTBETWEEN UNICODEANDNONUNICODE STRINGCOLUMN	Column "%1" cannot convert between unicode and non-unicode string data types.
0xC02020F8	-1071636232	DTS_E_COULDNOTFINDINPUT BUFFERCOLUMNBYLINE AGE	The %1 cannot find the column with LineageID %2!ld! in its input buffer.
0xC02020F9	-1071636231	DTS_E_COULDNOTGETCOLUMN INFOFORINPUTBUFFER	The %1 cannot get the column information for column %2!lu! from its input buffer.
0xC02020FA	-1071636230	DTS_E_COULDNOTGETCOLUMN INFOFORCOPYBUFFER	The %1 cannot get the column information for column "%2!lu!" from its copy buffer.
0xC02020FB	-1071636229	DTS_E_COULDNOTREGISTER COPYBUFFER	The %1 cannot register a buffer type for its copy buffer.
0xC02020FC	-1071636228	DTS_E_COULDNOTCREATECOPY BUFFER	The %1 cannot create a buffer to copy its data into for sorting.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC02020FD	-1071636227	DTS_E_DATAREADERDESTRE ADFAILED	DataReader client has failed to call Read or has closed the DataReader.
0xC02020FE	-1071636226	DTS_E_NOSCHEMAINFOFO UND	No column information was returned by the SQL command.
0xC02020FF	-1071636225	DTS_E_GETSCHEMATABLFA ILED	The %1 was unable to retrieve column information for the SQL command. The following error occurred: %2
0xC0202100	-1071636224	DTS_E_SOURCETABLENAME NOTPROVIDED	A source table name has not been provided.
0xC0203110	-1071632112	DTS_E_CACHE_INVALID_IND EXPOS	The cache index position, %1!d!, is not valid. For non-index columns, the index position should be 0. For index columns, the index position should be a sequential, positive number.
0xC0203111	-1071632111	DTS_E_CACHE_DUPLICATE_I NDEXPOS	The index position, %1!d!, is a duplicate. For non-index columns, the index position should be 0. For index columns, the index position should be a sequential, positive number.
0xC0203112	-1071632110	DTS_E_CACHE_TOO_FEW_IN DEX_COLUMNS	At least one index column should be specified for the Cache connection manager. To specify an index column, set the Index Position property of the cache column.
0xC0203113	-1071632109	DTS_E_CACHE_INDEXPOS_N OT_CONTINUOUS	Cache index positions must be contiguous. For non-index columns, the index position should be 0. For index columns, the index position should be a sequential, positive number.
0xC0204000	-1071628288	DTS_E_PROPERTYNOTSUPP ORTED	The property "%1" cannot be set on "%2". The property being set is not supported on the specified object. Check the property name, case, and spelling.



HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0204002	-1071628286	DTS_E_CANTCHANGEPROPERTYPE	The property type cannot be changed from the type that was set by the component.
0xC0204003	-1071628285	DTS_E_CANTADDOUTPUTID	Output ID %1!d! failed during insert. The new output was not created.
0xC0204004	-1071628284	DTS_E_CANTDELETEOUTPUTID	Cannot delete output ID %1!d! from the output collection. The ID may not be valid, or the ID may have been the default or error output.
0xC0204006	-1071628282	DTS_E_FAILEDTOSETPROPERTY	Failed to set property "%1" on "%2".
0xC0204007	-1071628281	DTS_E_FAILEDTOSETOUTPUTCOLUMNTYPE	Failed to set the type of %1 to type: "%2", length: %3!d!, precision: %4!d!, scale: %5!d!, codepage: %6!d!.
0xC0204008	-1071628280	DTS_E_MORETHANONEERROROUTPUTFOUND	More than one error output was found on the component, and there can be only one.
0xC020400A	-1071628278	DTS_E_CANTSETOUTPUTCOLUMNPROPERTY	The property on an output column cannot be set.
0xC020400B	-1071628277	DTS_E_CANTMODIFYERROROUTPUTCOLUMNDATATYPE	The data type for "%1" cannot be modified in the error "%2".
0xC020400E	-1071628274	DTS_E_CANONLYSETISSORTEDONSOURCE	The "%1" cannot have its IsSorted property set to TRUE because it is not a source output. A source output has a SynchronousInputID value of zero.
0xC020400F	-1071628273	DTS_E_CANONLYSETSORTKEYONSOURCE	The "%1" cannot have a SortKeyPosition property set to non-zero because "%2" is not a source output. The output column "colname" (ID) cannot have its SortKeyPosition property set to non-zero because its output "outputname" (ID) is not a source output.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0204010	-1071628272	DTS_E_CANONLYSETCOMPLAGSONSOURCE	The ComparisonFlags property cannot be set to a non-zero value for "%1" because the "%2" is not a source output. The output column "colname" (ID) cannot have a ComparisonFlags property set to non-zero because its output "outputname" (ID) is not a source output.
0xC0204011	-1071628271	DTS_E_NONSTRINGCOMPARISONFLAGSNOTZERO	The comparison flags for "%1" must be zero because its type is not a string type. ComparisonFlags can only be non-zero for string type columns.
0xC0204012	-1071628270	DTS_E_COMPFLAGSONLYO NSORTCOL	The "%1" cannot have a ComparisonFlags property set to non-zero because its SortKeyPosition is set to zero. An output column's ComparisonFlags can only be non-zero if its SortKeyPosition is also non-zero.
0xC0204013	-1071628269	DTS_E_READONLYSTOCKPROPERTY	The property is read-only.
0xC0204014	-1071628268	DTS_E_INVALIDDATATYPE	The %1 had an invalid datatype value (%2!ld!) set.
0xC0204015	-1071628267	DTS_E_CODEPAGEREQUIRED	The "%1" requires a code page to be set but the value passed was zero.
0xC0204016	-1071628266	DTS_E_INVALIDSTRINGLENGTH	The "%1" has a length that is not valid. The length must be between %2!ld! and %3!ld!.
0xC0204017	-1071628265	DTS_E_INVALIDSCALE	The "%1" has a scale that is not valid. The scale must be between %2!ld! and %3!ld!.
0xC0204018	-1071628264	DTS_E_INVALIDPRECISION	The "%1" has a precision that is not valid. The precision must be between %2!ld! and %3!ld!.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0204019	-1071628263	DTS_E_PROPVALUEIGNORE D	The "%1" has a value set for length, precision, scale, or code page that is a value other than zero, but the data type requires the value to be zero.
0xC020401A	-1071628262	DTS_E_CANTSETOUTPUTCO LUMNDATATYPEPROPERTIE S	The %1 does not allow setting output column datatype properties.
0xC020401B	-1071628261	DTS_E_INVALIDDATATYPEF ORERRORCOLUMNS	The "%1" contains an invalid data type. "%1 " is a special error column, and the only valid data type is DT_I4.
0xC020401C	-1071628260	DTS_E_NOERRORDESCFORC OMPONENT	The component does not supply error code descriptions.
0xC020401D	-1071628259	DTS_E_UNRECOGNIZEDERR ORCODE	The specified error code is not associated with this component.
0xC020401F	-1071628257	DTS_E_TRUNCATIONTRIGGE REDREDIRECTION	A truncation caused a row to be redirected, based on the truncation disposition settings.
0xC0204020	-1071628256	DTS_E_CANTSETUSAGETYPE TOREADWRITE	The "%1" is unable to make the column with lineage ID %2!d! read/write because that usage type is not allowed on this column. An attempt was made to change the usage type of an input column to a type, UT_READWRITE, that is not supported on this component.
0xC0204023	-1071628253	DTS_E_CANTSETUSAGETYPE	The %1 has forbidden the requested use of the input column with lineage ID %2!d!.
0xC0204024	-1071628252	DTS_E_FAILEDTOSETUSAGET YPE	The "%1" was unable to make the requested change to the input column with lineage ID %2!d!. The request failed with error code 0x%3!8.X!. The specified error occurred while attempting to set the usage type of an input column.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0204025	-1071628251	DTS_E_FAILEDTOSETOUTPUTCOLUMNDATATYPEPROPERTIES	Attempt to set the data type properties on "%1" failed with error code 0x%2!8.8X!. The error occurred while attempting to set one or more of the data type properties of the output column.
0xC0204026	-1071628250	DTS_E_UNABLETORETRIEVEMETADATA	The metadata for "%1" cannot be retrieved. Make sure the object name is correct and the object exists.
0xC0204027	-1071628249	DTS_E_CANNOTMAPOUTPUTCOLUMN	The output column cannot be mapped to an external metadata column.
0xC0204028	-1071628248	DTS_E_UNSUPPORTEDVARIABLETYPE	The variable %1 is required to be of type "%2".
0xC020402A	-1071628246	DTS_E_CANTSETEXTERNALMETADATACOLUMNDATATYPEPROPERTIES	The %1 does not allow setting external metadata column datatype properties.
0xC020402B	-1071628245	DTS_E_IDNOTINPUTNOROUTPUT	The ID, %1!lu!, is neither an input ID nor an output ID. The specified ID must be the input ID or the output ID that the external metadata collection is associated with.
0xC020402C	-1071628244	DTS_E_METADATACOLLECTIONNOTUSED	The external metadata collection on "%1" is marked as not used, so no operations can be performed on it.
0xC020402D	-1071628243	DTS_E_NOBUFFERTYPEONSYNCOOUTPUT	The %1 is a synchronous output and the buffer type cannot be retrieved for a synchronous output.
0xC0207000	-1071616000	DTS_E_INPUTCOLUMNUSAGENOTREADONLY	The input column "%1" must be read-only. The input column has a usage type other than read-only, which is not allowed.
0xC0207001	-1071615999	DTS_E_MISSINGCUSTOMPROPERTY	The "%1" is missing the required property "%2". The object is required to have the specified custom property.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0207002	-1071615998	DTS_E_ILLEGALCUSTOMOUTPUTPROPERTY	The output %1 cannot not have property "%2", but currently has that property assigned.
0xC0207003	-1071615997	DTS_E_INVALIDOUTPUTEXCLUSIONGROUP	The %1 must be in exclusion group %2!d!. All outputs must be in the specified exclusion group.
0xC0207004	-1071615996	DTS_E_PROPERTYISEMPTY	The property "%1" is empty. The property cannot be empty.
0xC0207005	-1071615995	DTS_E_CREATEEXPRESSIONOBJECTFAILED	Memory cannot be allocated for the expression "%1". There was an out-of-memory error while creating an internal object to hold the expression.
0xC0207006	-1071615994	DTS_E_EXPRESSIONPARSEFAILED	Cannot parse the expression "%1". The expression was not valid, or there is an out-of-memory error.
0xC0207007	-1071615993	DTS_E_EXPRESSIONCOMPUTEFALLED	Computing the expression "%1" failed with error code 0x%2!8.X!. The expression may have errors, such as divide by zero, that cannot be detected at parse time, or there may be an out-of-memory error.
0xC0207008	-1071615992	DTS_E_FAILEDTOCREATEEXPRESSIONARRAY	Memory cannot be allocated for the Expression objects. An out-of-memory error occurred while creating the array of Expression object pointers.
0xC020700A	-1071615990	DTS_E_FAILEDTOCREATEEXPRESSIONMANAGER	The %1 failed with error code 0x%2!8.X! while creating the Expression Manager.
0xC020700B	-1071615989	DTS_E_SPLITEXPRESSIONNOTBOOLEAN	The expression "%1" is not Boolean. The result type of the expression must be Boolean.
0xC020700C	-1071615988	DTS_E_EXPRESSIONVALIDATIONFAILED	The expression "%1" on "%2" is not valid.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC020700E	-1071615986	DTS_E_COLUMNNOTMATCHED	The column "%1" (%2!d!) cannot be matched to any input file column. The output column name or input column name cannot be found in the file.
0xC020700F	-1071615985	DTS_E_SETRESULTCOLUMN FAILED	Attempting to set the result column for the expression "%1" on %2 failed with error code 0x%3!8.X!. The input or output column that was to receive the result of the expression cannot be determined, or the expression result cannot be cast to the column type.
0xC0207011	-1071615983	DTS_E_FAILEDTOGETLOCAL EIDFROMPACKAGE	The %1 failed to get the locale ID from the package.
0xC0207012	-1071615982	DTS_E_INCORRECTPARAMETERMAPPINGFORMAT	The parameter mapping string is not in the correct format.
0xC0207013	-1071615981	DTS_E_NOTENOUGHPARAMETERSPROVIDED	The SQL command requires %1!d! parameters, but the parameter mapping only has %2!d! parameters.
0xC0207014	-1071615980	DTS_E_PARAMETERNOTFOUNDINMAPPING	The SQL command requires a parameter named "%1", which is not found in the parameter mapping.
0xC0207015	-1071615979	DTS_E_DUPLICATEDDATA SOURCECOLUMNNAME	There is more than one data source column with the name "%1". The data source column names must be unique.
0xC0207016	-1071615978	DTS_E_DATASOURCECOLUMNWITHNONAMEFOUND	There is a data source column with no name. Each data source column must have a name.
0xC0208001	-1071611903	DTS_E_DISCONNECTEDCOMPONENT	A component is disconnected from the layout.
0xC0208002	-1071611902	DTS_E_INVALIDCOMPONENTID	The ID for a layout component is not valid.
0xC0208003	-1071611901	DTS_E_INVALIDINPUTCOUNT	A component has an invalid number of inputs.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0208004	-1071611900	DTS_E_INVALIDOUTPUTCO UNT	A component has an invalid number of outputs.
0xC0208005	-1071611899	DTS_E_NOINPUTSOROUTPU TS	A component does not have any inputs or outputs.
0xC0208007	-1071611897	DTS_E_CANTALLOCATECOL UMNINFO	Not enough memory was available to allocate a list of the columns that are being manipulated by this component.
0xC0208008	-1071611896	DTS_E_OUTPUTCOLUMNN OTINPUT	Output column "%1" (%2!d!) references input column with lineage ID %3!d!, but no input could be found with that lineage ID.
0xC0208009	-1071611895	DTS_E_SORTNEEDSONEKEY	At least one input column must be marked as a sort key, but no keys were found.
0xC020800A	-1071611894	DTS_E_SORTDUPLICATEKEY WEIGHT	Both column "%1" (%2!d!) and column "%3" (%4!d!) were marked with sort key weight %5!d!.
0xC020800D	-1071611891	DTS_E_CANTMODIFYINVALI D	The component cannot perform the requested metadata change until the validation problem is fixed.
0xC020800E	-1071611890	DTS_E_CANTADDINPUT	An input cannot be added to the inputs collection.
0xC020800F	-1071611889	DTS_E_CANTADDDOUTPUT	An output cannot be added to the outputs collection.
0xC0208010	-1071611888	DTS_E_CANTDELETEINPUT	An input cannot be deleted from the inputs collection.
0xC0208011	-1071611887	DTS_E_CANTDELETEOUTPUT	An output cannot be removed from the outputs collection.
0xC0208014	-1071611884	DTS_E_CANTCHANGEUSAGE TYPE	The usage type of the column cannot be changed.
0xC0208016	-1071611882	DTS_E_INVALIDUSAGETYPEF ORCUSTOMPROPERTY	The %1 must be read/write to have custom property "%2". The input or output column has the specified custom property, but is not read/write. Remove the property, or make the column read/write.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0208017	-1071611881	DTS_E_READWRITECOLUMN MISSINGREQUIREDCUSTOMPROPERTY	The %1 is read/write and is required to have custom property "%2". Add the property, or make remove the read/write attribute from the column.
0xC0208018	-1071611880	DTS_E_CANTDELETECOLUMN	The column cannot be deleted. The component does not allow columns to be deleted from this input or output.
0xC0208019	-1071611879	DTS_E_CANTADDCOLUMN	The component does not allow adding columns to this input or output.
0xC020801A	-1071611878	DTS_E_CANNOTTFINDRUNTIMECONNECTIONOBJECT	The connection "%1" cannot be found. Verify that the connection manager has a connection with that name.
0xC020801B	-1071611877	DTS_E_CANNOTFINDRUNTIMECONNECTIONMANAGER	The runtime connection manager with the ID "%1" cannot be found. Verify that the connection manager collection has a connection manager with that ID.
0xC020801C	-1071611876	DTS_E_CANNOTACQUIRECONNECTIONFROMCONNECTIONMANAGER	SSIS Error Code DTS_E_CANNOTACQUIRECONNECTIONFROMCONNECTIONMANAGER. The AcquireConnection method call to the connection manager "%1" failed with error code 0x%2!8.X!. There may be error messages posted before this with more information on why the AcquireConnection method call failed.
0xC020801D	-1071611875	DTS_E_ACQUIREDCONNECTIONISINVALID	The connection acquired from the connection manager "%1" is not valid.
0xC020801E	-1071611874	DTS_E_INCORRECTCONNECTIONMANAGERTYPE	The connection manager "%1" is an incorrect type. The type required is "%2". The type available to the component is "%3".
0xC020801F	-1071611873	DTS_E_CANNOTACQUIREMANAGEDCONNECTIONFROMCONNECTIONMANAGER	Cannot acquire a managed connection from the runtime connection manager.



HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0208020	-1071611872	DTS_E_CANTINITINPUT	An input cannot be created to initialize the inputs collection.
0xC0208021	-1071611871	DTS_E_CANTINITOUTPUT	An output cannot be created to initialize the outputs collection.
0xC0208023	-1071611869	DTS_E_EXTRACTORCANTWRITE	Writing to the file "%1" failed with error code 0x%2!8.X!.
0xC0208024	-1071611868	DTS_E_INCORRECTCONNECTIONOBJECTTYPE	The connection manager "%1" returned an object of an incorrect type from the AcquireConnection method.
0xC0208025	-1071611867	DTS_E_INPUTCOLPROPERTYNOTFOUND	The "%3" property is required on input column "%1" (%2!d!), but is not found. The missing property should be added.
0xC0208026	-1071611866	DTS_E_EXTRACTORUNREFERENCED	The "%1" is marked read-only, but is not referenced by any other column. Unreferenced columns are not allowed.
0xC0208027	-1071611865	DTS_E_EXTRACTORREFERENCEDCOLUMNNOTFOUND	The "%1" references column ID %2!d!, and that column is not found on the input. A reference points to a nonexistent column.
0xC0208028	-1071611864	DTS_E_EXTRACTORDATACOLUMNNOTBLOB	The "%1" references "%2", and that column is not of a BLOB type.
0xC0208029	-1071611863	DTS_E_INSERTERREFERENCEDCOLUMNNOTFOUND	The "%1" references output column ID %2!d!, and that column is not found on the output.
0xC020802A	-1071611862	DTS_E_INSERTERCANTREAD	Reading from the file "%1" failed with error code 0x%2!8.X!.
0xC020802B	-1071611861	DTS_E_TXSCD_NOTYPEDCOLUMNNSATINPUT	There must be at least one column of Fixed, Changing, or Historical type on the input of a Slowly Changing Dimension transform. Verify that at least one column is a FixedAttribute, ChangingAttribute, or HistoricalAttribute.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC020802C	-1071611860	DTS_E_TXSCD_INVALIDINPUTCOLUMNTYPE	The ColumnType property of "%1" is not valid. The current value is outside the range of acceptable values.
0xC020802D	-1071611859	DTS_E_TXSCD_CANNOTMAPDIFFERENTTYPES	The input column "%1" cannot be mapped to external column "%2" because they have different data types. The Slowly Changing Dimension transform does not allow mapping between column of different types except for DT_STR and DT_WSTR.
0xC020802E	-1071611858	DTS_E_NTEXTDATATYPENOTSUPPORTEDWITHANSIFILES	The data type for "%1" is DT_NTEXT, which is not supported with ANSI files. Use DT_TEXT instead and convert the data to DT_NTEXT using the data conversion component.
0xC020802F	-1071611857	DTS_E_TEXTDATATYPENOTSUPPORTEDWITHUNICODEFILES	The data type for "%1" is DT_TEXT, which is not supported with Unicode files. Use DT_NTEXT instead and convert the data to DT_TEXT using the data conversion component.
0xC0208030	-1071611856	DTS_E_IMAGEDATATYPENOTSUPPORTED	The data type for "%1" is DT_IMAGE, which is not supported. Use DT_TEXT or DT_NTEXT instead and convert the data from, or to, DT_IMAGE using the data conversion component.
0xC0208031	-1071611855	DTS_E_FLATFILEFORMATNOTSUPPORTED	Format "%1" is not supported by Flat File Connection Manager. Supported formats are Delimited, FixedWidth, RaggedRight, and Mixed.
0xC0208032	-1071611854	DTS_E_EXTRACTORFILENAMEMISMATCHCOLUMNTYPE	The "%1" should contain a file name, but it is not of a String type.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0208033	-1071611853	DTS_E_EXTRACTORCANTAPPENDTRUNCATE	Error caused by conflicting property settings. The "%1" has both the AllowAppend property and the ForceTruncate property set to TRUE. Both properties cannot be set to TRUE. Set one of the two properties to FALSE.
0xC0208034	-1071611852	DTS_E_EXTRACTORCOLUMNALREADYREFERENCED	The %1 references column ID %2!d!, but that column is already referenced by %3. Remove one of the two reference to the column.
0xC0208035	-1071611851	DTS_E_CONNECTIONMANAGERNOTASSIGNED	A connection manager has not been assigned to the %1.
0xC0208036	-1071611850	DTS_E_INSERTERCOLUMNALREADYREFERENCED	The %1 references the output column with ID %2!d!, but that column is already referenced by %3.
0xC0208037	-1071611849	DTS_E_INSERTERCOLUMNNOTREFERENCED	The "%1" is not referenced by any input column. Each output column must be referenced by exactly one input column.
0xC0208038	-1071611848	DTS_E_INSERTERDATACOLUMNNOTBLOB	The "%1" references "%2", and that column is not the correct type. It must be DT_TEXT, DT_NTEXT, or DT_IMAGE. A reference points to a column that must be a BLOB.
0xC0208039	-1071611847	DTS_E_INSERTERFILENAMECOLUMNNOTSTRING	The "%1" should contain a file name, but it is not a String type.
0xC020803A	-1071611846	DTS_E_INSERTEREXPECTBOMINVALIDTYPE	The "%1" has the ExpectBOM property set to TRUE for %2, but the column is not NT_NTEXT. The ExpectBOM specifies that the Import Column transformation expects a byte-order mark (BOM). Either set the ExpectBOM property to false or change the output column data type to DT_NTEXT.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC020803B	-1071611845	DTS_E_INSERTERINVALIDDATA COLUMNSETTYPE	Data output columns must be DT_TEXT, DT_NTEXT, or DT_IMAGE. The data output column may only be set to a BLOB type.
0xC020803C	-1071611844	DTS_E_TXSCD_FIXEDATTRIBUTECHANGE	If the FailOnFixedAttributeChange property is set to TRUE, the transformation will fail when a fixed attribute change is detected. To send rows to the Fixed Attribute output, set the FailOnFixedAttributeChange property to FALSE.
0xC020803D	-1071611843	DTS_E_TXSCD_LOOKUPFAILURE	The Lookup transformation failed to retrieve any rows. The transform fails when the FailOnLookupFailure is set to TRUE and no rows are retrieved.
0xC020803E	-1071611842	DTS_E_TXSCD_INVALIDNUMBERSOFPARAMETERS	There must be at least one column type of Key on the input of a Slowly Changing Dimension transformation. Set at least one column type to Key.
0xC020803F	-1071611841	DTS_E_TXSCD_CANNOTFINDEXTERNALCOLUMN	Cannot find external column with name "%1".
0xC0208040	-1071611840	DTS_E_TXSCD_INFERREDINDICATORNOTBOOL	Inferred indicator column "%1" must be of type DT_BOOL.
0xC0208107	-1071611641	DTS_E_ERRORROWDISPOSITIONNOTUSED	The %1 must have its error row disposition value set to RD_NotUsed.
0xC0208108	-1071611640	DTS_E_TRUNCROWDISPOSITIONNOTUSED	The %1 must have its truncation row disposition value set to RD_NotUsed.
0xC0208201	-1071611391	DTS_E_TXAGG_INPUTNOTFOUNDFOROUTPUT	Cannot find input column with lineage ID %1!d! needed by output column with ID %2!d!.
0xC0208202	-1071611390	DTS_E_TXAGG_INVALIDOUTPUTDATATYPEFORAGGREGATE	Invalid output data type for aggregate type specified at output column ID %1!d!.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0208203	-1071611389	DTS_E_TXAGG_INVALIDINPUTDATATYPEFORAGGREGATE	Invalid input data type for %1 used for the specified aggregate at %2.
0xC0208204	-1071611388	DTS_E_TXAGG_INPUTOUTPUTDATATYPEMISMATCH	Data types of input column lineage ID %1!d! and output column ID %2!d! do not match.
0xC0208205	-1071611387	DTS_E_UNABLETOGETINPUTBUFFERHANDLE	Cannot get input buffer handle for input ID %1!d!.
0xC0208206	-1071611386	DTS_E_UNABLETOGETOUTPUTBUFFERHANDLE	Cannot get output buffer handle for output ID %1!d!.
0xC0208207	-1071611385	DTS_E_UNABLETOFINDCOLUMNHANDLEINOUTPUTBUFFER	Cannot find column with lineage ID %1!d! in output buffer.
0xC0208208	-1071611384	DTS_E_UNABLETOFINDCOLUMNHANDLEININPUTBUFFER	Cannot find column with lineage ID %1!d! in input buffer.
0xC0208209	-1071611383	DTS_E_CANNOTHAVEZEROOUTPUTCOLUMNS	The number of output columns for %1 cannot be zero.
0xC020820A	-1071611382	DTS_E_CONNECTIONMANAGERCOLUMNCOUNTMISMATCH	The number of columns in the flat file connection manager must be the same as the number of columns in the flat file adapter. The number of columns for the flat file connection manager is %1!d!, while the number of columns for the flat file adapter is %2!d!.
0xC020820B	-1071611381	DTS_E_MISMATCHCONNECTIONMANAGERCOLUMN	The column "%1" at index %2!d! in the flat file connection manager was not found at index %3!d! in the column collection of the flat file adapter.
0xC020820D	-1071611379	DTS_E_EXTERNALMETADATACOLUMNISALREADYMAPPED	The external metadata column with ID %1!d! has already been mapped to %2.
0xC020820E	-1071611378	DTS_E_TXAGG_STRING_TOO_LONG	The transform encountered a key column that was larger than %1!u! characters.
0xC020820F	-1071611377	DTS_E_DERIVEDRESULT_TOO_LONG	The transform encountered a result value that was longer than %1!u! bytes.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0208210	-1071611376	DTS_E_TXAGG_MEMALLOCE RROUTPUTDESCRIPTORS	Unable to allocate memory.
0xC0208211	-1071611375	DTS_E_TXAGG_MEMALLOCE RRWORKSPACEDESCRIPTOR S	Unable to allocate memory.
0xC0208212	-1071611374	DTS_E_TXAGG_MEMALLOCE RRSORTORDERDESCRIPTOR S	Unable to allocate memory.
0xC0208213	-1071611373	DTS_E_TXAGG_MEMALLOCE RRNUMERICDESCRIPTORS	Unable to allocate memory.
0xC0208214	-1071611372	DTS_E_TXAGG_MEMALLOCE RRCOUNTDISTINCTDESCRIP TOR	Unable to allocate memory.
0xC0208215	-1071611371	DTS_E_TXAGG_MEMALLOCE RRWORKSPACESORTORDER DESCRIPTORS	Unable to allocate memory.
0xC0208216	-1071611370	DTS_E_TXAGG_MEMALLOCE RRWORKSPACENUMERICDE SCRIPTORS	Unable to allocate memory.
0xC0208217	-1071611369	DTS_E_TXAGG_MEMALLOCE RRWORKSPACEBUFFCOLS	Unable to allocate memory.
0xC0208218	-1071611368	DTS_E_UNREFERENCEDINPU TCOLUMN	The input column "%1" is not referenced.
0xC0208219	-1071611367	DTS_E_CANTBUILDTHREADP OOL	The Sort transformation could not create a thread pool with %1!d! threads. Not enough memory is available.
0xC020821A	-1071611366	DTS_E_QUEUEWORKITEMFA ILED	The Sort transformation cannot queue a work item to its thread pool. There is not enough memory available.
0xC020821B	-1071611365	DTS_E_SORTTHREADSTOPPE D	A worker thread in the Sort transformation stopped with error code 0x%1!8X!. A catastrophic error was encountered while sorting a buffer.
0xC020821E	-1071611362	DTS_E_SORTBADTHREADCO UNT	MaxThreads was %1!d!, and should be between 1 and %2!d!, inclusive or -1 to default to the number of CPUs.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC020821F	-1071611361	DTS_E_DTPXMLLOADFAILURE	Unable to load from XML.
0xC0208220	-1071611360	DTS_E_DTPXMLSAVEFAILURE	Unable to save to XML.
0xC0208221	-1071611359	DTS_E_DTPXMLINT32CONVERTERR	Unable to convert the value "%1" to an integer.
0xC0208222	-1071611358	DTS_E_DTPXMLBOOLCONVERTERR	Unable to convert the value "%1" to a Boolean.
0xC0208223	-1071611357	DTS_E_DTPXMLPARSEERRORNEARID	Load error encountered near object with ID %1!d!.
0xC0208226	-1071611354	DTS_E_DTPXMLPROPERTYTYPEERR	The value "%1" is not valid for the attribute "%2".
0xC0208228	-1071611352	DTS_E_DTPXMLSETUSAGETYPEERR	The value "%1" is not valid for the attribute "%2".
0xC0208229	-1071611351	DTS_E_DTPXMLDATATYPEERR	The value "%1" is not valid for the attribute "%2".
0xC020822A	-1071611350	DTS_E_UNMAPPEDINPUTCOLUMN	The %1 is not mapped to an output column.
0xC020822B	-1071611349	DTS_E_INPUTCOLUMNBADMAP	The %1 has a mapping that is not valid. An output column with an ID of %2!d! does not exist on this component.
0xC020822D	-1071611347	DTS_E_MULTIPLYMAPPEDOUTPUTCOL	The %1 is mapped to an output column that already has a mapping on this input.
0xC020822E	-1071611346	DTS_E_TXAGG_STRINGPRODUCTIONFAILED	Could not convert input column with Lineage ID %1!d! to DT_WSTR due to error 0x%2!8.X!.
0xC0208230	-1071611344	DTS_E_DTPXMLIDLOOKUPERROR	Referenced object with ID %1!d! not found in package.
0xC0208231	-1071611343	DTS_E_DTPXMLINVALIDXMLPERSISTPROPERTY	Cannot read a persistence property required for the pipelinexml module. The property was not provided by the pipeline.
0xC0208232	-1071611342	DTS_E_DTPXMLPROPERTYSTATEERR	The value "%1" is not valid for the attribute "%2".

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0208233	-1071611341	DTS_E_CANTGETCUSTOMPROPERTY	Cannot retrieve custom property "%1".
0xC0208234	-1071611340	DTS_E_UNABLETOLOCATEINPUTCOLUMNID	An input column with the lineage ID %1d!, referenced in the ParameterMap custom property with the parameter on position number %2d!, cannot be found in the input columns collection.
0xC0208235	-1071611339	DTS_E_TXLOOKUP_UNABLETOLOCATEREFCOLUMN	Unable to locate reference column "%1".
0xC0208236	-1071611338	DTS_E_TXLOOKUP_INCOMPATIBLEDATATYPES	%1 and reference column named "%2" have incompatible data types.
0xC0208237	-1071611337	DTS_E_TXLOOKUP_PARAMETERMETADATAMISMATCH	The parameterized SQL statement yields metadata which does not match the main SQL statement.
0xC0208238	-1071611336	DTS_E_TXLOOKUP_INCORRECTNUMOFPARAMETERS	The parameterized SQL statement contains an incorrect number of parameters. Expected %1d!, but found %2d!.
0xC0208239	-1071611335	DTS_E_TXLOOKUP_INVALIDJOINTYPE	%1 has a datatype which cannot be joined on.
0xC020823A	-1071611334	DTS_E_TXLOOKUP_INVALIDCOPYTYPE	%1 has a datatype which cannot be copied.
0xC020823B	-1071611333	DTS_E_INSERTERINVALIDCOLUMNDATATYPE	The %1 has an unsupported datatype. It must be DT_STR or DT_WSTR.
0xC020823C	-1071611332	DTS_E_EXTRACTORINVALIDCOLUMNDATATYPE	The %1 has an unsupported datatype. It must be DT_STR, DT_WSTR, DT_TEXT, DT_NTEXT, or DT_IMAGE.
0xC020823D	-1071611331	DTS_E_TXCHARMAPINVALIDCOLUMNDATATYPE	The %1 has an unsupported datatype. It must be DT_STR, DT_WSTR, DT_TEXT, or DT_NTEXT.
0xC020823E	-1071611330	DTS_E_SORTCANTCREATEEVENT	The Sort transformation cannot create an event to communicate with its worker threads. Not enough system handles are available to the Sort transformation.



HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC020823F	-1071611329	DTS_E_SORTCANTCREATETH READ	The Sort transformation cannot create a worker thread. Not enough memory is available to Sort transformation.
0xC0208240	-1071611328	DTS_E_SORTCANTCOMPARE	The Sort transformation failed to compare row %1!d! in buffer ID %2!d! to row %3!d! in buffer ID %4!d!.
0xC0208242	-1071611326	DTS_E_TXLOOKUP_TOOFEW REFERENCECOLUMNS	The Lookup transformation reference metadata contains too few columns. Check the SQLCommand property. The SELECT statement must return at least one column.
0xC0208243	-1071611325	DTS_E_TXLOOKUP_MALLO C_ERR_REFERENCECOLUM NINFO	Unable to allocate memory for an array of ColumnInfo structures.
0xC0208244	-1071611324	DTS_E_TXLOOKUP_MALLO C_ERR_REFERENCECOLUM NPAIR	Could not allocate memory for an array of ColumnPair structures.
0xC0208245	-1071611323	DTS_E_TXLOOKUP_MALLO C_ERR_BUFFCOL	Unable to allocate memory for an array of BUFFCOL structures for the creation of a main workspace.
0xC0208246	-1071611322	DTS_E_TXLOOKUP_MAINW ORKSPACE_CREATEERR	Unable to create a main workspace buffer.
0xC0208247	-1071611321	DTS_E_TXLOOKUP_HASHT ABLE_MALLOCCERR	Unable to allocate memory for hash table.
0xC0208248	-1071611320	DTS_E_TXLOOKUP_HASHN ODE_HEAP_CREATEERR	Unable to allocate memory to create a heap for hash nodes.
0xC0208249	-1071611319	DTS_E_TXLOOKUP_HASHN ODE_HEAP_MALLOCCERR	Unable to allocate memory for a hash node heap.
0xC020824A	-1071611318	DTS_E_TXLOOKUP_LRU NODE_HEAP_CREATEERR	Unable to create a heap for LRU nodes. An out-of-memory condition occurred.
0xC020824B	-1071611317	DTS_E_TXLOOKUP_LRU NODE_HEAP_MALLOCCERR	Unable to allocate memory for the LRU node heap. An out-of-memory condition occurred.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC020824C	-1071611316	DTS_E_TXLOOKUP_OLEDBE RR_LOADCOLUMNMETADA TA	OLE DB error occurred while loading column metadata. Check SQLCommand and SqlCommandParam properties.
0xC020824D	-1071611315	DTS_E_TXLOOKUP_OLEDBE RR_GETIROWSET	OLE DB error occurred while fetching rowset. Check SQLCommand and SqlCommandParam properties.
0xC020824E	-1071611314	DTS_E_TXLOOKUP_OLEDBE RR_FILLBUFFER	OLE DB error occurred while populating internal cache. Check SQLCommand and SqlCommandParam properties.
0xC020824F	-1071611313	DTS_E_TXLOOKUP_OLEDBE RR_BINDPARAMETERS	OLE DB error occurred while binding parameters. Check SQLCommand and SqlCommandParam properties.
0xC0208250	-1071611312	DTS_E_TXLOOKUP_OLEDBE RR_CREATEBINDING	OLE DB error occurred while creating bindings. Check SQLCommand and SqlCommandParam properties.
0xC0208251	-1071611311	DTS_E_TXLOOKUP_INVALID _CASE	An invalid case was encountered in a switch statement during runtime.
0xC0208252	-1071611310	DTS_E_TXLOOKUP_MAINW ORKSPACE_MALLOCERR	Unable to allocate memory for a new row for the main workspace buffer. An out-of-memory condition occurred.
0xC0208253	-1071611309	DTS_E_TXLOOKUP_OLEDBE RR_GETPARAMIROWSET	OLE DB error occurred while fetching parameterized rowset. Check SQLCommand and SqlCommandParam properties.
0xC0208254	-1071611308	DTS_E_TXLOOKUP_OLEDBE RR_GETPARAMSINGLEROW	OLE DB error occurred while fetching parameterized row. Check SQLCommand and SqlCommandParam properties.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0208255	-1071611307	DTS_E_TXAGG_MAINWORKSPACE_MALLOCERR	Unable to allocate memory for a new row for the main workspace buffer. An out-of-memory condition occurred.
0xC0208256	-1071611306	DTS_E_TXAGG_MAINWORKSPACE_CREATEERR	Unable to create a main workspace buffer.
0xC0208257	-1071611305	DTS_E_TXAGG_HASHTABLE_MALLOCERR	Unable to allocate memory for the hash table.
0xC0208258	-1071611304	DTS_E_TXAGG_HASHNODE_HEAP_CREATEERR	Unable to allocate memory to create a heap for the hash nodes.
0xC0208259	-1071611303	DTS_E_TXAGG_HASHNODE_HEAP_MALLOCERR	Unable to allocate memory for the hash node heap.
0xC020825A	-1071611302	DTS_E_TXAGG_CDNODEHEAP_CREATEERR	Unable to allocate memory to create a heap for CountDistinct nodes.
0xC020825B	-1071611301	DTS_E_TXAGG_CDNODEHEAP_MALLOCERR	Unable to allocate memory for CountDistinct node heap.
0xC020825C	-1071611300	DTS_E_TXAGG_CDCHAINHEAD_CREATEERR	Unable to allocate memory to create a heap for CountDistinct chains.
0xC020825D	-1071611299	DTS_E_TXAGG_CDHASHTABLE_CREATEERR	Unable to allocate memory for CountDistinct hash table.
0xC020825E	-1071611298	DTS_E_TXAGG_CDWORKSPACE_MALLOCERR	Unable to allocate memory for a new row for the CountDistinct workspace buffer.
0xC020825F	-1071611297	DTS_E_TXAGG_CDWORKSPACE_CREATEERR	Unable to create a CountDistinct workspace buffer.
0xC0208260	-1071611296	DTS_E_TXAGG_CDCOLLAPSE_ARRAY_MALLOCERR	Unable to allocate memory for CountDistinct Collapse array.
0xC0208261	-1071611295	DTS_E_TXAGG_CDCHAINHEAD_MALLOCERR	Unable to allocate memory for CountDistinct chains.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0208262	-1071611294	DTS_E_TXCOPYMAP_MISMATCHED_COLUMN_METADATA	Columns with lineage IDs %1!d! and %2!d! have mismatched metadata. The input column that is mapped to an output column for copymap does not have the same metadata (datatype, precision, scale, length, or codepage).
0xC0208263	-1071611293	DTS_E_TXCOPYMAP_INCORRECT_OUTPUT_COLUMN_MAPPING	The output column with lineage ID "%1!d!" is incorrectly mapped to an input column. The CopyColumnId property of the output column is not correct.
0xC0208265	-1071611291	DTS_E_CANTGETBLOBDATA	Failed to retrieve long data for column "%1".
0xC0208266	-1071611290	DTS_E_CANTADDBLOBDATA	Long data was retrieved for a column but cannot be added to the Data Flow task buffer.
0xC0208267	-1071611289	DTS_E_MCASTOUTPUTCOLUMNNS	Output "%1" (%2!d!) has output columns, but multicast outputs do not declare columns. The package is damaged.
0xC0208273	-1071611277	DTS_E_UNABLETOGETLOCALIZEDRESOURCE	Unable to load a localized resource ID %1!d!. Verify that the RLL file is present.
0xC0208274	-1071611276	DTS_E_DTPXMLEVENTSCACHEERR	Cannot acquire Events Interface. An invalid Events interface was passed to the data flow module for persisting to XML.
0xC0208275	-1071611275	DTS_E_DTPXMLPATHLOADERR	An error occurred while setting a path object during XML load.
0xC0208276	-1071611274	DTS_E_DTPXMLINPUTLOADERR	Error setting input object during XML load.
0xC0208277	-1071611273	DTS_E_DTPXMLOUTPUTLOADERR	Error setting output object during XML load.
0xC0208278	-1071611272	DTS_E_DTPXMLINPUTCOLUMNLOADERR	Error setting input column object during XML load.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0208279	-1071611271	DTS_E_DTPXMLOUTPUTCOL UMNLOADERR	Error setting output column object during XML load.
0xC0208280	-1071611264	DTS_E_DTPXMLPROPERTYL OADERR	Error setting property object during XML load.
0xC0208281	-1071611263	DTS_E_DTPXMLCONNECTIO NLOADERR	Error setting connection object during XML load.
0xC0208282	-1071611262	DTS_E_FG_MISSING_OUTPU T_COLUMNS	Special transformation- specific columns are either missing or have incorrect types.
0xC0208283	-1071611261	DTS_E_FG_PREPARE_TABLES _AND_ACCESSORS	Fuzzy Grouping transformation failed to create required tables and accessors.
0xC0208284	-1071611260	DTS_E_FG_COPY_INPUT	Fuzzy Grouping transformation failed to copy input.
0xC0208285	-1071611259	DTS_E_FG_GENERATE_GROU PS	Fuzzy Grouping transformation failed to generate groups.
0xC0208286	-1071611258	DTS_E_FG_LEADING_TRAILI NG	An unexpected error occurred in Fuzzy Grouping when applying the settings of property '%1'.
0xC0208287	-1071611257	DTS_E_FG_PICK_CANONICA L	The Fuzzy Grouping transformation failed to pick a canonical row of data to use in standardizing the data.
0xC0208288	-1071611256	DTS_E_FG_NOBLOBS	Fuzzy Grouping does not support input columns of type IMAGE, TEXT, or NTEXT.
0xC0208289	-1071611255	DTS_E_FG_FUZZY_MATCH_ ON_NONSTRING	A fuzzy match is specified on column "%1" (%2!d!) that is not a data type of DT_STR or DT_WSTR.
0xC020828A	-1071611254	DTS_E_FUZZYGROUPINGINT ERNALPIPELINEERROR	A Fuzzy Grouping transformation pipeline error occurred and returned error code 0x%1!8.X!: "%2".

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC020828B	-1071611253	DTS_E_CODE_PAGE_NOT_SUPPORTED	The code page %1!d! specified on column "%2" (%3!d!) is not supported. You must first convert this column to DT_WSTR which can be done by inserting a Data Conversion Transform before this one.
0xC0208294	-1071611244	DTS_E_SETEODFAILED	Failure encountered while setting end of data flag for the buffer driving output "%1" (%2!d!).
0xC0208296	-1071611242	DTS_E_CANTCLONE	The input buffer could not be cloned. An out-of-memory condition occurred or there was an internal error.
0xC02082F9	-1071611143	DTS_E_TXCHARMAP_CANTKATAKANAHIRAGANA	Column "%1" requests that Katakana and Hiragana characters be produced at the same time.
0xC02082FA	-1071611142	DTS_E_TXCHARMAP_CANTSIMPLECHINESEANDTRADITIONALCHINESE	Column "%1" requests that Simple Chinese and Traditional Chinese characters be produced at the same time.
0xC02082FB	-1071611141	DTS_E_TXCHARMAP_CANTFULLWIDTHANDHALFWIDTH	Column "%1" requests operations to generate both full width and half width characters.
0xC02082FC	-1071611140	DTS_E_TXCHARMAP_CANTJAPANESEANDCHINESE	Column "%1" combines operations on Japanese characters with operations for Chinese characters.
0xC02082FD	-1071611139	DTS_E_TXCHARMAP_CANTCHINESEUPPERCASEANDLOWERCASE	Column "%1" combines operations on Chinese characters with uppercase and lowercase operations.
0xC02082FE	-1071611138	DTS_E_TXCHARMAP_CANTJAPANESEUPPERCASEANDLOWERCASE	Column "%1" combines operations on Japanese characters with uppercase and lowercase operations.
0xC02082FF	-1071611137	DTS_E_TXCHARMAP_CANTBOTHUPPERCASEANDLOWERCASE	Column "%1" maps the column to both uppercase and lowercase.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0208300	-1071611136	DTS_E_TXCHARMAP_CANTLI NGUISTIC	Column "%1" combines flags other than uppercase and lowercase with the linguistic casing operation.
0xC0208301	-1071611135	DTS_E_TXCHARMAP_INVALI DMAPFLAGANDDATATYPE	The data type of column "%1" cannot be mapped as specified.
0xC0208302	-1071611134	DTS_E_TXFUZZYLOOKUP_U NSUPPORTED_MATCH_IND EX_VERSION	The version (%1) of the pre-existing match index "%2" is not supported. The version expected is "%3". This error occurs if the version persisted in the index metadata does not match the version which the current code was built for. Fix the error by rebuilding the index with the current version of the code.
0xC0208303	-1071611133	DTS_E_TXFUZZYLOOKUP_IN VALID_MATCH_INDEX	The table "%1" does not appear to be a valid pre-built match index. This error occurs if the metadata record cannot be loaded from the specified pre-built index.
0xC0208304	-1071611132	DTS_E_TXFUZZYLOOKUP_U NABLE_TO_READ_MATCH_I NDEX	Unable to read specified pre-built match index "%1". OLEDB Error code: 0x%2!8.8X!.
0xC0208305	-1071611131	DTS_E_TXFUZZYLOOKUP_N O_JOIN_COLUMNS	There were no input columns with a valid join to a reference table column. Make sure that there is at least one join defined using the input column properties JoinToReferenceColumn and JoinType.
0xC0208306	-1071611130	DTS_E_TXFUZZYLOOKUP_IN DEX_DOES_NOT_CONTAIN_ COLUMN	The specified pre-existing match index "%1" was not originally built with fuzzy match information for column "%2". It must be rebuilt to include this information. This error occurs when the index was built with the column not being a fuzzy join column.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0208307	-1071611129	DTS_E_TXFUZZYLOOKUP_ID ENTIFIER_PROPERTY	The name "%1" given for property "%2" is not a valid SQL identifier name. This error occurs if the name for the property does not conform to the specifications for a valid SQL identifier name.
0xC0208309	-1071611127	DTS_E_TXFUZZYLOOKUP_M INSIMILARITY_INVALID	The MinSimilarity threshold property on the Fuzzy Lookup transformation must be a value greater than or equal to 0.0 but less than 1.0.
0xC020830A	-1071611126	DTS_E_TXFUZZYLOOKUP_IN VALID_PROPERTY_VALUE	The value "%1" for property "%2" is not valid.
0xC020830B	-1071611125	DTS_E_TXFUZZYLOOKUP_IN COMPATIBLE_FUZZY_JOIN_ DATATYPES	The fuzzy lookup specified between input column "%1" and reference column "%2" is not valid because fuzzy joins are only supported between string columns, types DT_STR and DT_WSTR.
0xC020830C	-1071611124	DTS_E_TXFUZZYLOOKUP_IN COMPATIBLE_EXACT_JOIN_ DATATYPES	The exact lookup columns, "%1" and "%2", do not have equal data types or are not comparable string types. Exact joins are supported between columns with equal data types or a DT_STR and DT_WSTR combination.
0xC020830D	-1071611123	DTS_E_TXFUZZYLOOKUP_IN COMPATIBLE_COPYCOLUM N_DATATYPES	The copy columns, "%1" and "%2", do not have equal data types or are not trivially convertible string types. This occurs because copying from reference to output between columns with equal data types, or a DT_STR and DT_WSTR combination, is supported, but other types are not.
0xC020830E	-1071611122	DTS_E_TXFUZZYLOOKUP_IN COMPATIBLE_PASSTHRUCO LUMN_DATATYPES	The passthrough columns, "%1" and "%2", do not have equal data types. Only columns with equal data types are supported as passthrough columns from input to output.



HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC020830F	-1071611121	DTS_E_TXFUZZYLOOKUP_UNABLETOLOCATEREFCOLUMN	Cannot locate reference column "%1".
0xC0208311	-1071611119	DTS_E_TXFUZZYLOOKUP_OUTPUT_COLUMN_MUST_BE_PASSTHRU_COLUMN_OR_A_COPY_COLUMN	An output column must have exactly one CopyColumn or PassThruColumn property specified. This error occurs when neither the CopyColumn or the PassThruColumn properties, or both the CopyColumn and PassThruColumn properties, are set to non-empty values.
0xC0208312	-1071611118	DTS_E_TXFUZZYLOOKUP_PASSTHRU_COLUMN_NOT_FOUND	The source lineage id '%1!d!' specified for property '%2' on output column '%3' was not found in the input column collection. This occurs when the input column id specified on an output column as a passthrough column is not found in the set of inputs.
0xC0208313	-1071611117	DTS_E_TXFUZZYLOOKUP_INDEXED_COLUMN_NOT_FOUND_IN_REF_TABLE	The column "%1" in the pre-built index "%2" was not found in the reference table/query. This happens if the schema/query of the reference table has changed since the pre-existing match index was built.
0xC0208314	-1071611116	DTS_E_TXFUZZYLOOKUP_TOKEN_TOO_LONG	The component encountered a token that was larger than 2147483647 characters.
0xC0208315	-1071611115	DTS_E_RAWMETADATAMISMATCHTYPE	The output file cannot be appended. Column "%1" matches by name, but the column in the file has type %2 and the input column has type %3. The metadata for the column does not match on data type.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0208316	-1071611114	DTS_E_RAWMETADATAMIS MATCHSIZE	The output file cannot be appended. Column "%1" matches by name, but the column in the file has maximum length %2!d! and the input column has maximum length %3!d!. The metadata for the column does not match in length.
0xC0208317	-1071611113	DTS_E_RAWMETADATAMIS MATCHCODEPAGE	The output file cannot be appended. Column "%1" matches by name, but the column in the file has code page %2!d! and the input column has code page %3!d!. The metadata for the named column does not match on code page.
0xC0208318	-1071611112	DTS_E_RAWMETADATAMIS MATCHPRECISION	The output file cannot be appended. Column "%1" matches by name, but the column in the file has precision %2!d! and the input column has precision %3!d!. The metadata for the named column does not match on precision.
0xC0208319	-1071611111	DTS_E_RAWMETADATAMIS MATCHSCALE	The output file cannot be appended. Column "%1" matches by name, but the column in the file has scale %2!d! and the input column has scale %3!d!. The metadata for the named column does not match on scale.
0xC020831A	-1071611110	DTS_E_COULD_NOT_DETER MINE_DATASOURCE_DBMS NAME	Unable to determine the DBMS name and version on "%1". This occurs if the IDBProperties on the connection did not return information needed to verify the DBMS name and version.
0xC020831B	-1071611109	DTS_E_INCORRECT_SQL_SER VER_VERSION	The DBMS type or version of "%1" is not supported. A connection to Microsoft SQL Server version 8.0 or later is required. This occurs if IDBProperties on the connection did not return a the correct version.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC020831D	-1071611107	DTS_E_CANTDELETEERRORCOLUMNS	The %1 is a special error output column and cannot be deleted.
0xC020831E	-1071611106	DTS_E_UNEXPECTEDCOLUMN DATATYPE	The data type specified for column "%1" is not the expected type "%2".
0xC020831F	-1071611105	DTS_E_INPUTCOLUMNNOT FOUND	The input column lineage ID "%1" referenced by property "%2" on output column "%3" could not be located in the input column collection.
0xC0208320	-1071611104	DTS_E_TXGROUPDUPS_INPUTCOLUMNNOTJOINED	The input column "%1" referenced by the "%2" property on output column "%3" must have property ToBeCleaned=True and have a valid ExactFuzzy property value.
0xC0208322	-1071611102	DTS_E_TXFUZZYLOOKUP_REF_TABLE_MISSING_IDENTITY_INDEX	The reference table '%1' does not have a clustered index on an integer identity column, which is required if the property 'CopyRefTable' is set to FALSE. If CopyRefTable is false, the reference table must have a clustered index on an integer identity column.
0xC0208323	-1071611101	DTS_E_TXFUZZYLOOKUP_REF_CONTAINS_NON_INTEGER_IDENT_COLUMN	The reference table '%1' contains a non-integer type identity column which is not supported. Use a view of the table without the column '%2'. This error occurs because when a copy is made of the reference table, an integer identity column is added, and only one identity column is allowed per table.
0xC0208324	-1071611100	DTS_E_TXFUZZY_MATCHCONTRIBUTION_AND_HIERARCHY_SPECIFIED	Both MatchContribution and hierarchy information cannot be specified at the same time. This is not allowed because these properties are both weighing factors for scoring.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0208325	-1071611099	DTS_E_TXFUZZY_HIERARCHY_INCORRECT	Levels in hierarchy should be unique numbers . Valid level in hierarchy values are integers greater than or equal to 1. The smaller the number is, the lower the column is in the hierarchy. The default value is 0, indicating that the column is not part of a hierarchy. Overlaps and gaps are not allowed.
0xC0208326	-1071611098	DTS_E_TXFUZZYGROUPING_INSUFFICIENT_FUZZY_JOIN_COLUMNS	No columns to fuzzy group on were defined. There must be at least one input column with column properties ToBeCleaned=true and ExactFuzzy=2.
0xC0208329	-1071611095	DTS_E_TXFUZZYLOOKUP_COLUMNINVALID	The column with ID '%1!d!' was not valid for an undetermined reason.
0xC020832A	-1071611094	DTS_E_TXFUZZYLOOKUP_UNSUPPORTEDDATATYPE	The data type of column '%1' is not supported.
0xC020832C	-1071611092	DTS_E_TXFUZZYLOOKUP_OUTPUTLENGTHMISMATCH	The length of output column '%1' is less than that of its source column '%2'.
0xC020832F	-1071611089	DTS_E_TERMEXTRACTION_INCORRECTEXACTNUMBEROFINPUTCOLUMNS	There should be only one input column.
0xC0208330	-1071611088	DTS_E_TERMEXTRACTION_INCORRECTEXACTNUMBEROFOUTPUTCOLUMNS	There should be exactly two output columns.
0xC0208331	-1071611087	DTS_E_TERMEXTRACTION_INCORRECTDATATYPEOFINPUTCOLUMN	The input column can only have DT_WSTR or DT_NTEXT as its data type.
0xC0208332	-1071611086	DTS_E_TERMEXTRACTION_INCORRECTDATATYPEOFOUTPUTCOLUMN	The output column [%1!d!] can only have '%2' as its data type.
0xC0208333	-1071611085	DTS_E_TERMEXTRACTION_INCORRECTDATATYPEOFREFERENCECOLUMN	The reference column can only have DT_STR or DT_WSTR as its data type.
0xC0208334	-1071611084	DTS_E_TERMEXTRACTION_UNABLETOLOCATEREFCOLUMN	An error occurred while locating the reference column '%1'.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0208335	-1071611083	DTS_E_TERMEEXTRACTION_I NCORRECTTERMTYPE	The Term Type of the transformation can only be WordOnly, PhraseOnly or WordPhrase.
0xC0208336	-1071611082	DTS_E_TERMEEXTRACTION_I NCORRECTFREQUENCYTHRESHOLD	The value of Frequency Threshold should not be lower than '%1!d!'.
0xC0208337	-1071611081	DTS_E_TERMEEXTRACTION_I NCORRECTMAXLENOFTERM	The value of Max Length of Term should not be lower than '%1!d!'.
0xC0208338	-1071611080	DTS_E_TERMEEXTRACTION_T OOFEWREFERENCECOLUMNS	Term Extraction reference metadata contains too few columns.
0xC0208339	-1071611079	DTS_E_TERMEEXTRACTION_ MALLOCERR_REFERENCECOLUMNINFO	An error occurred while allocating memory.
0xC020833A	-1071611078	DTS_E_TERMEEXTRACTION_ MAINWORKSPACE_CREATEERROR	An error occurred while creating a workspace buffer.
0xC020833B	-1071611077	DTS_E_TERMEEXTRACTION_O LEDBERR_CREATEBINDING	An OLEDB error occurred while creating bindings.
0xC020833C	-1071611076	DTS_E_TERMEEXTRACTION_O LEDBERR_GETIROWSET	An OLEDB error occurred while fetching rowsets.
0xC020833D	-1071611075	DTS_E_TERMEEXTRACTION_O LEDBERR_FILLBUFFER	An OLEDB error occurred while populating internal cache.
0xC020833E	-1071611074	DTS_E_TERMEEXTRACTION_P ROCESSERR	An error occurred while extracting terms on row %1!d!, column %2!d!. The error code returned was 0x%3!8X!. Please remove it from the input as a work-around.
0xC020833F	-1071611073	DTS_E_TERMEEXTRACTIONO RLOOKUP_PROCESSERR_DEPOSITFULL	The number of the term candidates exceeds its limit, 4G.
0xC0208340	-1071611072	DTS_E_TERMEEXTRACTION_I NVALIDOUTTERMTABLEORCOLUMN	The reference table, view, or column that is used for Exclusion Terms is not valid.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0208341	-1071611071	DTS_E_TXFUZZYLOOKUP_ST RINGCOLUMNTOOLONG	The length of string column '%1' exceeds 4000 characters. A conversion from DT_STR to DT_WSTR is necessary, so a truncation would occur. Either reduce the column width or use only DT_WSTR column types.
0xC0208342	-1071611070	DTS_E_TERMEXTRACTION_O UTTERMTABLEANDCOLUM NNOTSET	The reference table, view, or column to be used for an Exclusion Terms has not been set.
0xC0208343	-1071611069	DTS_E_TERMLOOKUP_TOOF EWOUTPUTCOLUMNS	Term Lookup contains too few output columns.
0xC0208344	-1071611068	DTS_E_TERMLOOKUP_INCO RRECTDATATYPEOFREFEREN CECOLUMN	The reference column can only have DT_STR or DT_WSTR as its data type.
0xC0208345	-1071611067	DTS_E_TERMLOOKUP_UNAB LETOLOCATEREFCOLUMN	An error occurred while locating the reference column '%1'.
0xC0208346	-1071611066	DTS_E_TERMLOOKUP_TOOF EWREFERENCECOLUMNS	Term Lookup reference metadata contains too few columns.
0xC0208347	-1071611065	DTS_E_TERMEXTRACTIONO RLOOKUP_TESTOFFSETERR OR	An error occurred while normalizing words.
0xC0208348	-1071611064	DTS_E_TERMLOOKUP_MAIN WORKSPACE_CREATEERR	An error occurred while creating a workspace buffer.
0xC0208349	-1071611063	DTS_E_TERMLOOKUP_OLED BERR_CREATEBINDING	An OLEDB error occurred while creating bindings.
0xC020834A	-1071611062	DTS_E_TERMLOOKUP_OLED BERR_GETIROWSET	An OLEDB error occurred while fetching rowsets.
0xC020834B	-1071611061	DTS_E_TERMLOOKUP_OLED BERR_FILLBUFFER	An OLEDB error occurred while populating internal cache.
0xC020834C	-1071611060	DTS_E_TERMLOOKUP_PRO CESSERR	An error occurred while looking up terms on row %1!d!, column %2!d!. The error code returned was 0x%3!8.X!. Please remove it from the input as a work-around.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC020834D	-1071611059	DTS_E_TERMLOOKUP_TEXTI NINPUTCOLUMNNOTMAPP EDWITHOUTPUTCOLUMN	At least one Passthrough column is not mapped to an output column.
0xC020834E	-1071611058	DTS_E_TERMLOOKUP_INCO RRECTEXACTNUMBEROFTEX TCOLUMNS	There should be exactly one input column mapped to one reference column.
0xC020834F	-1071611057	DTS_E_TERMLOOKUP_TEXTI NPUTCOLUMNHAVEINCOR RECTDATATYPE	The input column mapped to a reference column can only have DT_NTXT or DT_WSTR as its data type.
0xC0208354	-1071611052	DTS_E_TXFUZZYLOOKUP_IN VALID_MATCH_INDEX_NAM E	The reference table name "%1" is not a valid SQL identifier. This error occurs if the table name cannot be parsed from the input string. There may be unquoted spaces in the name. Verify that the name is correctly quoted.
0xC0208355	-1071611051	DTS_E_TERMEEXTRACTION_T ERMFILTERSTARTITERATION ERROR	An error occurred while the Term Filter was starting its iteration.
0xC0208356	-1071611050	DTS_E_TERMEEXTRACTION_E MPYTERMRESULTEERROR	An error occurred while reclaiming the buffer used for caching terms. The error code returned was 0x%1!8.8X!.
0xC0208357	-1071611049	DTS_E_TERMEEXTRACTION_S TDLENGTHERROR	An std::length_error occurred from the STL containers.
0xC0208358	-1071611048	DTS_E_TERMLOOKUP_SAVE WORDWITHPUNCTERROR	An error occurred while saving words with punctuation characters. The error code returned was 0x%1!8.8X!.
0xC0208359	-1071611047	DTS_E_TERMLOOKUP_ADDR EFERENCETERM	An error occurred while processing the %1!ld!th reference term. The error code returned was 0x%2!8.8X!.. Please remove the reference term from your reference table as a work-around.
0xC020835A	-1071611046	DTS_E_TERMLOOKUP_SORR EFERENCETERM	An error occurred while sorting reference terms. The error code returned was 0x%1!8.8X!.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC020835B	-1071611045	DTS_E_TERMLOOKUP_COUNTTERM	An error occurred while counting term candidates. The error code returned was 0x%1!8.8X!.
0xC020835C	-1071611044	DTS_E_FUZZYLOOKUP_REFERENCECACHEFULL	Fuzzy Lookup was unable to load the entire reference table into main memory as is required when the Exhaustive property is enabled. Either we ran out of system memory or a limit was specified for MaxMemoryUsage which was not sufficient to load the reference table. Either set MaxMemoryUsage to 0 or increase it significantly. Alternatively, disable Exhaustive.
0xC020835D	-1071611043	DTS_E_TERMLOOKUP_INITIALIZE	An error occurred while initializing the engine of Term Lookup. The error code returned was 0x%1!8.8X!.
0xC020835E	-1071611042	DTS_E_TERMLOOKUP_PROCESSSENTENCE	An error occurred while processing sentences. The error code returned was 0x%1!8.8X!.
0xC020835F	-1071611041	DTS_E_TEXTMININGBASE_APPENDTOTEMPBUFFER	An error occurred while adding strings to an internal buffer. The error code returned was 0x%1!8.8X!.
0xC0208360	-1071611040	DTS_E_TERMEXTRACTION_SAVEPOSTAG	An error occurred while saving part-of-speech tags from an internal buffer. The error code returned was 0x%1!8.8X!.
0xC0208361	-1071611039	DTS_E_TERMEXTRACTION_COUNTTERM	An error occurred while counting term candidates. The error code returned was 0x%1!8.8X!.
0xC0208362	-1071611038	DTS_E_TERMEXTRACTION_INITPOSPROCESSOR	An error occurred while initializing the part-of-speech processor. The error code returned was 0x%1!8.8X!.
0xC0208363	-1071611037	DTS_E_TERMEXTRACTION_INITFSA	An error occurred while loading the finite state automata. The error code returned was 0x%1!8.8X!.



HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0208364	-1071611036	DTS_E_TERMEEXTRACTION_I NITIALIZE	An error occurred while initializing the engine of Term Extraction. The error code returned was 0x%1!8.8X!.
0xC0208365	-1071611035	DTS_E_TERMEEXTRACTION_P ROCESSSENTENCE	An error occurred while processing within a sentence. The error code returned was 0x%1!8.8X!.
0xC0208366	-1071611034	DTS_E_TERMEEXTRACTION_I NITPOSTAGVECTOR	An error occurred while initializing the part-of-speech processor. The error code returned was 0x%1!8.8X!.
0xC0208367	-1071611033	DTS_E_TERMEEXTRACTION_S AVEPTRSTRING	An error occurred while adding strings to an internal buffer. The error code returned was 0x%1!8.8X!.
0xC0208368	-1071611032	DTS_E_TERMEEXTRACTION_A DDWORDTODECODER	An error occurred while adding words to a statistical decoder. The error code returned was 0x%1!8.8X!.
0xC0208369	-1071611031	DTS_E_TERMEEXTRACTION_D ECODE	An error occurred while decoding for a sentence. The error code returned was 0x%1!8.8X!.
0xC020836A	-1071611030	DTS_E_TERMEEXTRACTION_S ETEXCLUDEDTERM	An error occurred while setting exclusion terms. The error code returned was 0x%1!8.8X!.
0xC020836B	-1071611029	DTS_E_TERMEEXTRACTION_P ROCESSDOCUMENT	An error occurred while processing a document in the input. The error code returned was 0x%1!8.8X!.
0xC020836C	-1071611028	DTS_E_TEXTMININGBASE_TE STPERIOD	An error occurred while testing whether a dot is a part of an acronym. The error code returned was 0x%1!8.8X!.
0xC020836D	-1071611027	DTS_E_TERMLOOKUP_ENGI NEADDRESSREFERENCETERM	An error occurred while setting reference terms. The error code returned was 0x%1!8.8X!.
0xC020836E	-1071611026	DTS_E_TERMLOOKUP_PROC ESSDOCUMENT	An error occurred while processing a document in the input. The error code returned was 0x%1!8.8X!.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC020836F	-1071611025	DTS_E_INVALIDBULKINSERT PROPERTYVALUE	The value for the property %1 is %2!d!, which is not allowed. The value must be greater than or equal to %3!d!.
0xC0208370	-1071611024	DTS_E_INVALIDBULKINSERT FIRSTROWLASTROWVALUES	The value for the property %1 is %2!d!, which must be less than or equal to the value of %3!d! for property %4.
0xC0208371	-1071611023	DTS_E_FUZZYLOOKUPUNAB LEDELETEEXISTINGMATC HINDEX	An error was encountered when trying to delete the existing fuzzy match index named "%1". It is possible that this table was not created by Fuzzy Lookup (or this version of Fuzzy Lookup), it has been damaged, or there is another problem. Try manually deleting the table named "%2" or specify a different name for the MatchIndexName property.
0xC0208372	-1071611022	DTS_E_TERMEEXTRACTION_I NCORRECTSCORETYPE	The Score Type of the transformation can only be Frequency or TFIDF.
0xC0208373	-1071611021	DTS_E_FUZZYLOOKUPREFT ABLETOOBIG	The reference table specified has too many rows. Fuzzy Lookup only works with reference tables having less than 1 billion rows. Consider using a smaller view of your reference table.
0xC0208374	-1071611020	DTS_E_FUZZYLOOKUPUNAB LEDETERMINEREFERENCE TABLESIZE	Unable to determine the size of the reference table '%1'. It is possible that this object is a view and not a table. Fuzzy Lookup does not support views when CopyReferentaceTable=false. Make sure that the table exists and that CopyReferenceTable=true.
0xC0208377	-1071611017	DTS_E_XMLSRCOUTPUTCOL UMNDATATYPENOTSUPPO RTED	The SSIS Data Flow Task data type "%1" on the %2 is not supported for the %3.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0208378	-1071611016	DTS_E_XMLSRCCANNOTFIN DCOLUMNTOSETDATATYPE	Unable to set data type properties for the output column with ID %1!d! on the output with ID %2!d!. The output or column could not be found.
0xC0208379	-1071611015	DTS_E_CUSTOMPROPERTYIS READONLY	The value of custom property "%1" on the %2 cannot be changed.
0xC020837A	-1071611014	DTS_E_OUTPUTCOLUMNHA SNOERRORCOLUMN	The %1 on the non-error output has no corresponding output column on the error output.
0xC020837B	-1071611013	DTS_E_ERRORCOLUMNHAS NOOUTPUTCOLUMN	The %1 on the error output has no corresponding output column on the non-error output.
0xC020837C	-1071611012	DTS_E_ERRORCOLUMNHASI NCORRECTPROPERTIES	The %1 on the error output has properties that do not match the properties of its corresponding data source column.
0xC020837D	-1071611011	DTS_E_ADOSRCOUTPUTCO LUMNDATATYPECANNOTBE CHANGED	The data type of output columns on the %1 cannot be changed, except for DT_WSTR and DT_NTEXT columns.
0xC020837F	-1071611009	DTS_E_ADOSRCDATATYPEM ISMATCH	The data type of "%1" does not match the data type "%2" of the source column "%3".
0xC0208380	-1071611008	DTS_E_ADOSRCCOLUMNN OTINSCHMAROWSET	The %1 does not have a matching source column in the schema.
0xC0208381	-1071611007	DTS_E_TERMLOOKUP_INVA LIDREFERENCETERMTABLEO RCOLUMN	The reference table/view or column used for the reference terms is invalid.
0xC0208382	-1071611006	DTS_E_TERMLOOKUP_REFER ENCETERMTABLEANDCOLU MNNOTSET	The reference table/view or column used for the reference terms has not been set.
0xC0208383	-1071611005	DTS_E_COLUMNMAPPEDTO ALREADYMAPPEDEXTERNAL METADATACOLUMN	The %1 is mapped to the external metadata column with ID %2!d!, which is already mapped to another column.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0208384	-1071611004	DTS_E_TXFUZZYLOOKUP_T OOMANYPREFIXES	The SQL object name '%1' specified for property '%2' contains more than the maximum number of prefixes. The maximum is 2.
0xC0208385	-1071611003	DTS_E_MGDSRCSTATIC_OVE RFLOW	The value was too large to fit in the column.
0xC0208386	-1071611002	DTS_E_DATAREADERDESTRE ADERISCLOSED	The SSIS IDataReader is closed.
0xC0208387	-1071611001	DTS_E_DATAREADERDESTRE ADERISATEND	The SSIS IDataReader is past the end of the result set.
0xC0208388	-1071611000	DTS_E_DATAREADERDESTIN VALIDCOLUMNORDINAL	The ordinal position of the column is not valid.
0xC0208389	-1071610999	DTS_E_DATAREADERDESTCA NNOTCONVERT	Cannot convert the %1 from data type "%2" to data type "%3".
0xC020838A	-1071610998	DTS_E_DATAREADERDESTIN VALIDCODEPAGE	The %1 has unsupported code page %2!d!.
0xC020838B	-1071610997	DTS_E_XMLSRCEXTERNALM ETADATACOLUMNNOTINSC HEMA	The %1 has no mapping to the XML schema.
0xC020838D	-1071610995	DTS_E_TXTERMLOOKUP_MI SMATCHED_COLUMN_MET ADATA	Columns with lineage IDs %1!d! and %2!d! have mismatched metadata. The input column that is mapped to an output column does not have the same metadata (datatype, precision, scale, length, or codepage).
0xC020838E	-1071610994	DTS_E_DATAREADERDESTRE ADERTIMEOUT	The SSIS IDataReader is closed. The read timeout has expired.
0xC020838F	-1071610993	DTS_E_ADOSRCINVALIDSQL COMMAND	An error occurred executing the provided SQL command: "%1". %2

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0208390	-1071610992	DTS_E_JOINTYPEDOESNTMATCHETI	The JoinType property specified for input column '%1' differs from the JoinType specified for the corresponding reference table column when the Match Index was initially created. Either rebuild the Match Index with the given JoinType or change the JoinType to match the type used when the Match Index was created.
0xC0208392	-1071610990	DTS_E_SQLCEDESTDATATYPENOTSUPPORTED	The data type "%1" found on column "%2" is not supported for the %3.
0xC0208393	-1071610989	DTS_E_DATAREADERDESTDATATYPENOTSUPPORTED	The data type "%1" found on %2 is not supported for the %3.
0xC0208394	-1071610988	DTS_E_RECORDSETDESTDATATYPENOTSUPPORTED	The data type of the %1 is not supported for the %2.
0xC0208446	-1071610810	DTS_E_TXSCRIPTMIGRATIONCOULDNOTADDREFERENCE	Failed to add project reference "%1" while migrating %2. Migration might need to be completed manually.
0xC0208447	-1071610809	DTS_E_TXSCRIPTMIGRATIONMULTIPLEENTRYPOINTSFOUND	Multiple entry points with the name "%1" were found during the migration of %2. Migration might need to be completed manually.
0xC0208448	-1071610808	DTS_E_TXSCRIPTMIGRATIONNOENTRYPOINTFOUND	No entry point was found during the migration of %1. Migration might need to be completed manually.
0xC020844B	-1071610805	DTS_E_ADODESTINSERTIONFAILURE	An exception has occurred during data insertion, the message returned from the provider is: %1
0xC020844C	-1071610804	DTS_E_ADODESTCONNECTIONTYPENOTSUPPORTED	Failed to retrieve the provider invariant name from %1, it is currently not supported by ADO NET Destination component

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC020844D	-1071610803	DTS_E_ADODESTARGUMENTEXCEPTION	An argument exception has occurred while data provider tried to insert data to destination. The returned message is : %1
0xC020844E	-1071610802	DTS_E_ADODESTWRONGBATCHSIZE	The BatchSize property must be a non-negative integer
0xC020844F	-1071610801	DTS_E_ADODESTERRORUPDATEROW	An error has occurred while sending this row to destination data source.
0xC0208450	-1071610800	DTS_E_ADODESTEXECUTEREDEREXCEPTION	Executing tSQL command throws an exception, the message is : %1
0xC0208451	-1071610799	DTS_E_ADODESTDATATYPENOTSUPPORTED	The data type "%1" found on column "%2" is not supported for the %3.
0xC0208452	-1071610798	DTS_E_ADODESTFAILEDTOACQUIRECONNECTION	ADO NET Destination has failed to acquire the connection %1. The connection may have been corrupted.
0xC0208453	-1071610797	DTS_E_ADODESTNOTMANAGEDCONNECTION	The specified connection %1 is not managed, please use managed connection for ADO NET destination.
0xC0208454	-1071610796	DTS_E_ADODESTNOERROROUTPUT	The destination component does not have an error output. It may have been corrupted.
0xC0208455	-1071610795	DTS_E_ADODESTNOLINEAGEID	The lineageID %1 associated with external column %2 does not exist at run time.
0xC0208456	-1071610794	DTS_E_ADODESTEXTERNALCOLUMNNOTEXIST	The %1 does not exist in the database. It may have been removed or renamed.
0xC0208457	-1071610793	DTS_E_ADODESTGETSCHEMATABLEFAILED	Failed to get properties of external columns. The table name you entered may not exist, or you do not have SELECT permission on the table object and an alternative attempt to get column properties through connection has failed. Detailed error messages are: %1

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0208458	-1071610792	DTS_E_ADODESTCOLUMNERRORDISPNOTSUPPORTED	Input column error disposition is not supported by ADO NET Destination component.
0xC0208459	-1071610791	DTS_E_ADODESTCOLUMNTRUNDISPNOTSUPPORTED	Input column truncation disposition is not supported by ADO NET Destination component.
0xC020845A	-1071610790	DTS_E_ADODESTINPUTTRUNDISPNOTSUPPORTED	Input truncation row disposition is not supported by ADO NET Destination component.
0xC020845B	-1071610789	DTS_E_ADODESTTABLENAMEERROR	The Table or View name is not expected. \n\t If you are quoting the table name, please use the prefix %1 and the suffix %2 of your selected data provider for quotation. \n\t If you are using multipart name, please use at most three parts for the table name.
0xC0209001	-1071607807	DTS_E_FAILEDTOFINDCOLUMNINBUFFER	Failed to find column "%1" with lineage ID %2!d! in the buffer. The buffer manager returned error code 0x%3!8.8X!.
0xC0209002	-1071607806	DTS_E_FAILEDTOGETCOLUMNINFOFROMBUFFER	Failed to get information for column "%1" (%2!d!) from the buffer. The error code returned was 0x%3!8.8X!.
0xC0209011	-1071607791	DTS_E_TXAGG_ARITHMETIC_OVERFLOW	Arithmetic overflow encountered while aggregating "%1".
0xC0209012	-1071607790	DTS_E_FAILEDTOGETCOLUMNINFO	Failed to get information for row %1!d!, column %2!d! from the buffer. The error code returned was 0x%3!8.8X!.
0xC0209013	-1071607789	DTS_E_FAILEDTOSETCOLUMNINFO	Failed to set information for row %1!d!, column %2!d! into the buffer. The error code returned was 0x%3!8.8X!.
0xC0209015	-1071607787	DTS_E_REQUIREDBUFFERISNOTAVAILABLE	A required buffer is not available.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0209016	-1071607786	DTS_E_FAILEDTOGETBUFFER BOUNDARYINFO	The attempt to get buffer boundary information failed with error code 0x%1!8.8X!.
0xC0209017	-1071607785	DTS_E_FAILEDTOSETBUFFER ENDOFROWSET	Setting the end of rowset for the buffer failed with error code 0x%1!8.8X!.
0xC0209018	-1071607784	DTS_E_FAILEDTOGETDATAF ORERROROUTPUTBUFFER	Failed to get data for the error output buffer.
0xC0209019	-1071607783	DTS_E_FAILEDTOREMOVER OWFROMBUFFER	Removing a row from the buffer failed with error code 0x%1!8.8X!.
0xC020901B	-1071607781	DTS_E_FAILEDTOSETBUFFER ERRORINFO	The attempt to set buffer error information failed with error code 0x%1!8.8X!.
0xC020901C	-1071607780	DTS_E_COLUMNSTATUSERR OR	There was an error with %1 on %2. The column status returned was: "%3".
0xC020901D	-1071607779	DTS_E_TXLOOKUP_METADA TAXMLCACHEERR	Unable to cache reference metadata.
0xC020901E	-1071607778	DTS_E_TXLOOKUP_ROWLO OKUPERROR	Row yielded no match during lookup.
0xC020901F	-1071607777	DTS_E_INVALIDERRORDISP OSITION	The %1 has an invalid error or truncation row disposition.
0xC0209022	-1071607774	DTS_E_FAILEDTODIRECTERR ORROW	Directing the row to the error output failed with error code 0x%1!8.8X!.
0xC0209023	-1071607773	DTS_E_FAILEDTOPREPAREC OLUMNSTATUSESFORINSER T	Preparing column statuses for insert failed with error code 0x%1!8.8X!.
0xC0209024	-1071607772	DTS_E_FAILEDTOFINDCOLU MNBYLINEAGEID	An attempt to find %1 with lineage ID %2!d! in the Data Flow Task buffer failed with error code 0x%3!8.8X!.
0xC0209025	-1071607771	DTS_E_FAILEDTOFINDNONS PECIALERRORCOLUMN	Failed to find any non-special error column in %1.



HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0209029	-1071607767	DTS_E_INDUCEDTRANSFORMFAILUREONERROR	SSIS Error Code DTS_E_INDUCEDTRANSFORMFAILUREONERROR. The "%1" failed because error code 0x%2!8.X! occurred, and the error row disposition on "%3" specifies failure on error. An error occurred on the specified object of the specified component. There may be error messages posted before this with more information about the failure.
0xC020902A	-1071607766	DTS_E_INDUCEDTRANSFORMFAILUREONTRUNCATION	The "%1" failed because truncation occurred, and the truncation row disposition on "%2" specifies failure on truncation. A truncation error occurred on the specified object of the specified component.
0xC020902B	-1071607765	DTS_E_TXSPLITEXPRESSIONEVALUATEDTONULL	The expression "%1" on "%2" evaluated to NULL, but the "%3" requires a Boolean results. Modify the error row disposition on the output to treat this result as False (Ignore Failure) or to redirect this row to the error output (Redirect Row). The expression results must be Boolean for a Conditional Split. A NULL expression result is an error.
0xC020902C	-1071607764	DTS_E_TXSPLITSTATIC_EXPRESSIONEVALUATEDTONULL	The expression evaluated to NULL, but a Boolean result is required. Modify the error row disposition on the output to treat this result as False (Ignore Failure) or to redirect this row to the error output (Redirect Row). The expression results must be Boolean for a Conditional Split. A NULL expression result is an error.
0xC020902D	-1071607763	DTS_E_UTF16BIGENDIANFORMATNOTSUPPORTED	The file format of UTF-16 big endian is not supported. Only UTF-16 little endian format is supported.
0xC020902E	-1071607762	DTS_E_UTF8FORMATNOTSUPPORTEDASUNICODE	The file format of UTF-8 is not supported as Unicode.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC020902F	-1071607761	DTS_E_DTPXMLCANTREADIDATTR	Cannot read ID attribute.
0xC020903E	-1071607746	DTS_E_TXLOOKUP_INDEXC OLUMNREUSED	The cache index column %1 is referenced by more than one lookup input column.
0xC020903F	-1071607745	DTS_E_TXLOOKUP_INDEXC OLUMNMISMATCH	Lookup does not reference all cache connection manager index columns. Number of joined columns in lookup: %1!d!. Number of index columns: %2!d!.
0xC0209069	-1071607703	DTS_E_COMMANDDESTINATIONADAPTERSTATIC_CANT CONVERTVALUE	The data value cannot be converted for reasons other than sign mismatch or data overflow.
0xC020906A	-1071607702	DTS_E_COMMANDDESTINATIONADAPTERSTATIC_SCHE MAVIOLATION	The data value violated the schema constraint.
0xC020906B	-1071607701	DTS_E_COMMANDDESTINATIONADAPTERSTATIC_TRUN CATED	The data was truncated.
0xC020906C	-1071607700	DTS_E_COMMANDDESTINATIONADAPTERSTATIC_SIGN MISMATCH	Conversion failed because the data value was signed and the type used by the provider was unsigned.
0xC020906D	-1071607699	DTS_E_COMMANDDESTINATIONADAPTERSTATIC_DATA OVERFLOW	Conversion failed because the data value overflowed the type used by the provider.
0xC020906E	-1071607698	DTS_E_COMMANDDESTINATIONADAPTERSTATIC_UNA VAILABLE	No status is available.
0xC020906F	-1071607697	DTS_E_COMMANDDESTINATIONADAPTERSTATIC_PERM ISSIONDENIED	The user did not have the correct permissions to write to the column.
0xC0209070	-1071607696	DTS_E_COMMANDDESTINATIONADAPTERSTATIC_INTE GRITYVIOLATION	The data value violated the integrity constraints for the column.
0xC0209071	-1071607695	DTS_E_OLEDBSOURCEADAPTERSTATIC_UNAVAILABLE	No status is available.
0xC0209072	-1071607694	DTS_E_OLEDBSOURCEADAPTERSTATIC_CANTCONVERTV ALUE	The data value cannot be converted for reasons other than sign mismatch or data overflow.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0209073	-1071607693	DTS_E_OLEDBSOURCEADAPTERSTATIC_TRUNCATED	The data was truncated.
0xC0209074	-1071607692	DTS_E_OLEDBSOURCEADAPTERSTATIC_SIGNMISMATCH	Conversion failed because the data value was signed and the type used by the provider was unsigned.
0xC0209075	-1071607691	DTS_E_OLEDBSOURCEADAPTERSTATIC_DATAOVERFLOW	Conversion failed because the data value overflowed the type used by the provider.
0xC0209076	-1071607690	DTS_E_OLEDBDESTINATIONADAPTERSTATIC_SCHEMAVIOLATION	The data value violated the schema constraint.
0xC0209077	-1071607689	DTS_E_OLEDBDESTINATIONADAPTERSTATIC_CANTCONVERTVALUE	The data value cannot be converted for reasons other than sign mismatch or data overflow.
0xC0209078	-1071607688	DTS_E_OLEDBDESTINATIONADAPTERSTATIC_TRUNCATED	The data was truncated.
0xC0209079	-1071607687	DTS_E_OLEDBDESTINATIONADAPTERSTATIC_SIGNMISMATCH	Conversion failed because the data value was signed and the type used by the provider was unsigned.
0xC020907A	-1071607686	DTS_E_OLEDBDESTINATIONADAPTERSTATIC_DATAOVERFLOW	Conversion failed because the data value overflowed the type used by the provider.
0xC020907B	-1071607685	DTS_E_OLEDBDESTINATIONADAPTERSTATIC_UNAVAILABLE	No status is available.
0xC020907C	-1071607684	DTS_E_OLEDBDESTINATIONADAPTERSTATIC_PERMISSIONDENIED	The user did not have the correct permissions to write to the column.
0xC020907D	-1071607683	DTS_E_OLEDBDESTINATIONADAPTERSTATIC_INTEGRITYVIOLATION	The data value violates integrity constraints.
0xC020907F	-1071607681	DTS_E_TXDATA CONVERTSTATIC_CANTCONVERTVALUE	The data value cannot be converted for reasons other than sign mismatch or data overflow.
0xC0209080	-1071607680	DTS_E_TXDATA CONVERTSTATIC_TRUNCATED	The data was truncated.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0209081	-1071607679	DTS_E_TXDATAACONVERTST ATIC_SIGNMISMATCH	Conversion failed because the data value was signed and the type used by the provider was unsigned.
0xC0209082	-1071607678	DTS_E_TXDATAACONVERTST ATIC_DATAOVERFLOW	Conversion failed because the data value overflowed the type used by the data conversion transform.
0xC0209083	-1071607677	DTS_E_FLATFILESOURCEAD APTERSTATIC_UNAVAILABL E	No status is available.
0xC0209084	-1071607676	DTS_E_FLATFILESOURCEAD APTERSTATIC_CANTCONVE RTVALUE	The data value cannot be converted for reasons other than sign mismatch or data overflow.
0xC0209085	-1071607675	DTS_E_FLATFILESOURCEAD APTERSTATIC_TRUNCATED	The data was truncated.
0xC0209086	-1071607674	DTS_E_FLATFILESOURCEAD APTERSTATIC_SIGNMISMAT CH	Conversion failed because the data value was signed and the type used by the flat file source adapter was unsigned.
0xC0209087	-1071607673	DTS_E_FLATFILESOURCEAD APTERSTATIC_DATAOVERFL OW	Conversion failed because the data value overflowed the type used by the flat file source adapter.
0xC020908E	-1071607666	DTS_E_TXDATAACONVERTST ATIC_UNAVAILABLE	No status is available.
0xC0209090	-1071607664	DTS_E_FILEOPENERR_FORRE AD	Opening the file "%1" for reading failed with error code 0x%2!8.8X!.
0xC0209091	-1071607663	DTS_E_TXFILEINSERTERSTATI C_FILEOPENERR_FORREAD	Failed to open file for reading.
0xC0209092	-1071607662	DTS_E_FILEOPENERR_FORW RITE	Opening the file "%1" for writing failed with error code 0x%2!8.8X!.
0xC0209093	-1071607661	DTS_E_TXFILEEXTRACTORST ATIC_FILEOPENERR_FORWRI TE	Failed to open file for writing.
0xC0209094	-1071607660	DTS_E_TXFILEINSERTERSTATI C_INSERTERCANTREAD	Failed to read from file.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0209095	-1071607659	DTS_E_TXFILEEXTRACTORST ATIC_EXTRACTORCANTWRIT E	Failed to write to file.
0xC0209099	-1071607655	DTS_E_DTPXMLINVALIDPRO PERTYARRAYTOOMANYVAL UES	Too many array elements were found when parsing a property of type array. The elementCount is less than the number of array elements found.
0xC020909A	-1071607654	DTS_E_DTPXMLINVALIDPRO PERTYARRAYNOTENOUGHV ALUES	Too few array elements were found when parsing a property of type array. The elementCount is more than the number of array elements found.
0xC020909E	-1071607650	DTS_E_FILEOPENERR_FORW RITE_FILENOTFOUND	Opening the file "%1" for writing failed. The file cannot be found.
0xC020909F	-1071607649	DTS_E_TXFILEEXTRACTORST ATIC_FILEOPENERR_FORWRI TE_FILENOTFOUND	Opening the file for writing failed. The file cannot be found.
0xC02090A0	-1071607648	DTS_E_FILEOPENERR_FORW RITE_PATHNOTFOUND	Opening the file "%1" for writing failed. The path cannot be found.
0xC02090A1	-1071607647	DTS_E_TXFILEEXTRACTORST ATIC_FILEOPENERR_FORWRI TE_PATHNOTFOUND	Opening the file for writing failed. The path cannot be found.
0xC02090A2	-1071607646	DTS_E_FILEOPENERR_FORW RITE_TOOMANYOPENFILES	Opening the file "%1" for writing failed. There are too many files open.
0xC02090A3	-1071607645	DTS_E_TXFILEEXTRACTORST ATIC_FILEOPENERR_FORWRI TE_TOOMANYOPENFILES	Opening the file for writing failed. There are too many files open.
0xC02090A4	-1071607644	DTS_E_FILEOPENERR_FORW RITE_ACCESSDENIED	Opening the file "%1" for writing failed. You do not have the correct permissions.
0xC02090A5	-1071607643	DTS_E_TXFILEEXTRACTORST ATIC_FILEOPENERR_FORWRI TE_ACCESSDENIED	Opening the file for writing failed. You do not have the correct permissions.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC02090A6	-1071607642	DTS_E_FILEOPENERR_FORWRITE_FILEEXISTS	Opening the file "%1" for writing failed. The file exists and cannot be overwritten. If the AllowAppend property is FALSE and the ForceTruncate property is set to FALSE, the existence of the file will cause this failure.
0xC02090A7	-1071607641	DTS_E_TXFILEEXTRACTORSTATIC_FILEOPENERR_FORWRITE_FILEEXISTS	Opening a file for writing failed. The file already exists and cannot be overwritten. If both the AllowAppend property and the ForceTruncate property are set to FALSE, the existence of the file will cause this failure.
0xC02090A8	-1071607640	DTS_E_INCORRECTCUSTOMPROPERTYVALUEFOROBJECT	The value for custom property "%1" on %2 is incorrect.
0xC02090A9	-1071607639	DTS_E_COLUMNSHAVEINCOMPATIBLEMETADATA	Columns "%1" and "%2" have incompatible metadata.
0xC02090AD	-1071607635	DTS_E_FILEWRITEERR_DISKFULL	Opening the file "%1" for writing failed because the disk is full. There is not sufficient disk space to save this file.
0xC02090AE	-1071607634	DTS_E_TXFILEEXTRACTORSTATIC_FILEWRITEERR_DISKFULL	Attempting to open the file for writing failed because the disk is full.
0xC02090B9	-1071607623	DTS_E_TXAGG_SORTKEYGENFAILED	Generating a sort key failed with error 0x%1!8.X!. The ComparisonFlags are enabled, and generating a sortkey with LCMapString failed.
0xC02090BA	-1071607622	DTS_E_TXCHARMAPLCMAPFAILED	Transform failed to map string and returned error 0x%1!8.X!. The LCMapString failed.
0xC02090BB	-1071607621	DTS_E_FILEOPENERR_FORREAD_FILENOTFOUND	Opening the file "%1" for reading failed. The file was not found.
0xC02090BC	-1071607620	DTS_E_TXFILEINSERTERSTATIC_FILEOPENERR_FORREAD_FILENOTFOUND	Opening a file for reading failed. The file was not found.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC02090BD	-1071607619	DTS_E_FILEOPENERR_FORREAD_PATHNOTFOUND	Opening the file "%1" for reading failed. The path cannot be found.
0xC02090BE	-1071607618	DTS_E_TXFILEINSERTERSTATIC_FILEOPENERR_FORREAD_PATHNOTFOUND	Opening a file for reading failed. The path was not found.
0xC02090BF	-1071607617	DTS_E_FILEOPENERR_FORREAD_TOOMANYOPENFILES	Opening the file "%1" for reading failed. There are too many files open.
0xC02090C0	-1071607616	DTS_E_TXFILEINSERTERSTATIC_FILEOPENERR_FORREAD_TOOMANYOPENFILES	Opening the file for reading failed. There are too many files open.
0xC02090C1	-1071607615	DTS_E_FILEOPENERR_FORREAD_ACCESSDENIED	Attempting to open the file "%1" for reading failed. Access is denied.
0xC02090C2	-1071607614	DTS_E_TXFILEINSERTERSTATIC_FILEOPENERR_FORREAD_ACCESSDENIED	Opening the file for reading failed. You do not have the correct permissions.
0xC02090C3	-1071607613	DTS_E_INSERTERINVALIDBOM	The byte order mark (BOM) value for the file "%1" is 0x%2!4.X!, but the expected value is 0x%3!4.X!. The ExpectBOM property was set for this file, but the BOM value in the file is missing or not valid.
0xC02090C4	-1071607612	DTS_E_TXFILEINSERTERSTATIC_INSERTERINVALIDBOM	The byte order mark (BOM) value for the file is not valid. The ExpectBOM property was set for this file, but the BOM value in the file is missing or not valid.
0xC02090C5	-1071607611	DTS_E_NOCOMPONENTATTACHED	The %1 is not attached to a component. It is required that a component be attached.
0xC02090C9	-1071607607	DTS_E_TXLOOKUP_INVALID_MAXMEMORYPROP	The value for custom property %1 is incorrect. It should be a number between %2!d! and %3!64d!.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC02090CA	-1071607606	DTS_E_TXAGG_COMPFLAGS_BADAGGREGATIONTYPE	The custom property "%1" cannot be specified for the aggregation type selected for this column. The comparison flags custom property can only be specified for group by and count distinct aggregation types.
0xC02090CB	-1071607605	DTS_E_TXAGG_COMPFLAGS_BADDATATYPE	The comparison flags custom property "%1" can only be specified for columns of with datatype DT_STR or DT_WSTR.
0xC02090CD	-1071607603	DTS_E_TXAGG_AGGREGATION_FAILURE	Aggregation on %1 failed with error code 0x%2!8.8X!.
0xC02090CF	-1071607601	DTS_E_MAPPINGSETUPERROR	There was an error setting up the mapping. %1
0xC02090D0	-1071607600	DTS_E_XMLSRCUNABLETOREADXMLDATA	The %1 was unable to read the XML data.
0xC02090D1	-1071607599	DTS_E_XMLSRCUNABLETOGETXMLDATAVARIABLE	The %1 was unable to get the variable specified by the "%2" property.
0xC02090D2	-1071607598	DTS_E_NODATATABLEMATCHROWID	The %1 contains a RowsetID with a value of %2 that does not reference a data table in the schema.
0xC02090D6	-1071607594	DTS_E_TXAGG_BADKEYSVALUE	The property %1 must either be empty, or a number between %2!u! and %3!u!. The Keys or CountDistinctKeys property has a invalid value. The property should be a number between 0 and ULONG_MAX, inclusive, or not be set.
0xC02090D7	-1071607593	DTS_E_TXAGG_TOOMANYKEYS	The aggregate component encountered too many distinct key combinations. It cannot accommodate more than %1!u! distinct key values. There are more than ULONG_MAX distinct key values in the main workspace.



HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC02090D8	-1071607592	DTS_E_TXAGG_TOOMANYCOUNTDISTINCTVALUES	The aggregate component encountered too many distinct values while calculating the count distinct aggregate. It cannot accommodate more than %1! distinct values. There were more than ULONG_MAX distinct values while calculating the count distinct aggregation.
0xC02090D9	-1071607591	DTS_E_FAILEDTOWRITE TO THE FILENAME COLUMN	The attempt to write to the filename column failed with error code 0x%1!8.8X!.
0xC02090DC	-1071607588	DTS_E_FAILEDTOFINDER ROW COLUMN	An error occurred, but the column that caused the error cannot be determined.
0xC02090E3	-1071607581	DTS_E_TXLOOKUP_FAILEDUPGRADE_BAD_VERSION	Unable to upgrade lookup metadata from version %1!d! to %2!d!. The Lookup transform was unable to upgrade metadata from the existing version number in a call to PerformUpgrade().
0xC02090E5	-1071607579	DTS_E_TERMEXTRACTION_OVERFLOW_LOOKUP_NTEXTSPLIT	Failed to locate the ending boundary of a sentence.
0xC02090E6	-1071607578	DTS_E_TERMEXTRACTION_EXCEED_MAX_WORD_NUM	The Term Extraction transformation is unable to process the input text because a sentence from the input text is too long. The sentence is segmented into several sentences.
0xC02090E7	-1071607577	DTS_E_XMLSRC_FAILED_TO_CREATE_READER	The %1 was unable to read the XML data. %2
0xC02090F0	-1071607568	DTS_E_TXLOOKUP_REINIT_METADATA_FAILED	The call to Lookup transform method, ReinitializeMetadata, failed.
0xC02090F1	-1071607567	DTS_E_TXLOOKUP_NO_JOIN_SPECIFIED	The lookup transform must contain at least one input column joined to a reference column, and none were specified. You must specify at least one join column.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC02090F2	-1071607566	DTS_E_MANAGEDERR_BADFORMATSPECIFICATION	The message string being posted by the managed error infrastructure contains a bad format specification. This is an internal error.
0xC02090F3	-1071607565	DTS_E_MANAGEDERR_UNSUPPORTEDTYPE	While formatting a message string using the managed error infrastructure, there was a variant type that does not have formatting support. This is an internal error.
0xC02090F5	-1071607563	DTS_E_DATAREADERSRCUNABLETOPROCESSDATA	The %1 was unable to process the data. %2
0xC02090F6	-1071607562	DTS_E_XMLSRCEMPTYPROPERTY	The property "%1" on the %2 was empty.
0xC02090F7	-1071607561	DTS_E_XMLSRCINVALIDOUTPUTNAME	Attempting to create an output with the name "%1" for the XML table with the path "%2" failed because the name is invalid.
0xC02090F8	-1071607560	DTS_E_MGDSRC_OVERFLOW	The value was too large to fit in the %1.
0xC02090F9	-1071607559	DTS_E_DATAREADERDESTUNABLETOPROCESSDATA	The %1 was unable to process the data.
0xC02090FA	-1071607558	DTS_E_XMLSRC_INDUCEDTRANSFORMFAILUREONTRUNCATION	The "%1" failed because truncation occurred, and the truncation row disposition on "%2" at "%3" specifies failure on truncation. A truncation error occurred on the specified object of the specified component.
0xC02090FB	-1071607557	DTS_E_XMLSRC_INDUCEDTRANSFORMFAILUREONERROR	The "%1" failed because error code 0x%2!8.8X! occurred, and the error row disposition on "%3" at "%4" specifies failure on error. An error occurred on the specified object of the specified component.
0xC0209291	-1071607151	DTS_E_SQLCEDESTSTATIC_FAILEDTOSETVALUES	The SQLCE destination could not set the column values for the row.
0xC0209292	-1071607150	DTS_E_SQLCEDESTSTATIC_FAILEDTOINSERT	The SQLCE destination could not insert the row.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC0209293	-1071607149	DTS_E_TXFUZZYLOOKUP_O LEDBERR_LOADCOLUMNM ETADATA	Encountered OLEDB error while loading column metadata.
0xC0209294	-1071607148	DTS_E_TXFUZZYLOOKUP_T OOFEWREFERENCECOLUM NS	Lookup reference metadata contains too few columns.
0xC0209295	-1071607147	DTS_E_TXSCD_OLEDBERR_L OADCOLUMNMETADATA	Encountered OLEDB error while loading column metadata.
0xC0209296	-1071607146	DTS_E_TXSCD_TOOFEWREFE RENCECOLUMNS	Lookup reference metadata contains too few columns.
0xC0209297	-1071607145	DTS_E_TXSCD_MALLOCERR_ REFERENCECOLUMNINFO	Unable to allocate memory.
0xC0209298	-1071607144	DTS_E_TXSCD_MALLOCERR_ BUFFCOL	Unable to allocate memory.
0xC0209299	-1071607143	DTS_E_TXSCD_MAINWORKS PACE_CREATEERR	Unable to create workspace buffer.
0xC020929A	-1071607142	DTS_E_DTPXMLDOMCREAT ERROR	Unable to instantiate XML DOM document, please verify that MSXML binaries are properly installed and registered.
0xC020929B	-1071607141	DTS_E_DTPXMLDOMLOADE RROR	Unable to load XML data into a local DOM for processing.
0xC020929C	-1071607140	DTS_E_RSTDESTBADVARIAB LETYPE	The type of the runtime variable "%1" is incorrect. The runtime variable type must be Object.
0xC020929E	-1071607138	DTS_E_XMLDATAREADERM ULTIPLEINLINEXMLSCHEMA SNOTSUPPORTED	The XML Source Adapter was unable to process the XML data. Multiple inline schemas are not supported.
0xC020929F	-1071607137	DTS_E_XMLDATAREADERAN YTYPENOTSUPPORTED	The XML Source Adapter was unable to process the XML data. The content of an element can not be declared as anyType.
0xC02092A0	-1071607136	DTS_E_XMLDATAREADERGR OUPREFNOTSUPPORTED	The XML Source Adapter was unable to process the XML data. The content of an element can not contain a reference (ref) to a group.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC02092A1	-1071607135	DTS_E_XMLDATAREADERMI XEDCONTENTFORCOMPLEX TYPESNOTSUPPORTED	The XML Source Adapter does not support mixed content model on Complex Types.
0xC02092A2	-1071607134	DTS_E_XMLDATAREADERINL INESCHEMAFOUNDINSOUR CEXML	The XML Source Adapter was unable to process the XML data. An inline schema must be the first child node in the source Xml.
0xC02092A3	-1071607133	DTS_E_XMLDATAREADERNO INLINESCHEMAFOUND	The XML Source Adapter was unable to process the XML data. No inline schema was found in the source XML, but the "UseInlineSchema" property was set to true.
0xC02092A4	-1071607132	DTS_E_CONNECTIONMANA GERTRANSACTIONANDRETAI NEDINBULKINSERT	The component cannot use a connection manager that retains its connection in a transaction with fastload or bulk insert.
0xC02092A5	-1071607131	DTS_E_OUTPUTREDIRECTIN TRANSACTIONNOTALLOWE D	The %1 cannot be set to redirect on error using a connection in a transaction.
0xC02092A6	-1071607130	DTS_E_FOUNDORPHANEDE XTERNALMETADATACOLU MN	The %1 does not have a corresponding input or output column.
0xC02092A9	-1071607127	DTS_E_RAWDESTNOINPUTC OLUMNS	There is no selected column to be written to the file.
0xC02092AA	-1071607126	DTS_E_RAWDESTBLOBDATA TYPE	The %1 has an invalid data type. Columns with data types DT_IMAGE, DT_TEXT and DT_NTEXT cannot be written to raw files.
0xC02092AB	-1071607125	DTS_E_RAWDESTWRONGEX TERNALMETADATOUSAGE	The external metadata collection is improperly used by this component. The component should use external metadata when appending or truncating an existing file. Otherwise, the external metadata is not needed.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC02092AC	-1071607124	DTS_E_RAWDESTMAPPEDIN PUTCOLUMN	The %1 is mapped to an external metadata column with the id %2!d!. Input columns should not be mapped to external metadata columns when selected Write Option value is Create Always.
0xC02092AD	-1071607123	DTS_E_RAWFILECANTOPENF ORMETADATA	The file cannot be opened for reading the metadata. If the file does not exist, and the component has already defined external metadata, you can set the "ValidateExternalMetadata" property to "false" and the file will be created at the runtime.
0xC02092AE	-1071607122	DTS_E_FAILEDTOACCESSLO BCOLUMN	Failed to access LOB data from the data flow buffer for data source column "%1" with error code 0x%2!8.8X!.
0xC02092AF	-1071607121	DTS_E_XMLSRCUNABLETOP ROCESSXMLDATA	The %1 was unable to process the XML data. %2
0xC02092B0	-1071607120	DTS_E_XMLSRCSTATIC_UNA BLETOPROCESSXMLDATA	The XML Source Adapter was unable to process the XML data.
0xC02092B1	-1071607119	DTS_E_RAWINVALIDACCESS MODE	The value %1!d! is not recognized as a valid access mode.
0xC02092B2	-1071607118	DTS_E_INCOMPLETEDATAS OURCECOLUMNFOUND	Complete metadata information for the data source column "%1" is not available. Make sure the column is correctly defined in the data source.
0xC02092B3	-1071607117	DTS_E_TXAUDIT_ONLYSTRIN GLENGTHCHANGEALLOWE D	Only lengths of User Name column, Package Name column, Task Name column and Machine Name column can be changed. All other audit column datatype information is read only.
0xC02092B4	-1071607116	DTS_E_ROWSETUNAVAILAB LE	A rowset based on the SQL command was not returned by the OLE DB provider.
0xC02092B5	-1071607115	DTS_E_COMMITFAILED	A commit failed.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC02092B6	-1071607114	DTS_E_USEBINARYFORMAT REQUIRESANSIFILE	The custom property "%1" on %2 can only be used with ANSI files.
0xC02092B7	-1071607113	DTS_E_USEBINARYFORMAT REQUIRESBYTES	The custom property "%1" on %2 can only be used with DT_BYTES.
0xC0209302	-1071607038	DTS_E_OLEDB_NOPROVIDE R_ERROR	SSIS Error Code DTS_E_OLEDB_NOPROVIDE R_ERROR. The requested OLE DB provider %2 is not registered. Error code: 0x%1!8.8X!.
0xC0209303	-1071607037	DTS_E_OLEDB_NOPROVIDE R_64BIT_ERROR	SSIS Error Code DTS_E_OLEDB_NOPROVIDE R_64BIT_ERROR. The requested OLE DB provider %2 is not registered -- perhaps no 64-bit provider is available. Error code: 0x%1!8.8X!.
0xC0209306	-1071607034	DTS_E_MULTICACHECOLM APPINGS	The cache column, "%1", is mapped to more than one column. Remove the duplicate column mappings.
0xC0209307	-1071607033	DTS_E_COLNOTMAPPEDTO CACHECOL	The %1 is not mapped to valid cache column.
0xC0209308	-1071607032	DTS_E_CACHECOLDATATYP EINCOMPAT	Cannot map the input column, "%1", and the cache column, "%2", because the data types do not match.
0xC0209309	-1071607031	DTS_E_INCORRECTINPUTCA CHECOLCOUNT	The number of input columns does not match the number of cache columns.
0xC020930A	-1071607030	DTS_E_INVALIDCACHEFILEN AME	The cache file name is either not provided or is not valid. Provide a valid cache file name.
0xC020930B	-1071607029	DTS_E_CACHECOLINDEXPO SMISMATCH	The index position of column, "%1", is different from index position of Cache connection manager column, "%2".
0xC020930C	-1071607028	DTS_E_FAILEDTOLOADCAC HE	Failed to load the cache from file, "%1".

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC020930D	-1071607027	DTS_E_TXLOOKUP_REFCOL UMNISNOTINDEX	The lookup input column %1 references non-index cache column %2.
0xC020930E	-1071607026	DTS_E_FAILEDTOGETCONNE CTIONSTRING	Failed to get the connection string.
0xC020930F	-1071607025	DTS_E_CACHECOLDATATYP EPROPINCOMPAT	Cannot map the input column, "%1", and the cache column, "%2", because one or more data type properties do not match.
0xC0209311	-1071607023	DTS_E_CACHECOLUMNNOTF OUND	Cache column "%1" was not found in the cache.
0xC0209312	-1071607022	DTS_E_CACHECOLUMNMA PPINGFAILED	Failed to map %1 to a cache column. The hresult is 0x%2!8.8X!.
0xC0209313	-1071607021	DTS_E_CACHELOADED FRO MFILE	The %1 cannot write to the cache because the cache has been loaded from a file by %2.
0xC0209314	-1071607020	DTS_E_CACHERELOADED DI FFERENTFILES	The %1 cannot load the cache from file "%2" because the cache has already been loaded from file "%3".
0xC0209316	-1071607018	DTS_E_OUTPUTNOTUSED	The output with ID %1!d! of Aggregate component is not used by any component. Please either remove it or associate it with an input of some component.
0xC0209317	-1071607017	DTS_E_CACHEFILEWRITEFAI LED	The %1 failed to write the cache to file "%2". The hresult is 0x%3!8.8X!.
0xC0209318	-1071607016	DTS_E_XMLDATATYPECHAN GED	The XML schema data type information for "%1" on element "%2" has changed. Please re-initialize the metadata for this component and review column mappings.
0xC0209319	-1071607015	DTS_E_TXLOOKUP_UNUSED INPUTCOLUMN	%1 not used in join or copy. Please remove the unused column from the input column list.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC020931A	-1071607014	DTS_E_SORTSTACKOVERFLOW	The sort failed due to a stack overflow while sorting an incoming buffer. Please reduce the DefaultBufferMaxRows property on the Data Flow Task.
0xC020F42A	-1071582166	DTS_E_OLEDB_OLDPROVIDER_ERROR	Consider changing the PROVIDER in the connection string to %1 or visit <a href="http://www.microsoft.com/downloads">http://www.microsoft.com/downloads</a> to find and install support for %2.
		DTS_E_INITTASKOBJECTFAILED	Failed to initialize the task object for task "%1!", type "%2!" due to error 0x%3!8.X! "%4!".
		DTS_E_GETCATMANAGERFAILED	Failed to create COM Component Categories Manager due to error 0x%1!8.X! "%2!".
		DTS_E_COMPONENTINITFAILED	Component %1! failed to initialize due to error 0x%2!8.X! "%3!".

## Warning Messages

The symbolic names of Integration Services warning messages begin with **DTS\_W\_**.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0x80000036	-2147483594	DTS_W_COUNTDOWN	There are %1! days left in the evaluation. When it expires, packages will not be able to be executed.
0x80010015	-2147418091	DTS_W_GENERICWARNING	Warning(s) raised. There should be more specific warnings preceding this one that explain the specifics of the warning(s).
0x80012010	-2147409904	DTS_W_FAILEDXMLDOCUMENTATION	Cannot create an XML document object instance. Verify that MSXML is installed and registered correctly.
0x80012011	-2147409903	DTS_W_FAILEDCONFIGLOAD	Cannot load the XML configuration file. The XML configuration file may be malformed or not valid.



HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0x80012012	-2147409902	DTS_W_CONFIGFILENAMEINVALID	The configuration file name "%1" is not valid. Check the configuration file name.
0x80012013	-2147409901	DTS_W_CONFIGFILEINVALID	The configuration file loaded, but is not valid. The file is not formatted correctly, may be missing an element, or may be damaged.
0x80012014	-2147409900	DTS_W_CONFIGFILENOTFOUND	The configuration file "%1" cannot be found. Check the directory and file name.
0x80012015	-2147409899	DTS_W_CONFIGKEYNOTFOUND	Configuration registry key "%1" was not found. A configuration entry specifies a registry key that is not available. Check the registry to ensure that the key is there.
0x80012016	-2147409898	DTS_W_CONFIGTYPEINVALID	The configuration type in one of the configuration entries was not valid. Valid types are listed in the DTSType enumeration.
0x80012017	-2147409897	DTS_W_CANNOTFINDOBJECT	The package path referenced an object that cannot be found: "%1". This occurs when an attempt is made to resolve a package path to an object that cannot be found.
0x80012018	-2147409896	DTS_W_CONFIGFORMATINVALID_PACKAGEDELIMITER	The configuration entry, "%1", has an incorrect format because it does not begin with the package delimiter. Prepend "\package" to the package path.
0x80012019	-2147409895	DTS_W_CONFIGFORMATINVALID	The configuration entry "%1" had an incorrect format. This can occur because of a missing delimiter or formatting errors, like an invalid array delimiter.
0x8001201A	-2147409894	DTS_W_NOPARENTVARIABLES	Configuration from a parent variable "%1" did not occur because there was no parent variable collection.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0x8001201B	-2147409893	DTS_W_CONFIGFILEFAILEDIMPORT	Failure importing configuration file: "%1".
0x8001201C	-2147409892	DTS_W_PARENTVARIABLENOTFOUND	Configuration from a parent variable "%1" did not occur because there was no parent variable. Error code: 0x%2!8.8X!.
0x8001201D	-2147409891	DTS_W_CONFIGFILEEMPTY	The configuration file was empty and contained no configuration entries.
0x80012023	-2147409885	DTS_W_INVALIDCONFIGURATIONTYPE	The configuration type for configuration "%1" is not valid. This may occur when an attempt is made to set the type property of a configuration object to an invalid configuration type.
0x80012025	-2147409883	DTS_W_REGISTRYCONFIGURATIONTYPENOTFOUND	The configuration type for the registry configuration was not found in key "%1". Add a value called ConfigType to the registry key and give it a string value of "Variable", "Property", "ConnectionManager", "LoggingProvider", or "ForEachEnumerator".
0x80012026	-2147409882	DTS_W_REGISTRYCONFIGURATIONVALUENOTFOUND	The configuration value for the registry configuration was not found in key "%1". Add a value called Value to the registry key of type DWORD or String.
0x80012028	-2147409880	DTS_W_PROCESSCONFIGURATIONFAILEDSET	Process configuration failed to set the destination at the package path of "%1". This occurs when attempting to set the destination property or variable fails. Check the destination property or variable.
0x80012032	-2147409870	DTS_W_CONFIGUREDVALUESECTIONEMPTY	Failed to retrieve value from the .ini file. The ConfiguredValue section is either empty, or does not exist: "%1".

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0x80012033	-2147409869	DTS_W_CONFIGUREDTYPES ECTIONEMPTY	Failed to retrieve value from the .ini file. The ConfiguredType section is either empty, or does not exist: "%1".
0x80012034	-2147409868	DTS_W_PACKAGEPATHSECT IONEMPTY	Failed to retrieve value from the .ini file. The PackagePath section is either empty, or does not exist: "%1".
0x80012035	-2147409867	DTS_W_CONFIGUREDVALU ETYPE	Failed to retrieve value from the .ini file. The ConfiguredValueType section is either empty, or does not exist: "%1".
0x80012051	-2147409839	DTS_W_SQLSERVERFAILED IMPORT	Configuration from SQL Server was not successfully imported: "%1".
0x80012052	-2147409838	DTS_W_INICONFIGURATIO NPROBLEM	The .ini configuration file is not valid due to empty or missing fields.
0x80012054	-2147409836	DTS_W_NORECORDSFOUN DINTABLE	Table "%1" does not have any records for configuration. This occurs when configuring from a SQL Server table that has no records for the configuration.
0x80012055	-2147409835	DTS_W_DUPLICATECUSTO MEVENT	Error using same name for different custom events. The custom event "%1" was defined differently by different children of this container. There may be an error when executing the event handler.
0x80012057	-2147409833	DTS_W_CONFIGREADONLY VARIABLE	The configuration attempted to change a read-only variable. The variable is at the package path "%1".
0x80012058	-2147409832	DTS_W_CONFIGPROCESSC ONFIGURATIONFAILED	Calling ProcessConfiguration on the package failed. The configuration attempted to change the property at the package path "%1".

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0x80012059	-2147409831	DTS_W_ONEORMORECONFIGLOADFAILED	Failed to load at least one of the configuration entries for the package. Check configuration entries for "%1" and previous warnings to see descriptions of which configuration failed.
0x8001205A	-2147409830	DTS_W_CONFIGNODEINVALID	The configuration entry "%1" in the configuration file was not valid, or failed to configure the variable. The name indicates which entry failed. In some cases, the name will not be available.
0x80014058	-2147401640	DTS_W_FAILURENOTRESTARTABLE	This task or container has failed, but because FailPackageOnFailure property is FALSE, the package will continue. This warning is posted when the SaveCheckpoints property of the package is set to TRUE and the task or container fails.
0x80017101	-2147389183	DTS_W_EMPTYPATH	The path is empty.
0x80019002	-2147381246	DTS_W_MAXIMUMERRORCOUNTREACHED	SSIS Warning Code DTS_W_MAXIMUMERRORCOUNTREACHED. The Execution method succeeded, but the number of errors raised (%1!d!) reached the maximum allowed (%2!d!); resulting in failure. This occurs when the number of errors reaches the number specified in MaximumErrorCount. Change the MaximumErrorCount or fix the errors.
0x80019003	-2147381245	DTS_W_CONFIGENVVARIABLENOTFOUND	The configuration environment variable was not found. The environment variable was: "%1". This occurs when a package specifies an environment variable for a configuration setting but it cannot be found. Check the configurations collection in the package and verify that the specified environment variable is available and valid.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0x80019316	-2147380458	DTS_W_CONNECTIONPROVIDERCHANGE	The provider name for the connection manager "%1" has been changed from "%2" to "%3".
0x80019317	-2147380457	DTS_W_READEXTMAPFAILED	An exception occurred while reading the upgrade mapping files. The exception is "%1".
0x80019318	-2147380456	DTS_W_DUPLICATEMAPPINGKEY	There is a duplicate mapping in file, "%1". The tag is "%2", and the key is "%3".
0x80019319	-2147380455	DTS_W_IMPLICITUPGRADE MAPPING	The extension, "%1", was implicitly upgraded to "%2". Add a mapping for this extension to the UpgradeMappings directory.
0x8001931A	-2147380454	DTS_W_INVALIDEXTENSION MAPPING	A mapping in the file, "%1", is not valid. Values cannot be null or empty. The tag is "%2", the key is "%3", and the value is "%4".
0x8001931C	-2147380452	DTS_W_ADOCONNECTION DATATYPECOMPATCHANGE	The DataTypeCompatibility property of ADO type connection manager "%1" was set to 80 for backward compatibility reasons.
0x8001C004	-2147368956	DTS_W_FILEENUMEMPTY	The For Each File enumerator is empty. The For Each File enumerator did not find any files that matched the file pattern, or the specified directory was empty.
0x8001F02F	-2147356625	DTS_W_COULDNOTRESOLVE PACKAGEPATH	Cannot resolve a package path to an object in the package "%1". Verify that the package path is valid.
0x8001F203	-2147356157	DTS_W_ITERATIONEXPRESSIONISNOTASSIGNMENT	The iteration expression is not an assignment expression: "%1". This error usually occurs when the expression in the assignment expression on the ForLoop is not an assignment expression.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0x8001F204	-2147356156	DTS_W_INITIALIZATIONEXPRESSIONISNOTASSIGNMENT	The initialization expression is not an assignment expression: "%1". This error usually occurs when the expression in the iterate expressions on the ForLoop is not an assignment expression.
0x8001F205	-2147356155	DTS_W_LOGPROVIDERNOTDEFINED	The executable "%1" was pasted successfully. However a log provider that was associated with the executable was not found in the collection "LogProviders". The executable was pasted without log provider information.
0x8001F300	-2147355904	DTS_W_PACKAGEUPGRADE	Succeeded in upgrading the package.
0x8001F42B	-2147355605	DTS_W_LEGACYPROGID	The "%1" ProgID has been deprecated. The new ProgID for this component "%2" should be used instead.
0x80020918	-2147350248	DTS_W_FTPTASK_OPERATIONFAILURE	Operation "%1" failed.
0x800283A5	-2147318875	DTS_W_MSMQTASK_USE_WEAK_ENCRYPTION	The encryption algorithm "%1" uses weak encryption.
0x80029164	-2147315356	DTS_W_FSTASK_OPERATIONFAILURE	Task failed to execute operation "%1".
0x80029185	-2147315323	DTS_W_EXECPROCTASK_FILENOTINPATH	File/Process "%1" is not in path.
0x800291C6	-2147315258	DTS_W_SENDMAILTASK_SUBJECT_MISSING	The subject is empty.
0x800291C7	-2147315257	DTS_W_SENDMAILTASK_ERROR_IN_TO_LINE	The address in the "To" line is malformed. It is either missing the "@" symbol or is not valid.
0x800291C8	-2147315256	DTS_W_SENDMAILTASK_ADDRESS_MISSING_IN_FROM	The address in the "From" line is malformed. It is either missing the "@" symbol or is not valid.
0x8002927A	-2147315078	DTS_W_XMLTASK_DIFFFAILURE	The two XML documents are different.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0x8002928C	-2147315060	DTS_W_XMLTASK_DTDVALIDATIONWARNING	DTD Validation will use the DTD file defined in the DOCTYPE line in the XML document. It will not use what is assigned to the property "%1".
0x8002928D	-2147315059	DTS_W_XMLTASK_VALIDATIONFAILURE	Task failed to validate "%1".
0x80029291	-2147315055	DTS_W_TRANSFERDBTASK_ACTIONSETTOCOPY	The transfer action value was invalid. It is being set to copy.
0x80029292	-2147315054	DTS_W_TRANSFERDBTASK_METHODSETTOONLINE	The transfer method value was invalid. It is being set to an online transfer.
0x8002F304	-2147290364	DTS_W_PROBLEMOCCURREDWITHFOLLOWINGMESSAGE	A problem occurred with the following messages: "%1".
0x8002F322	-2147290334	DTS_W_ERRMSGTASK_ERRORMESSAGEALREADYEXISTS	The error message "%1" already exists at destination server.
0x8002F331	-2147290319	DTS_W_JOBSTASK_JOBEXISTSATDEST	The job "%1" already exists at destination server.
0x8002F332	-2147290318	DTS_W_JOBSTASK_SKIPPINGJOBEXISTSATDEST	Skipping the transfer of job "%1" since it already exists at destination.
0x8002F333	-2147290317	DTS_W_JOBSTASK_OVERWRITINGJOB	Overwriting the job "%1" at destination server.
0x8002F339	-2147290311	DTS_W_LOGINTASK_ENUMVALUEINCORRECT	Persisted enumeration value of property "FailIfExists" was changed and rendered invalid. Resetting to default.
0x8002F343	-2147290301	DTS_W_LOGINTASK_OVERWRITINGLOGINATDEST	Overwriting Login "%1" at destination.
0x8002F356	-2147290282	DTS_W_TRANSOBJECTSTASK_SPALREADYATDEST	Stored procedure "%1" already exists at destination.
0x8002F360	-2147290272	DTS_W_TRANSOBJECTSTASK_RULEALREADYATDEST	Rule "%1" already exists at destination.
0x8002F364	-2147290268	DTS_W_TRANSOBJECTSTASK_TABLEALREADYATDEST	Table "%1" already exists at destination.
0x8002F368	-2147290264	DTS_W_TRANSOBJECTSTASK_VIEWALREADYATDEST	View "%1" already exists at destination.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0x8002F372	-2147290254	DTS_W_TRANSOBJECTSTASK_UDFALREADYATDEST	User Defined Function "%1" already exists at destination.
0x8002F376	-2147290250	DTS_W_TRANSOBJECTSTASK_DEFAULTALREADYATDEST	Default "%1" already exists at destination.
0x8002F380	-2147290240	DTS_W_TRANSOBJECTSTASK_UDDTALREADYATDEST	User Defined Data Type "%1" already exists at destination.
0x8002F384	-2147290236	DTS_W_TRANSOBJECTSTASK_PFALREADYATDEST	Partition Function "%1" already exists at destination.
0x8002F388	-2147290232	DTS_W_TRANSOBJECTSTASK_PSQLALREADYATDEST	Partition Scheme "%1" already exists at destination.
0x8002F391	-2147290223	DTS_W_TRANSOBJECTSTASK_SCHEMAALREADYATDEST	Schema "%1" already exists at destination.
0x8002F396	-2147290218	DTS_W_TRANSOBJECTSTASK_SQLASSEMBLYALREADYATDEST	SqlAssembly "%1" already exists at destination.
0x8002F400	-2147290112	DTS_W_TRANSOBJECTSTASKAggregateALREADYATDEST	User Defined Aggregate "%1" already exists at destination.
0x8002F404	-2147290108	DTS_W_TRANSOBJECTSTASK_TYPEALREADYATDEST	User Defined Type "%1" already exists at destination.
0x8002F408	-2147290104	DTS_W_TRANSOBJECTSTASK_XMLSCHEMACOLLECTIONALREADYATDEST	XmlSchemaCollection "%1" already exists at destination.
0x8002F412	-2147290094	DTS_W_TRANSOBJECTSTASK_NOELEMENTSPECIFIEDTOTRANSFER	There are no elements specified to transfer.
0x8002F415	-2147290091	DTS_W_TRANSOBJECTSTASK_LOGINALREADYATDEST	Login "%1" already exists at destination.
0x8002F41A	-2147290086	DTS_W_TRANSOBJECTSTASK_USERALREADYATDEST	User "%1" already exists at destination.
0x80047007	-2147192825	DTS_W_NOLINEAGEVALIDATION	The lineage IDs of the input columns cannot be validated because the execution trees contain cycles.
0x80047034	-2147192780	DTS_W_EMPTYDATAFLOW	The DataFlow task has no components. Add components or remove the task.



HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0x80047069	-2147192727	DTS_W_SORTEDOUTPUTHASORTKEYPOSITIONS	The IsSorted property of %1 is set to TRUE, but all of its output columns' SortKeyPositions are set to zero.
0x8004706F	-2147192721	DTS_W_SOURCECEREMOVED	Source "%1" (%2!d!) will not be read because none of its data ever becomes visible outside the Data Flow Task.
0x80047076	-2147192714	DTS_W_UNUSEDOUTPUTDATA	The output column "%1" (%2!d!) on output "%3" (%4!d!) and component "%5" (%6!d!) is not subsequently used in the Data Flow task. Removing this unused output column can increase Data Flow task performance.
0x800470AE	-2147192658	DTS_W_COMPONENTREMOVED	Component "%1" (%2!d!) has been removed from the Data Flow task because its output is not used and its inputs either have no side effects or are not connected to outputs of other components. If the component is required, then the HasSideEffects property on at least one of its inputs should be set to true, or its output should be connected to something.
0x800470B0	-2147192656	DTS_W_NOWORKTODO	Rows were given to a thread, but that thread has no work to do. The layout has a disconnected output. Running the pipeline with the RunInOptimizedMode property set to TRUE will be faster, and prevents this warning.
0x800470C8	-2147192632	DTS_W_EXTERNALMETADATACOLUMNSOUTOFSYNC	The external columns for %1 are out of synchronization with the data source columns. %2
0x800470C9	-2147192631	DTS_W_EXTERNALMETADATACOLUMNCOLLECTIONNEEDSADDITION	The column "%1" needs to be added to the external columns.
0x800470CA	-2147192630	DTS_W_EXTERNALMETADATACOLUMNCOLLECTIONNEEDSUPDATE	The external column "%1" needs to be updated.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0x800470CB	-2147192629	DTS_W_EXTERNALMETADAT ACOLUMNCOLLECTIONNEE DSREMOVAL	The %1 needs to be removed from the external columns.
0x800470D8	-2147192616	DTS_W_EXPREVALPOTENTIAL STRINGTRUNCATION	The result string for expression "%1" may be truncated if it exceeds the maximum length of %2!d! characters. The expression could have a result value that exceeds the maximum size of a DT_WSTR.
0x800470E9	-2147192599	DTS_W_COMPONENTLEAKP ROCESSINPUT	A call to the ProcessInput method for input %1!d! on %2 unexpectedly kept a reference to the buffer it was passed. The refcount on that buffer was %3!d! before the call, and %4!d! after the call returned.
0x800470EB	-2147192597	DTS_W_EXPREVALUNREFERE NCEDINPUTCOLUMN	The "%1" on "%2" has usage type READONLY, but is not referenced by an expression. Remove the column from the list of available input columns, or reference it in an expression.
0x8004801E	-2147188706	DTS_W_COULDNOTFINDCU RRENTVERSION	Cannot find the "%1" value for component %2. The CurrentVersion value for the component cannot be located. This error occurs if the component has not set its registry information to contain a CurrentVersion value in the DTSInfo section. This message occurs during component development, or when the component is used in a package, if the component is not registered properly.
0x80049300	-2147183872	DTS_W_BUFFERGETTEMPFIL ENAME	The buffer manager could not get a temporary file name.
0x80049301	-2147183871	DTS_W_UNUSABLETEMPOR ARYPATH	The buffer manager could not create a temporary file on the path "%1". The path will not be considered for temporary storage again.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0x80049304	-2147183868	DTS_W_DF_PERFCOUNTERS_DISABLED	Warning: Could not open global shared memory to communicate with performance DLL; data flow performance counters are not available. To resolve, run this package as an administrator, or on the system's console.
0x8020200F	-2145378289	DTS_W_PARTIALROWFOUNDATENDOFFILE	There is a partial row at the end of the file.
0x8020202B	-2145378261	DTS_W_ENDOFFILEREACTIONHILEREADINGHEADERROWS	The end of the data file was reached while reading header rows. Make sure the header row delimiter and the number of header rows to skip are correct.
0x80202066	-2145378202	DTS_W_CANTRETRIEVECODEPAGEFROMOLEDBPROVIDER	Cannot retrieve the column code page info from the OLE DB provider. If the component supports the "%1" property, the code page from that property will be used. Change the value of the property if the current string code page values are incorrect. If the component does not support the property, the code page from the component's locale ID will be used.
0x802020F7	-2145378057	DTS_W_TXSORTSORTISTHESAME	The data is already sorted as specified so the transform can be removed.
0x8020400D	-2145370099	DTS_W_NOPIPELINEDATATYPEMAPPINGAVAILABLE	The %1 references an external data type that cannot be mapped to a Data Flow task data type. The Data Flow task data type DT_WSTR will be used instead.
0x802070CC	-2145357620	DTS_W_STATICTRUNCATIONINEXPRESSION	The expression "%1" will always result in a truncation of data. The expression contains a static truncation (the truncation of a fixed value).
0x8020820C	-2145353204	DTS_W_UNMAPPEDINPUTCOLUMN	The input column "%1" with ID %2!d! at index %3!d! is unmapped. The lineage ID for the column is zero.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0x80208305	-2145352955	DTS_W_TXFUZZYLOOKUP_D ELIMITERS_DONT_MATCH	The specified delimiters do not match the delimiters used to build the pre-existing match index "%1". This error occurs when the delimiters used to tokenize fields do not match. This can have unknown effects on the matching behavior or results.
0x80208308	-2145352952	DTS_W_TXFUZZYLOOKUP_ MAXRESULTS_IS_ZERO	The MaxOutputMatchesPerInput property on the Fuzzy Lookup transformation is zero. No results will be produced.
0x80208310	-2145352944	DTS_W_TXFUZZYLOOKUP_N O_FUZZY_JOIN_COLUMNS	There were no valid input columns with JoinType column property set to Fuzzy. Performance on Exact joins may be improved by using the Lookup transform instead of FuzzyLookup.
0x8020831C	-2145352932	DTS_W_TXFUZZYLOOKUP_T IMESTAMPCAVEAT	The reference column "%1" may be a SQL timestamp column. When the fuzzy match index is built, and a copy of the reference table is made, all reference table timestamps will reflect the current state of the table at the time of the copy. Unexpected behavior may occur if the CopyReferenceTable is set to false.
0x80208321	-2145352927	DTS_W_MATCHINDEXALREA DYEXISTS	A table with the name '%1' given for MatchIndexName already exists and DropExistingMatchIndex is set to FALSE. Transform execution will fail unless this table is dropped, a different name is specified, or DropExistingMatchIndex is set to TRUE.
0x8020832B	-2145352917	DTS_W_TXFUZZYLOOKUP_J OINLENGTHMISMATCH	The length of input column '%1' is not equal to the length of the reference column '%2' that it is being matched against.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0x8020832D	-2145352915	DTS_W_TXFUZZYLOOKUP_C ODEPAGE_MISMATCH	The code pages of the DT_STR source column "%1" and the DT_STR dest column "%2" do not match. This may cause unexpected results.
0x8020832E	-2145352914	DTS_W_FUZZYLOOKUP_TO OMANYEXACTMATCHCOLU MNS	There are more than 16 exact match joins, so performance may not be optimal. Reduce the number of exact match joins to improve performance. SQL Server has a limit of 16 columns per index, the inverted index will be used for all lookups.
0x80208350	-2145352880	DTS_W_FUZZYLOOKUP_ME MLIMITANDEXHAUSTIVESP ECIFIED	The Exhaustive option requires that the entire reference be loaded into main memory. Since a memory limit has been specified for the MaxMemoryUsage property, it is possible that the entire reference table will not fit within this bound and that the match operation will fail at runtime.
0x80208351	-2145352879	DTS_W_FUZZYLOOKUP_EXA CTMATCHCOLUMNSEXCEE DBYTELIMIT	The cumulative lengths of the columns specified in the exact match joins exceeds the 900 byte limit for index keys. Fuzzy Lookup creates an index on the exact match columns to increase lookup performance and there is a possibility that creation of this index may fail and the lookup will fall back to an alternative, possibly slower, method of finding matches. If performance is a problem, try removing some exact match join columns or reduce the maximum lengths of variable length exact match columns.
0x80208352	-2145352878	DTS_W_FUZZYLOOKUP_EXA CTMATCHINDEXCREATIONF AILED	Failed to create an index for exact match columns. Reverting to alternative fuzzy lookup method.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0x80208353	-2145352877	DTS_W_FUZZYGROUPINGINTERNALPIPELINEWARNING	The following Fuzzy Grouping internal pipeline warning occurred with warning code 0x%1!8.X!: "%2".
0x80208375	-2145352843	DTS_W_XMLSRCOUTPUTCOLUMNLENGTHSETTODEFAULT	No maximum length was specified for the %1 with external data type %2. The SSIS Data Flow Task data type "%3" with a length of %4!d! will be used.
0x80208376	-2145352842	DTS_W_XMLSRCOUTPUTCOLUMNDATATYPEEMAPPEDTOSTRING	The %1 references external data type %2, which cannot be mapped to a SSIS Data Flow Task data type. The SSIS Data Flow Task data type DT_WSTR with a length of %3!d! will be used instead.
0x80208385	-2145352827	DTS_W_NOREDIRECTWITHATTACHEDERROROUTPUTS	No rows will be sent to error output(s). Configure error or truncation dispositions to redirect rows to the error output(s), or delete data flow transformations or destinations that are attached to the error output(s).
0x80208386	-2145352826	DTS_W_REDIRECTWITHNOATTACHEDERROROUTPUTS	Rows sent to the error output(s) will be lost. Add new data flow transformations or destinations to receive error rows, or reconfigure the component to stop redirecting rows to the error output(s).
0x80208391	-2145352815	DTS_W_XMLSRCOUTPUTCOLUMNLENGTHSETTOMAXIMUM	For the %1 with external data type %2, the XML schema specified a maxLength constraint of %3!d!, which exceeds the maximum allowed column length of %4!d!. The SSIS Data Flow Task data type "%5" with a length of %6!d! will be used.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0x802090E4	-2145349404	DTS_W_TXLOOKUP_DUPLICATE_KEYS	The %1 encountered duplicate reference key values when caching reference data. This error occurs in Full Cache mode only. Either remove the duplicate key values, or change the cache mode to PARTIAL or NO_CACHE.
0x802092A7	-2145348953	DTS_W_POTENTIALTRUNCATIONFROMDATAINSERTION	Truncation may occur due to inserting data from data flow column "%1" with a length of %2!d! to database column "%3" with a length of %4!d!.
0x802092A8	-2145348952	DTS_W_POTENTIALTRUNCATIONFROMDATARETRIEVAL	Truncation may occur due to retrieving data from database column "%1" with a length of %2!d! to data flow column "%3" with a length of %4!d!.
0x802092AA	-2145348950	DTS_W_ADODESTBATCHNOTSUPPORTEDFORERRORDISPOSITION	Batch mode is not currently supported when error row disposition is used. The BatchSize property will be set to 1.
0x802092AB	-2145348949	DTS_W_ADODESTNOROWSINSERTED	No rows were successfully inserted into the destination. This may be due to a data type mismatch between columns, or due to the use of a datatype that is unsupported by your ADO.NET provider. Since the error disposition for this component is not "Fail component", error messages are not shown here; set the error disposition to "Fail component" to see error messages here.
0x802092AC	-2145348948	DTS_W_ADODESTPOTENTIALDATALOSS	Potential data loss may occur due to inserting data from input column "%1" with data type "%2" to external column "%3" with data type "%4". If this is intended, an alternative way to do conversion is using a Data Conversion component before ADO NET destination component.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0x802092AD	-2145348947	DTS_W_ADODESTEXTERNAL COLNOTMATCHSCHEMAC OL	The %1 has been out of synchronization with the database column. The latest column has %2. Use advanced editor to refresh available destination columns if needed.
0x802092AE	-2145348946	DTS_W_ADODESTEXTERNAL COLNOTEXIST	The %1 does not exist in the database. It may have been removed or renamed. Use Advanced Editor to refresh the available destination columns if needed.
0x802092AF	-2145348945	DTS_W_ADODESTNEWEXTC OL	A new column with name %1 has been added to the external database table. Use advanced editor to refresh available destination columns if needed.
0x8020930C	-2145348852	DTS_W_NOMATCHOUTPUT GETSNOROWS	No rows will be sent to the no match output. Configure the transformation to redirect rows with no matching entries to the no match output, or delete the data flow transformations or destinations that are attached to the no match output.
0x8020931B	-2145348837	DTS_W_ADODESTINVARIAN TEXCEPTION	Exception received while enumerating ADO.Net providers. The invariant was "%1". The exception message is: "%2"
0xC020822C	-1071611348	DTS_W_UNMAPPEDOUTPU TCOLUMN	The %1 has no input column mapped to it.
0x930D	37645	DTS_W_EXTERNALTABLECO LSOUTOFSYNC	The table "%1" has changed. New columns might have been added to the table.

## Informational Messages

The symbolic names of Integration Services informational messages begin with **DTS\_I\_**.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0x4001100A	1073811466	DTS_I_STARTINGTRANSACTION	Starting distributed transaction for this container.



HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0x4001100B	1073811467	DTS_I_COMMITTINGTRANS ACTION	Committing distributed transaction started by this container.
0x4001100C	1073811468	DTS_I_ABORTINGTRANSACT ION	Aborting the current distributed transaction.
0x40013501	1073820929	DTS_I_GOTMUTEXFROMW AIT	Mutex "%1" was successfully acquired.
0x40013502	1073820930	DTS_I_RELEASEACQUIRED MUTEX	Mutex "%1" was successfully released.
0x40013503	1073820931	DTS_I_NEWMUTEXCREATED	Mutex "%1" was successfully created.
0x40015101	1073828097	DTS_I_DUMP_ON_ANY_ERR	Debug dump files will be generated for any error event.
0x40015102	1073828098	DTS_I_DUMP_ON_CODES	Debug dump files will be generated for the following event codes: "%1"
0x40015103	1073828099	DTS_I_START_DUMP	Event code, 0x%1!8.8X!, triggered generation of debug dump files in the folder "%2".
0x40015104	1073828100	DTS_I_SSIS_INFO_DUMP	Creating SSIS information dump file "%1".
0x40015106	1073828102	DTS_I_FINISH_DUMP	Debug dump files successfully created.
0x40016019	1073831961	DTS_I_PACKAGEMIGRATED	The package format was migrated from version %1!d! to version %2!d!. It must be saved to retain migration changes.
0x4001601A	1073831962	DTS_I_SCRIPTSMIGRATED	The scripts in the package were migrated. The package must be saved to retain migration changes.
0x40016025	1073831973	DTS_I_FTPRECEIVEFILE	Receiving file "%1".
0x40016026	1073831974	DTS_I_FTPSENDFILE	Sending file "%1".
0x40016027	1073831975	DTS_I_FTPFILEEXISTS	File "%1" already exists.
0x40016028	1073831976	DTS_I_FTPERRORLOADING MSG	Cannot get extra error information due to an internal error.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0x40016036	1073831990	DTS_I_FTPDELETEFILE	The attempt to delete file "%1" failed. This may occur when the file does not exist, the file name was spelled incorrectly, or you do not have permissions to delete the file.
0x40016037	1073831991	DTS_I_CONFIGFROMREG	The package is attempting to configure from a registry entry using the registry key "%1".
0x40016038	1073831992	DTS_I_CONFIGFROMENVVAR	The package is attempting to configure from the environment variable "%1".
0x40016039	1073831993	DTS_I_CONFIGFROMINIFILE	The package is attempting to configure from the .ini file "%1".
0x40016040	1073832000	DTS_I_CONFIGFROMSQLSERVER	The package is attempting to configure from SQL Server using the configuration string "%1".
0x40016041	1073832001	DTS_I_CONFIGFROMFILE	The package is attempting to configure from the XML file "%1".
0x40016042	1073832002	DTS_I_CONFIGFROMPARENTVARIABLE	The package is attempting to configure from the parent variable "%1".
0x40016043	1073832003	DTS_I_ATTEMPTINGUPGRADEOFDTS	Attempting an upgrade of SSIS from version "%1" to version "%2". The package is attempting to upgrade the runtime.
0x40016044	1073832004	DTS_I_ATTEMPTINGUPGRADEOFANEXTOBJ	Attempting to upgrade "%1". The package is attempting to upgrade an extensible object.
0x40016045	1073832005	DTS_I_SAVECHECKPOINTSTOFILE	The package will be saving checkpoints to file "%1" during execution. The package is configured to save checkpoints.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0x40016046	1073832006	DTS_I_RESTARTFROMCHECKPOINTFILE	The package restarted from checkpoint file "%1". The package was configured to restart from checkpoint.
0x40016047	1073832007	DTS_I_CHECKPOINTSAVEDTOFILE	Checkpoint file "%1" was updated to record completion of this container.
0x40016048	1073832008	DTS_I_CHECKPOINTFILEDELETED	Checkpoint file "%1" was deleted after successful completion of the package.
0x40016049	1073832009	DTS_I_CHECKPOINTSAVINGTOFILE	Checkpoint file "%1" update starting.
0x40016051	1073832017	DTS_I_CHOSENMAXEXECUTABLES	Based on the system configuration, the maximum concurrent executables are set to %1!d!.
0x40016052	1073832018	DTS_I_MAXEXECUTABLES	Maximum concurrent executables are set to %1!d!.
0x40016053	1073832019	DTS_I_PACKAGESTART	Beginning of package execution.
0x40016054	1073832020	DTS_I_PACKAGEEND	End of package execution.
0x40029161	1073910113	DTS_I_FSTASK_DIRECTORYDELETED	Directory "%1" was deleted.
0x40029162	1073910114	DTS_I_FSTASK_FILEDELETED	File or directory "%1" was deleted.
0x400292A8	1073910440	DTS_I_TRANSFERDBTASK_OVERWRITEDB	Overwriting the database "%1" on the destination server "%2".
0x4002F304	1073935108	DTS_I_SOMETHINGHAPPENED	"%1".
0x4002F323	1073935139	DTS_I_ERRMSGTASK_SKIPPINGERRORMESSAGEALREADYEXISTS	Skipping error message "%1" since it already exists on the destination server.
0x4002F326	1073935142	DTS_I_ERRMSGTASK_TRANSFERREDNERRORMESSAGES	"%1" Error Messages were transferred.
0x4002F351	1073935185	DTS_I_STOREDPROCSTASKS_TRANSFEREDNSPS	The task transferred "%1" Stored Procedures.
0x4002F352	1073935186	DTS_I_TRANSOBJECTSTASKS_TRANSFEREDNOBJECTS	Transferred "%1" objects.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0x4002F358	1073935192	DTS_I_TRANSOBJECTTASK_NOSPSTOTTRANSFER	There are no Stored Procedures to transfer.
0x4002F362	1073935202	DTS_I_TRANSOBJECTTASK_NORULESTOTTRANSFER	There are no Rules to transfer.
0x4002F366	1073935206	DTS_I_TRANSOBJECTTASK_NOTABLESTOTTRANSFER	There are no Tables to transfer.
0x4002F370	1073935216	DTS_I_TRANSOBJECTTASK_NOVIEWSTOTTRANSFER	There are no Views to transfer.
0x4002F374	1073935220	DTS_I_TRANSOBJECTTASK_NOUDFSTOTTRANSFER	There are no User Defined Functions to transfer.
0x4002F378	1073935224	DTS_I_TRANSOBJECTTASK_NODEFAULTSTOTTRANSFER	There are no Defaults to transfer.
0x4002F382	1073935234	DTS_I_TRANSOBJECTTASK_NOUDDTSTOTTRANSFER	There are no User Defined Data Types to transfer.
0x4002F386	1073935238	DTS_I_TRANSOBJECTTASK_NOPFSTOTTRANSFER	There are no Partition Functions to transfer.
0x4002F390	1073935248	DTS_I_TRANSOBJECTTASK_NOPSSTOTTRANSFER	There are no Partition Schemes to transfer.
0x4002F394	1073935252	DTS_I_TRANSOBJECTTASK_NOSCHEMASTOTTRANSFER	There are no Schemas to transfer.
0x4002F398	1073935256	DTS_I_TRANSOBJECTTASK_NOSQLASSEMBLIESTOTTRANSFER	There are no SqlAssemblies to transfer.
0x4002F402	1073935362	DTS_I_TRANSOBJECTTASK_NOAGGREGATESTOTTRANSFER	There are no User Defined Aggregates to transfer.
0x4002F406	1073935366	DTS_I_TRANSOBJECTTASK_NOTYPESTOTTRANSFER	There are no User Defined Types to transfer.
0x4002F410	1073935376	DTS_I_TRANSOBJECTTASK_NOXMLSCHEMACOLLECTIONSTOTTRANSFER	There are no XmlSchemaCollections to transfer.
0x4002F418	1073935384	DTS_I_TRANSOBJECTTASK_NOLOGINSTOTTRANSFER	There are no Logins to transfer.
0x4002F41D	1073935389	DTS_I_TRANSOBJECTTASK_NOUSERSTOTTRANSFER	There are no Users to transfer.
0x4002F41E	1073935390	DTS_I_TRANSOBJECTTASK_TRUNCATINGTABLE	Truncating table "%1"

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0x40043006	1074016262	DTS_I_EXECUTIONPHASE_PR EPAREFOREXECUTE	Prepare for Execute phase is beginning.
0x40043007	1074016263	DTS_I_EXECUTIONPHASE_PR EEXECUTE	Pre-Execute phase is beginning.
0x40043008	1074016264	DTS_I_EXECUTIONPHASE_P OSTEXECUTE	Post Execute phase is beginning.
0x40043009	1074016265	DTS_I_EXECUTIONPHASE_CL EANUP	Cleanup phase is beginning.
0x4004300A	1074016266	DTS_I_EXECUTIONPHASE_V ALIDATE	Validation phase is beginning.
0x4004300B	1074016267	DTS_I_ROWS_WRITTEN	"%1" wrote %2!d! rows.
0x4004300C	1074016268	DTS_I_EXECUTIONPHASE_EX ECUTE	Execute phase is beginning.
0x4004800C	1074036748	DTS_I_CANTRELIEVEPRESSU RE	The buffer manager detected that the system was low on virtual memory, but was unable to swap out any buffers. %1!d! buffers were considered and %2!d! were locked. Either not enough memory is available to the pipeline because not enough is installed, other processes are using it, or too many buffers are locked.
0x4004800D	1074036749	DTS_I_CANTALLOCATEMEM ORYPRESSURE	The buffer manager failed a memory allocation call for %3!d! bytes, but was unable to swap out any buffers to relieve memory pressure. %1!d! buffers were considered and %2!d! were locked. Either not enough memory is available to the pipeline because not enough are installed, other processes were using it, or too many buffers are locked.
0x4004800E	1074036750	DTS_I_ALLOCATEDDURING MEMORYPRESSURE	The buffer manager has allocated %1!d! bytes, even though the memory pressure has been detected and repeated attempts to swap buffers have failed.
0x400490F4	1074041076	DTS_I_TXLOOKUP_CACHE_P ROGRESS	%1 has cached %2!d! rows.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0x400490F5	1074041077	DTS_I_TXLOOKUP_CACHE_FINAL	%1 has cached a total of %2!d! rows.
0x4020206D	1075847277	DTS_I_RAWSOURCECOLUMNS	The raw source adapter opened a file, but the file contains no columns. The adapter will not produce data. This could indicate a damaged file, or that there are zero columns and, therefore, no data.
0x402020DA	1075847386	DTS_I_OLEDBINFORMATIONMESSAGE	An OLE DB informational message is available.
0x40208327	1075872551	DTS_I_TXFUZZYLOOKUP_EXACT_MATCH_PERF_COLLATIONS_DONT_MATCH	Fuzzy match performance can be improved if the exact join FuzzyComparisonFlags on the input column "%1" are set to match with the default SQL collation for reference table column "%2". It is also necessary that no fold flags are set in FuzzyComparisonFlagsEx.
0x40208328	1075872552	DTS_I_TXFUZZYLOOKUP_EXACT_MATCH_PERF_INDEX_MISSING	Fuzzy match performance can be improved if an index is created upon the reference table across all of the specified exact match columns.
0x40208387	1075872647	DTS_I_DISPSNOTREVIEWED	Error and truncation dispositions were not reviewed. Make sure this component is configured to redirect rows to error outputs, if you wish to further transform those rows.
0x402090DA	1075876058	DTS_I_TXAGG_WORKSPACE_REHASH	The Aggregate transformation has encountered %1!d! key combinations. It has to re-hash data because the number of key combinations is more than expected. The component can be configured to avoid data re-hash by adjusting the Keys, KeyScale, and AutoExtendFactor properties.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0x402090DB	1075876059	DTS_I_TXAGG_COUNTDISTINCT_REHASH	The Aggregate transformation has encountered %1!d! distinct values while performing a "count distinct" aggregation on "%2". The transformation will re-hash data because the number of distinct values is more than expected. The component can be configured to avoid data re-hash by adjusting the CountDistinctKeys, CountDistinctKeyScale, and AutoExtendFactor properties.
0x402090DC	1075876060	DTS_I_STARTPROCESSINGFILE	The processing of file "%1" has started.
0x402090DD	1075876061	DTS_I_FINISHPROCESSINGFILE	The processing of file "%1" has ended.
0x402090DE	1075876062	DTS_I_TOTALDATAROWSPROCESSEDFORFILE	The total number of data rows processed for file "%1" is %2!l64d!.
0x402090DF	1075876063	DTS_I_FINALCOMMITSTARTED	The final commit for the data insertion in "%1" has started.
0x402090E0	1075876064	DTS_I_FINALCOMMITENDED	The final commit for the data insertion in "%1" has ended.
0x402090E1	1075876065	DTS_I_BEGINHASHINGCACHE	%1!u! rows are added to the cache. The system is processing the rows.
0x402090E2	1075876066	DTS_I_SUCCEEDEDHASHINGCACHE	The %1 processed %2!u! rows in the cache. The processing time was %3 seconds. The cache used %4!l64u! bytes of memory.
0x402090E3	1075876067	DTS_I_FAILEDHASHINGCACHE	The %1 failed to process the rows in the cache. The processing time was %2 second(s).
0x402090E4	1075876068	DTS_I_SUCCEEDEDPREPARINGCACHE	The %1 succeeded in preparing the cache. The preparation time was %2 seconds.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0x40209314	1075876628	DTS_I_TXLOOKUP_PARTIALP ERF	The %1 has performed the following operations: processed %2!l64u! rows, issued %3!l64u! database commands to the reference database, and performed %4!l64u! lookups using partial cache.
0x40209315	1075876629	DTS_I_TXLOOKUP_PARTIALP ERF2	The %1 has performed the following operations: processed %2!l64u! rows, issued %3!l64u! database commands to the reference database, performed %4!l64u! lookups using partial cache and %5!l64u! lookups using the cache for rows with no matching entries in the initial lookup.
0x40209316	1075876630	DTS_I_CACHEFILEWRITESTA RTED	The %1 is writing the cache to file "%2".
0x40209317	1075876631	DTS_I_CACHEFILEWRITESUC CEEDED	The %1 has written the cache to file "%2".
0x4020F42C	1075901484	DTS_I_OLEDBDESTZEROMA XCOMMITSIZE	The Maximum insert commit size property of the OLE DB destination "%1" is set to 0. This property setting can cause the running package to stop responding. For more information, see the F1 Help topic for OLE DB Destination Editor (Connection Manager Page).

## General and Event Messages

The symbolic names of Integration Services error messages begin with **DTS\_MSG\_**.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0x1	1	DTS_MSG_CATEGORY_SERVI CE_CONTROL	Incorrect function.
0x2	2	DTS_MSG_CATEGORY_RUN NING_PACKAGE_MANAGE MENT	The system cannot find the file specified.
0x100	256	DTS_MSG_SERVER_STARTIN G	Starting Microsoft SSIS Service.  Server version %1



HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0x101	257	DTS_MSG_SERVER_STARTED	Microsoft SSIS Service started.  Server version %1
0x102	258	DTS_MSG_SERVER_STOPPING	The wait operation timed out.
0x103	259	DTS_MSG_SERVER_STOPPED	No more data is available.
0x104	260	DTS_MSG_SERVER_START_FAILED	Microsoft SSIS Service failed to start.  Error: %1
0x105	261	DTS_MSG_SERVER_STOP_ERROR	Error stopping Microsoft SSIS Service.  Error: %1
0x110	272	DTS_MSG_SERVER_MISSING_CONFIG	Microsoft SSIS Service configuration file does not exist.  Loading with default settings.
0x111	273	DTS_MSG_SERVER_BAD_CONFIG	Microsoft SSIS Service configuration file is incorrect.  Error reading config file: %1  Loading server with default settings.
0x112	274	DTS_MSG_SERVER_MISSING_CONFIG_REG	Microsoft SSIS Service:  Registry setting specifying configuration file does not exist.  Attempting to load default config file.
0x150	336	DTS_MSG_SERVER_STOPPING_PACKAGE	Microsoft SSIS Service: stopping running package.  Package instance ID: %1  Package ID: %2  Package name: %3  Package description: %4  Package started by: %5.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0x40013000	1073819648	DTS_MSG_PACKAGESTART	Package "%1" started.
0x40013001	1073819649	DTS_MSG_PACKAGESUCCESS	Package "%1" finished successfully.
0x40013002	1073819650	DTS_MSG_PACKAGECANCEL	Package "%1" has been cancelled.
0x40013003	1073819651	DTS_MSG_PACKAGEFAILURE	Package "%1" failed.
0x40013004	1073819652	DTS_MSG_CANTDELAYLOADDLL	Module %1 cannot load DLL %2 to call entry point %3 because of error %4. The product requires that DLL to run, but the DLL could not be found on the path.
0x40013005	1073819653	DTS_MSG_CANTDELAYLOADDLLFUNCTION	Module %1 loaded DLL %2, but cannot find entry point %3 because of error %4. The named DLL could not be found on the path, and the product requires that DLL to run.
0x40103100	1074802944	DTS_MSG_EVENTLOGENTRY	Event Name: %1  Message: %9  Operator: %2  Source Name: %3  Source ID: %4  Execution ID: %5  Start Time: %6  End Time: %7  Data Code: %8

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0x40103101	1074802945	DTS_MSG_EVENTLOGENTRY _PREEXECUTE	Event Name: %1  Message: %9  Operator: %2  Source Name: %3  Source ID: %4  Execution ID: %5  Start Time: %6  End Time: %7  Data Code: %8
0x40103102	1074802946	DTS_MSG_EVENTLOGENTRY _POSTEXECUTE	Event Name: %1  Message: %9  Operator: %2  Source Name: %3  Source ID: %4  Execution ID: %5  Start Time: %6  End Time: %7  Data Code: %8
0x40103103	1074802947	DTS_MSG_EVENTLOGENTRY _PREVALIDATE	Event Name: %1  Message: %9  Operator: %2  Source Name: %3  Source ID: %4  Execution ID: %5  Start Time: %6  End Time: %7  Data Code: %8

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0x40103104	1074802948	DTS_MSG_EVENTLOGENTRY _POSTVALIDATE	Event Name: %1  Message: %9  Operator: %2  Source Name: %3  Source ID: %4  Execution ID: %5  Start Time: %6  End Time: %7  Data Code: %8
0x40103105	1074802949	DTS_MSG_EVENTLOGENTRY _WARNING	Event Name: %1  Message: %9  Operator: %2  Source Name: %3  Source ID: %4  Execution ID: %5  Start Time: %6  End Time: %7  Data Code: %8
0x40103106	1074802950	DTS_MSG_EVENTLOGENTRY _ERROR	Event Name: %1  Message: %9  Operator: %2  Source Name: %3  Source ID: %4  Execution ID: %5  Start Time: %6  End Time: %7  Data Code: %8

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0x40103107	1074802951	DTS_MSG_EVENTLOGENTRY _TASKFAILED	Event Name: %1  Message: %9  Operator: %2  Source Name: %3  Source ID: %4  Execution ID: %5  Start Time: %6  End Time: %7  Data Code: %8
0x40103108	1074802952	DTS_MSG_EVENTLOGENTRY _PROGRESS	Event Name: %1  Message: %9  Operator: %2  Source Name: %3  Source ID: %4  Execution ID: %5  Start Time: %6  End Time: %7  Data Code: %8
0x40103109	1074802953	DTS_MSG_EVENTLOGENTRY _EXECSTATCHANGE	Event Name: %1  Message: %9  Operator: %2  Source Name: %3  Source ID: %4  Execution ID: %5  Start Time: %6  End Time: %7  Data Code: %8

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0x4010310A	1074802954	DTS_MSG_EVENTLOGENTRY _VARVALCHANGE	Event Name: %1  Message: %9  Operator: %2  Source Name: %3  Source ID: %4  Execution ID: %5  Start Time: %6  End Time: %7  Data Code: %8
0x4010310B	1074802955	DTS_MSG_EVENTLOGENTRY _CUSTOMEVENT	Event Name: %1  Message: %9  Operator: %2  Source Name: %3  Source ID: %4  Execution ID: %5  Start Time: %6  End Time: %7  Data Code: %8
0x4010310C	1074802956	DTS_MSG_EVENTLOGENTRY _PACKAGESTART	Event Name: %1  Message: %9  Operator: %2  Source Name: %3  Source ID: %4  Execution ID: %5  Start Time: %6  End Time: %7  Data Code: %8

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0x4010310D	1074802957	DTS_MSG_EVENTLOGENTRY_PACKAGEEND	Event Name: %1  Message: %9  Operator: %2  Source Name: %3  Source ID: %4  Execution ID: %5  Start Time: %6  End Time: %7  Data Code: %8
0x4010310E	1074802958	DTS_MSG_EVENTLOGENTRY_INFORMATION	Event Name: %1  Message: %9  Operator: %2  Source Name: %3  Source ID: %4  Execution ID: %5  Start Time: %6  End Time: %7  Data Code: %8

## Success Messages

The symbolic names of Integration Services success messages begin with **DTS\_S\_**.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0x40003	262147	DTS_S_NULLDATA	The value is NULL.
0x40005	262149	DTS_S_TRUNCATED	The string value was truncated. The buffer received a string that was too long for the column, and the string was truncated by the buffer.
0x200001	2097153	DTS_S_EXPREVALTRUNCATIONOCCURRED	A truncation occurred during evaluation of the expression. The truncation occurred during evaluation, which may include any point in an intermediate step.

# Data Flow Component Error Messages

The symbolic names of Integration Services error messages begin with **DTSBC\_E\_**, where "BC" refers to the native base class from which most Microsoft data flow components are derived.

HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC8000002	-939524094	DTSBC_E_INCORRECTEXACT NUMBEROFTOTALOUTPUTS	The total number of outputs and error outputs, %1!lu!, is incorrect. There must be exactly %2!lu!.
0xC8000003	-939524093	DTSBC_E_FAILEDTOGETOUT PUTBYINDEX	Cannot retrieve output with index %1!lu!.
0xC8000005	-939524091	DTSBC_E_INCORRECTEXACT NUMBEROFFERROROUTPUTS	The number of error outputs, %1!lu!, is incorrect. There must be exactly %2!lu!.
0xC8000006	-939524090	DTSBC_E_INVALIDVALIDATI ONSTATUSVALUE	Incorrect validation status value, "%1!lu! ". It must be one of the values found in the DTSValidationStatus enumeration.
0xC8000007	-939524089	DTSBC_E_INPUTHASNOOUT PUT	The input "%1!lu!" has no synchronous output.
0xC8000008	-939524088	DTSBC_E_INPUTHASNOERR OROUTPUT	The input "%1!lu!" has no synchronous error output.
0xC8000009	-939524087	DTSBC_E_INVALIDHTPIVAL UE	The HowToProcessInput value, %1!lu!, is not valid. It must be one of the values from the HowToProcessInput enumeration.
0xC800000A	-939524086	DTSBC_E_FAILEDTOGETCOL INFO	Failed to get information for row "%1!ld!", column "%2!ld!" from the buffer. The error code returned was 0x%3!8.X!.
0xC800000B	-939524085	DTSBC_E_FAILEDTOSETCOLI NFO	Failed to set information for row "%1!ld!", column "%2!ld!" into the buffer. The error code returned was 0x%3!8.X!.
0xC800000C	-939524084	DTSBC_E_INVALIDPROPERT Y	The property "%1" is not valid.
0xC800000D	-939524083	DTSBC_E_PROPERTYNOTFO UND	The property "%1" was not found.



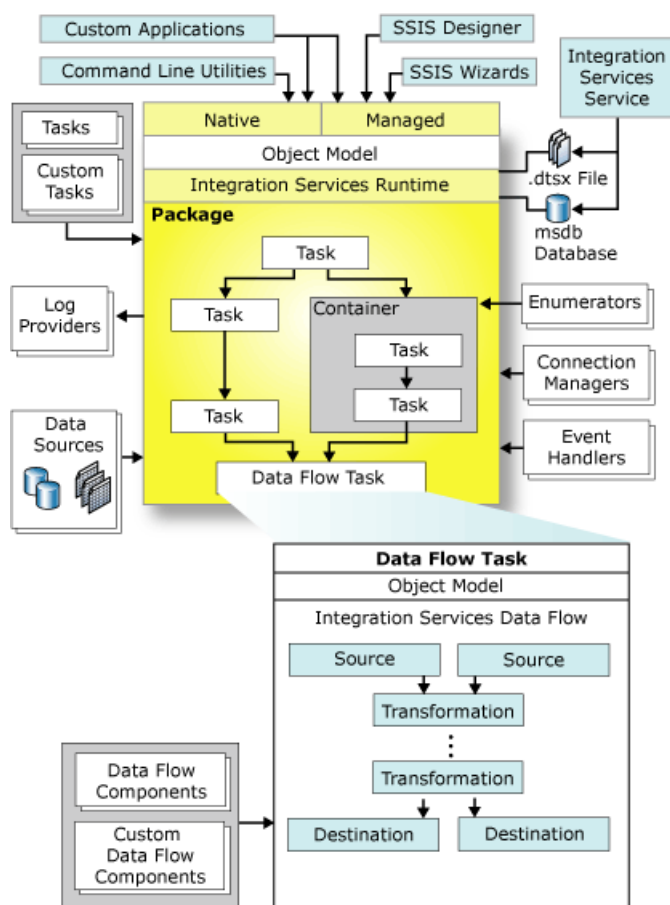
HEXADECIMAL CODE	DECIMAL CODE	SYMBOLIC NAME	DESCRIPTION
0xC8000010	-939524080	DTSBC_E_READONLYPROPERTY	Error assigning a value to the read-only property "%1".
0xC8000011	-939524079	DTSBC_E_CANTINSERTOUTPUTCOLUMN	The %1 does not allow the insertion of output columns.
0xC8000012	-939524078	DTSBC_E_OUTPUTCOLUMNMETADATAMISMATCH	The output columns' metadata does not match the associated input columns' metadata. The output columns' metadata will be updated.
0xC8000013	-939524077	DTSBC_E_OUTPUTCOLUMNSMISSING	There are input columns that do not have associated output columns. The output columns will be added.
0xC8000014	-939524076	DTSBC_E_TOOMANYOUTPUTCOLUMNS	There are output columns that do not have associated input columns. The output columns will be removed.
0xC8000015	-939524075	DTSBC_E_OUTPUTCOLUMNMETADATAMISMATCHUNMAP	The output columns' metadata does not match the associated input columns' metadata. The input columns will be unmapped.
0xC8000016	-939524074	DTSBC_E_UNMAPINPUTCOLUMNS	There are input columns that do not have associated output columns. The input columns will be unmapped.
0xC8000017	-939524073	DTSBC_E_MULTIPLEINPUTCOLS	There is an input column associated with an output column, and that output column is already associated with another input column on the same input.
0xC8000018	-939524072	DTSBC_E_CANTINSERTEXTERNALMETADATACOLUMN	The %1 does not allow the insertion of external metadata columns.

# Integration Services Programming Overview

3/24/2017 • 3 min to read • [Edit Online](#)

SQL Server Integration Services has an architecture that separates data movement and transformation from package control flow and management. There are two distinct engines that define this architecture and that can be automated and extended when programming Integration Services. The run-time engine implements the control flow and package management infrastructure that lets developers control the flow of execution and set options for logging, event handlers, and variables. The data flow engine is a specialized, high performance engine that is exclusively dedicated to extracting, transforming, and loading data. When programming Integration Services, you will be programming against these two engines.

The following image depicts the architecture of Integration Services.



## Integration Services Run-time Engine

The Integration Services run-time engine controls the management and execution of packages, by implementing the infrastructure that enables execution order, logging, variables, and event handling. Programming the Integration Services run-time engine lets developers automate the creation, configuration, and execution of packages and create custom tasks and other extensions.

For more information, see [Extending the Package with the Script Task](#), [Developing a Custom Task](#), and [Building Packages Programmatically](#).

## Integration Services Data Flow Engine

The data flow engine manages the data flow task, which is a specialized, high performance task dedicated to

moving and transforming data from disparate sources. Unlike other tasks, the data flow task contains additional objects called data flow components, which can be sources, transformations, or destinations. These components are the core moving parts of the task. They define the movement and transformation of data. Programming the data flow engine lets developers automate the creation and configuration of the components in a data flow task, and create custom components.

For more information, see [Extending the Data Flow with the Script Component](#), [Developing a Custom Data Flow Component](#), and [Building Packages Programmatically](#).

## Supported Languages

Integration Services fully supports the Microsoft .NET Framework. This lets developers program Integration Services in their choice of .NET-compliant languages. Although both the run-time engine and the data flow engine are written in native code, they are both available through a fully managed object model.

You can program Integration Services packages, custom tasks, and components in Microsoft Visual Studio or in another code or text editor. Visual Studio offers the developer many tools and features to simplify and accelerate the iterative cycles of coding, debugging, and testing. Visual Studio also makes deployment easier. However, you do not need Visual Studio to compile and build Integration Services code projects. The .NET Framework SDK includes the Visual Basic and Visual C# compilers and related tools.

### IMPORTANT

By default, the .NET Framework is installed with SQL Server, but the .NET Framework SDK is not. Unless the SDK is installed on the computer and the SDK documentation is included in the Books Online collection, links to SDK content in this section will not work. After you have installed the .NET Framework SDK, you can add the SDK documentation to the Books Online collection and table of contents by following the instructions in [Add or Remove Product Documentation for SQL Server](#).

The Integration Services Script task and Script component use Microsoft Visual Studio Tools for Applications (VSTA) as an embedded scripting environment. VSTA supports Microsoft Visual Basic and Microsoft Visual C#.

### NOTE

The Integration Services application programming interfaces are incompatible with COM-based scripting languages such as VBScript.

## Locating Assemblies

In SQL Server 2017, the Integration Services assemblies were upgraded to .NET 4.0. There is a separate global assembly cache for .NET 4, located in `<drive>:\Windows\Microsoft.NET\assembly`. You can find all of the Integration Services assemblies under this path, usually in the GAC\_MSIL folder.

As in previous versions of SQL Server, the core Integration Services extensibility .dll files are also located at `<drive>:\Program Files\Microsoft SQL Server\100\SDK\Assemblies`.

## Commonly Used Assemblies

The following table lists the assemblies that are frequently used when programming Integration Services using the .NET Framework.

ASSEMBLY	DESCRIPTION
Microsoft.SqlServer.ManagedDTS.dll	Contains the managed run-time engine.

ASSEMBLY	DESCRIPTION
Microsoft.SqlServer.RuntimeWrapper.dll	Contains the primary interop assembly (PIA), or wrapper, for the native run-time engine.
Microsoft.SqlServer.PipelineHost.dll	Contains the managed data flow engine.
Microsoft.SqlServer.PipelineWrapper.dll	Contains the primary interop assembly (PIA), or wrapper, for the native data flow engine.

# Understanding Synchronous and Asynchronous Transformations

3/24/2017 • 3 min to read • [Edit Online](#)

To understand the difference between a synchronous and an asynchronous transformation in Integration Services, it is easiest to start with an understanding of a synchronous transformation. If a synchronous transformation does not meet your needs, your design might require an asynchronous transformation.

## Synchronous Transformations

A synchronous transformation processes incoming rows and passes them on in the data flow one row at a time. Output is synchronous with input, meaning that it occurs at the same time. Therefore, to process a given row, the transformation does not need information about other rows in the data set. In the actual implementation, rows are grouped into buffers as they pass from one component to the next, but these buffers are transparent to the user, and you can assume that each row is processed separately.

An example of a synchronous transformation is the Data Conversion transformation. For each incoming row, it converts the value in the specified column and sends the row on its way. Each discrete conversion operation is independent of all the other rows in the data set.

In Integration Services scripting and programming, you specify a synchronous transformation by looking up the ID of a component's input and assigning it to the **SynchronousInputID** property of the component's outputs. This tells the data flow engine to process each row from the input and send each row automatically to the specified outputs. If you want every row to go to every output, you do not have to write any additional code to output the data. If you use the **ExclusionGroup** property to specify that rows should only go to one or another of a group of outputs, as in the Conditional Split transformation, you must call the **DirectRow** method to select the appropriate destination for each row. When you have an error output, you must call **DirectErrorRow** to send rows with problems to the error output instead of the default output.

## Asynchronous Transformations

You might decide that your design requires an asynchronous transformation when it is not possible to process each row independently of all other rows. In other words, you cannot pass each row along in the data flow as it is processed, but instead must output data asynchronously, or at a different time, than the input. For example, the following scenarios require an asynchronous transformation:

- The component has to acquire multiple buffers of data before it can perform its processing. An example is the Sort transformation, where the component has to process the complete set of rows in a single operation.
- The component has to combine rows from multiple inputs. An example is the Merge transformation, where the component has to examine multiple rows from each input and then merge them in sorted order.
- There is no one-to-one correspondence between input rows and output rows. An example is the Aggregate transformation, where the component has to add a row to the output to hold the computed aggregate values.

In Integration Services scripting and programming, you specify an asynchronous transformation by assigning a value of 0 to the **SynchronousInputID** property of the component's outputs. This tells the data flow engine not to send each row automatically to the outputs. Then you must write code to send each row explicitly to the appropriate output by adding it to the new output buffer that is created for the output of an asynchronous transformation.

**NOTE**

Since a source component must also explicitly add each row that it reads from the data source to its output buffers, a source resembles a transformation with asynchronous outputs.

It would also be possible to create an asynchronous transformation that emulates a synchronous transformation by explicitly copying each input row to the output. By using this approach, you could rename columns or convert data types or formats. However this approach degrades performance. You can achieve the same results with better performance by using built-in Integration Services components, such as Copy Column or Data Conversion.

## See Also

[Creating a Synchronous Transformation with the Script Component](#)

[Creating an Asynchronous Transformation with the Script Component](#)

[Developing a Custom Transformation Component with Synchronous Outputs](#)

[Developing a Custom Transformation Component with Asynchronous Outputs](#)

# Working with Connection Managers Programmatically

3/24/2017 • 2 min to read • [Edit Online](#)

In Integration Services, the `AcquireConnection` method of the associated connection manager class is the method that you call most often when you are working with connection managers in managed code. When you write managed code, you have to call the `AcquireConnection` method to use the functionality of a connection manager. You have to call this method regardless of whether you are writing managed code in a Script task, Script component, custom object, or custom application.

To call the `AcquireConnection` method successfully, you have to know the answers to the following questions:

- **Which connection managers return a managed object from the `AcquireConnection` method?**

Many connection managers return unmanaged COM objects (`System.__ComObject`) and these objects cannot easily be used from managed code. The list of these connection managers includes the frequently used OLE DB connection manager.

- **For those connection managers that return a managed object, what objects do their `AcquireConnection` methods return?**

To cast the return value to the appropriate type, you have to know what type of object the `AcquireConnection` method returns. For example, the `AcquireConnection` method for the ADO.NET connection manager returns an open `SqlConnection` object when you use the `SqlClient` provider. However, the `AcquireConnection` method for the File connection manager just returns a string.

This topic answers these questions for the connection managers that are included with Integration Services.

## Connection Managers That Do Not Return a Managed Object

The following table lists the connection managers that return a native COM object (`System.__ComObject`) from the `AcquireConnection` method. These unmanaged objects cannot easily be used from managed code.

CONNECTION MANAGER TYPE	CONNECTION MANAGER NAME
ADO	ADO Connection Manager
MSOLAP90	Analysis Services Connection Manager
EXCEL	Excel Connection Manager
FTP	FTP Connection Manager
HTTP	HTTP Connection Manager
ODBC	ODBC Connection Manager
OleDb	OLE DB Connection Manager

Typically, you can use an ADO.NET connection manager from managed code to connect to an ADO, Excel, ODBC, or OLE DB data source.

## Return Values from the `AcquireConnection` Method

The following table lists the connection managers that return a managed object from the `AcquireConnection` method. These managed objects can easily be used from managed code.

CONNECTION MANAGER TYPE	CONNECTION MANAGER NAME	TYPE OF RETURN VALUE	ADDITIONAL INFORMATION
ADO.NET	ADO.NET Connection Manager	<b><code>System.Data.SqlClient.SqlConnection</code></b>	
FILE	File Connection Manager	<b><code>System.String</code></b>	Path to the file.

CONNECTION MANAGER TYPE	CONNECTION MANAGER NAME	TYPE OF RETURN VALUE	ADDITIONAL INFORMATION
FLATFILE	Flat File Connection Manager	<b>System.String</b>	Path to the file.
MSMQ	MSMQ Connection Manager	<b>System.Messaging.MessageQueue</b>	
MULTIFILE	Multiple Files Connection Manager	<b>System.String</b>	Path to one of the files.
MULTIFLATFILE	Multiple Flat Files Connection Manager	<b>System.String</b>	Path to one of the files.
SMO Server	SMO Connection Manager	<b>Microsoft.SqlServer.Management.Smo.Server</b>	
SMTP	SMTP Connection Manager	<b>System.String</b>	For example: <code>SmtpServer=&lt;server name&gt;;UseWindowsAuthentication=True;En</code>
WMI	WMI Connection Manager	<b>System.Management.ManagementScope</b>	
SQLMOBILE	SQL Server Compact Connection Manager	<b>System.Data.SqlServerCe.SqlCeConnection</b>	

## See Also

- [Connecting to Data Sources in the Script Task](#)
- [Connecting to Data Sources in the Script Component](#)
- [Connecting to Data Sources in a Custom Task](#)



# Comparing Scripting Solutions and Custom Objects

3/24/2017 • 1 min to read • [Edit Online](#)

An Integration Services Script task or Script component can implement much of the same functionality that is possible in a custom managed task or data flow component. Here are some considerations to help you choose the appropriate type of task for your needs:

- If the configuration or functionality is specific to an individual package, you should use the Script task or the Script component instead of a developing a custom object.
- If the functionality is generic, and might be used in the future for other packages and by other developers, you should create a custom object instead of using a scripting solution. You can use a custom object in any package, whereas a script can be used only in the package for which it was created.
- If the code will be reused within the same package, you should consider creating a custom object. Copying code from one Script task or component to another leads to reduced maintainability by making it more difficult to maintain and update multiple copies of the code.
- If the implementation will change over time, consider using a custom object. Custom objects can be developed and deployed separately from the parent package, whereas an update made to a scripting solution requires the redeployment of the whole package.

## See Also

[Extending Packages with Custom Objects](#)

# Building, Deploying, and Debugging Custom Objects

3/24/2017 • 8 min to read • [Edit Online](#)

After you have written the code for a custom object for Integration Services, you must build the assembly, deploy it, and integrate it into SSIS Designer to make it available for use in packages, and test and debug it.

## Steps in Building, Deploying, and Debugging a Custom Object for Integration Services

You have already written the custom functionality for your object. Now you have to test it and to make it available to users. The steps are very similar for all the types of custom objects that you can create for Integration Services.

Here are the steps to build, deploy, and test it.

1. [Sign](#) the assembly to be generated with a strong name.
2. [Build](#) the assembly.
3. [Deploy](#) the assembly by moving or copying it to the appropriate Integration Services folder.
4. [Install](#) the assembly in the global assembly cache (GAC).

The object is automatically added to the Toolbox.

5. [Troubleshoot](#) the deployment, if necessary.
6. [Test](#) and debug your code.

You can now use SSIS Designer in SQL Server Data Tools (SSDT) to create, maintain, and run packages that target different versions of SQL Server. For more info about the impact of this improvement on your custom extensions, see [Getting your SSIS custom extensions to be supported by the multi-version support of SSDT 2015 for SQL Server 2016](#)

## Signing the Assembly

When an assembly is meant to be shared, it must be installed in the global assembly cache. After the assembly has been added to the global assembly cache, the assembly can be used by applications such as SQL Server Data Tools (SSDT). A requirement of the global assembly cache is that the assembly must be signed with a strong name, which guarantees that an assembly is globally unique. A strong-named assembly has a fully qualified name that includes the name, culture, public key, and version number of the assembly. The runtime uses this information to locate the assembly and to differentiate it from other assemblies with the same name.

To sign an assembly with a strong name, you must first have or create a public/private key pair. This public and private cryptographic key pair is used at build time to create a strong-named assembly.

For more information about strong names and on the steps that you must follow to sign an assembly, see the following topics in the .NET Framework SDK documentation:

- Strong-Named Assemblies
- Creating a Key Pair
- Signing an Assembly with a Strong Name

You can easily sign your assembly with a strong name in Visual Studio at build time. In the **Project**

**Properties** dialog box, select the **Signing** tab. Select the option to **Sign the assembly** and then provide the path of the key (.snk) file.

## Building the Assembly

After signing the project, you must build or rebuild the project or the solution by using the commands available on the **Build** menu of SQL Server Data Tools. Your solution may contain a separate project for a custom user interface, which must also be signed with a strong name, and can be built at the same time.

The most convenient method for performing the next two steps—deploying the assembly and installing it in the global assembly cache—is to script these steps as a post-build event in Visual Studio. Build events are available from the **Compile** page of Project Properties for a Visual Basic project, and from the **Build Events** page for a C# project. The full path is required for command prompt utilities such as **gacutil.exe**. Quotation marks are required both around paths that contain spaces and around macros such as \$(TargetPath) that expand to paths that contain spaces.

Here is an example of a post-build event command line for a custom log provider:

```
"C:\Program Files (x86)\Microsoft SDKs\Windows\v7.0A\bin\NETFX 4.0 Tools\gacutil.exe" -u $(TargetName)
"C:\Program Files (x86)\Microsoft SDKs\Windows\v7.0A\bin\NETFX 4.0 Tools\gacutil.exe" -i $(TargetFileName)
copy $(TargetFileName) "C:\Program Files\Microsoft SQL Server\130\DTS\LogProviders "
```

## Deploying the Assembly

The SSIS Designer locates the custom objects available for use in packages by enumerating the files found in a series of folders that are created when SQL Server Integration Services is installed. When the default SQL Server installation settings are used, this set of folders is located under **C:\Program Files\Microsoft SQL Server\130\DTS**. However if you create a setup program for your custom object, you should check the value of the **HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Microsoft SQL Server\130\SSIS\Setup\DtsPath** registry key to verify the location of this folder.

### NOTE

For info about how to deploy custom components to work well with the multi-version support in SQL Server Data Tools, see [Getting your SSIS custom extensions to be supported by the multi-version support of SSDT 2015 for SQL Server 2016](#).

You can put the assembly in the folder in two ways:

- Move or copy the compiled assembly to the appropriate folder after building it. (For convenience, you can include the copy command in a Post-build Event.)
- Build the assembly directly in the appropriate folder.

The following deployment folders under **C:\Program Files\Microsoft SQL Server\130\DTS** are used for the various types of custom objects:

CUSTOM OBJECT	DEPLOYMENT FOLDER
Task	Tasks
Connection manager	Connections
Log provider	LogProviders

CUSTOM OBJECT	DEPLOYMENT FOLDER
Data flow component	PipelineComponents

#### NOTE

Assemblies are copied to these folders to support the enumeration of available tasks, connection managers, and so on. Therefore you do not have to deploy assemblies that contain only the custom user interface for custom objects to these folders.

## Installing the Assembly in the Global Assembly Cache

To install the task assembly into the global assembly cache (GAC), use the command line tool **gacutil.exe**, or drag the assemblies to the `%system%\assembly` directory. For convenience, you can also include the call to **gacutil.exe** in a Post-build Event.

The following command installs a component named *MyTask.dll* into the GAC by using **gacutil.exe**.

```
gacutil /if MyTask.dll
```

You must close and reopen SSIS Designer after you install a new version of your custom object. If you have installed earlier versions of your custom object in the global assembly cache, you must remove them before installing the new version. To uninstall an assembly, run **gacutil.exe** and specify the assembly name with the `/u` option.

For more information about the global assembly cache, see Global Assembly Cache Tool (Gacutil.exe) in the .NET Framework Tools.

## Troubleshooting the Deployment

If your custom object appears in the **Toolbox** or the list of available objects, but you are not able to add it to a package, try the following:

1. Look in the global assembly cache for multiple versions of your component. If there are multiple versions of the component in the global assembly cache, the designer may not be able to load your component. Delete all instances of the assembly from the global assembly cache, and re-add the assembly.
2. Make sure that only a single instance of the assembly exists in the deployment folder.
3. Refresh the Toolbox.
4. Attach Visual Studio to **devenv.exe** and set a breakpoint to step through your initialization code to ensure that no exceptions occur.

## Testing and Debugging Your Code

The simplest approach to debugging the run-time methods of a custom object is to start **dtexec.exe** from Visual Studio after building your custom object and run a package that uses the component.

If you want to debug the component's design-time methods, such as the **Validate** method, open a package that uses the component in a second instance of Visual Studio, and attach to its **devenv.exe** process.

If you also want to debug the component's run-time methods when a package is open and running in SSIS designer, you must force a pause in the execution of the package so that you can also attach to the **DtsDebugHost.exe** process.

**To debug an object's run-time methods by attaching to dtexec.exe**

1. Sign and build your project in the Debug configuration, deploy it, and install it in the global assembly cache as described in this topic.
2. On the **Debug** tab of **Project Properties**, select **Start external program** as the **Start Action**, and locate **dtexec.exe**, which is installed by default in C:\Program Files\Microsoft SQL Server\130\DTS\Binn.
3. In the **Command line options** text box, under **Start Options**, enter the command line arguments required to run a package that uses your component. Often the command-line argument will consist of the /F[ILE] switch followed by the path and file name of the .dtsx file. For more information, see [dtexec Utility](#).
4. Set breakpoints in the source code where appropriate in the run-time methods of your component.
5. Run your project.

#### To debug a custom object's design-time methods by attaching to SQL Server Data Tools

1. Sign and build your project in the Debug configuration, deploy it, and install it in the global assembly cache as described in this topic.
2. Set breakpoints in the source code where appropriate in the design-time methods of your custom object.
3. Open a second instance of Visual Studio and load an Integration Services project that contains a package that uses the custom object.
4. From the first instance of Visual Studio, attach to the second instance of **devenv.exe** in which the package is loaded by selecting **Attach to Process** from the **Debug** menu of the first instance.
5. Run the package from the second instance of Visual Studio.

#### To debug a custom object's run-time methods by attaching to SQL Server Data Tools

1. After you have completed the steps listed in the previous procedure, force a pause in the execution of your package so that you can attach to **DtsDebugHost.exe**. You can force this pause by adding a breakpoint to the **OnPreExecute** event, or by adding a Script task to your project and entering script that displays a modal message box.
2. Run the package. When the pause occurs, switch to the instance of Visual Studio in which your code project is open, and select **Attach to Process** from the **Debug** menu. Make sure to attach to the instance of **DtsDebugHost.exe** listed as **Managed, x86** in the **Type** column, not to the instance listed as **x86** only.
3. Return to the paused package and continue past the breakpoint, or click **OK** to dismiss the message box raised by the Script task, and continue package execution and debugging.

## See Also

[Developing Custom Objects for Integration Services](#)

[Persisting Custom Objects](#)

[Troubleshooting Tools for Package Development](#)

# Adding Connections Programmatically

4/25/2017 • 4 min to read • [Edit Online](#)

The [ConnectionManager](#) class represents physical connections to external data sources. The [ConnectionManager](#) class isolates the implementation details of the connection from the runtime. This enables the runtime to interact with each connection manager in a consistent and predictable manner. Connection managers contain a set of stock properties that all connections have in common, such as the [Name](#), [ID](#), [Description](#), and [ConnectionString](#). However, the [ConnectionString](#) and [Name](#) properties are ordinarily the only properties required to configure a connection manager. Unlike other programming paradigms, where connection classes expose methods such as **Open** or **Connect** to physically establish a connection to the data source, the run-time engine manages all the connections for the package while it runs.

The [Connections](#) class is a collection of the connection managers that have been added to that package and are available for use at run time. You can add more connection managers to the collection by using the [Add](#) method of the collection, and supplying a string that indicates the connection manager type. The [Add](#) method returns the [ConnectionManager](#) instance that was added to the package.

## Intrinsic Properties

The [ConnectionManager](#) class exposes a set of properties that are common to all connections. However, sometimes you need access to properties that are unique to the specific connection type. The [Properties](#) collection of the [ConnectionManager](#) class provides access to these properties. The properties can be retrieved from the collection using the indexer or the property name and the **GetValue** method, and the values are set using the **SetValue** method. The properties of the underlying connection object properties can also be set by acquiring an actual instance of the object and setting its properties directly. To get the underlying connection, use the [InnerObject](#) property of the connection manager. The following line of code shows a C# line that creates an ADO.NET connection manager that has the underlying class, [ConnectionManagerAdoNetClass](#).

```
ConnectionManagerAdoNetClass cmado = cm.InnerObject as ConnectionManagerAdoNet;
```

This casts the managed connection manager object to its underlying connection object. If you are using C++, the **QueryInterface** method of the [ConnectionManager](#) object is called and the interface of the underlying connection object is requested.

The following table lists the connection managers included with Integration Services, and the string that is used in the `package.Connections.Add("xxx")` statement. For a list of all connection managers, see [Integration Services \(SSIS\) Connections](#).

STRING	CONNECTION MANAGER
"OLEDB"	Connection manager for OLE DB connections.
"ODBC"	Connection manager for ODBC connections.
"ADO"	Connection manager for ADO connections.
"ADO.NET:SQL"	Connection manager for ADO.NET (SQL data provider) connections.

STRING	CONNECTION MANAGER
"ADO.NET:OLEDB"	Connection manager for ADO.NET (OLE DB data provider) connections.
"FLATFILE"	Connection manager for flat file connections.
"FILE"	Connection manager for file connections.
"MULTIFLATFILE"	Connection manager for multiple flat file connections.
"MULTIFILE"	Connection manager for multiple file connections.
"SQLMOBILE"	Connection manager for SQL Server Compact connections.
"MSOLAP100"	Connection manager for Analysis Services connections.
"FTP"	Connection manager for FTP connections.
"HTTP"	Connection manager for HTTP connections.
"MSMQ"	Connection manager for Message Queuing (also known as MSMQ) connections.
"SMTP"	Connection manager for SMTP connections.
"WMI"	Connection manager for Microsoft Windows Management Instrumentation (WMI) connections.

The following code example demonstrates adding an OLE DB and FILE connection to the [Connections](#) collection of a [Package](#). The example then sets the [ConnectionString](#), [Name](#), and [Description](#) properties.

```

using System;
using Microsoft.SqlServer.Dts.Runtime;

namespace Microsoft.SqlServer.Dts.Samples
{
    class Program
    {
        static void Main(string[] args)
        {
            // Create a package, and retrieve its connections.
            Package pkg = new Package();
            Connections pkgConns = pkg.Connections;

            // Add an OLE DB connection to the package, using the
            // method defined in the AddConnection class.
            CreateConnection myOLEDBConn = new CreateConnection();
            myOLEDBConn.CreateOLEDBConnection(pkg);

            // View the new connection in the package.
            Console.WriteLine("Connection description: {0}",
                pkg.Connections["SSIS Connection Manager for OLE DB"].Description);

            // Add a second connection to the package.
            CreateConnection myFileConn = new CreateConnection();
            myFileConn.CreateFileConnection(pkg);

            // View the second connection in the package.
            Console.WriteLine("Connection description: {0}",
                pkg.Connections["SSIS Connection Manager for Files"].Description);

            Console.WriteLine();
            Console.WriteLine("Number of connections in package: {0}", pkg.Connections.Count);

            Console.Read();
        }
    }
    // <summary>
    // This class contains the definitions for multiple
    // connection managers.
    // </summary>
    public class CreateConnection
    {
        // Private data.
        private ConnectionManager ConMgr;

        // Class definition for OLE DB Provider.
        public void CreateOLEDBConnection(Package p)
        {
            ConMgr = p.Connections.Add("OLEDB");
            ConMgr.ConnectionString = "Provider=SQLOLEDB.1;" +
                "Integrated Security=SSPI;Initial Catalog=AdventureWorks;" +
                "Data Source=(local);";
            ConMgr.Name = "SSIS Connection Manager for OLE DB";
            ConMgr.Description = "OLE DB connection to the AdventureWorks database.";
        }
        public void CreateFileConnection(Package p)
        {
            ConMgr = p.Connections.Add("File");
            ConMgr.ConnectionString = @"\\<yourserver>\<yourfolder>\books.xml";
            ConMgr.Name = "SSIS Connection Manager for Files";
            ConMgr.Description = "Flat File connection";
        }
    }
}

```



```
Imports Microsoft.SqlServer.Dts.Runtime

Module Module1

    Sub Main()

        ' Create a package, and retrieve its connections.
        Dim pkg As New Package()
        Dim pkgConns As Connections = pkg.Connections

        ' Add an OLE DB connection to the package, using the
        ' method defined in the AddConnection class.
        Dim myOLEDBConn As New CreateConnection()
        myOLEDBConn.CreateOLEDBConnection(pkg)

        ' View the new connection in the package.
        Console.WriteLine("Connection description: {0}", _
            pkg.Connections("SSIS Connection Manager for OLE DB").Description)

        ' Add a second connection to the package.
        Dim myFileConn As New CreateConnection()
        myFileConn.CreateFileConnection(pkg)

        ' View the second connection in the package.
        Console.WriteLine("Connection description: {0}", _
            pkg.Connections("SSIS Connection Manager for Files").Description)

        Console.WriteLine()
        Console.WriteLine("Number of connections in package: {0}", pkg.Connections.Count)

        Console.Read()

    End Sub

End Module

' This class contains the definitions for multiple
' connection managers.

Public Class CreateConnection
    ' Private data.
    Private ConMgr As ConnectionManager

    ' Class definition for OLE DB provider.
    Public Sub CreateOLEDBConnection(ByVal p As Package)
        ConMgr = p.Connections.Add("OLEDB")
        ConMgr.ConnectionString = "Provider=SQLOLEDB.1;" & _
            "Integrated Security=SSPI;Initial Catalog=AdventureWorks;" & _
            "Data Source=(local);"
        ConMgr.Name = "SSIS Connection Manager for OLE DB"
        ConMgr.Description = "OLE DB connection to the AdventureWorks database."
    End Sub

    Public Sub CreateFileConnection(ByVal p As Package)
        ConMgr = p.Connections.Add("File")
        ConMgr.ConnectionString = "\\<yourserver>\<yourfolder>\books.xml"
        ConMgr.Name = "SSIS Connection Manager for Files"
        ConMgr.Description = "Flat File connection"
    End Sub

End Class
```

### Sample Output:

```
Connection description: OLE DB connection to the AdventureWorks database.
```

Connection description: OLE DB connection to the AdventureWorks database.

Number of connections in package: 2

## External Resources

Technical article, [Connection Strings](#), on carlprothman.net.

## See Also

[Integration Services \(SSIS\) Connections](#)

[Create Connection Managers](#)

# Enumerating Available Packages Programmatically

3/24/2017 • 2 min to read • [Edit Online](#)

As you work programmatically with Integration Services packages, you may want to determine whether an individual package or folder exists, or to enumerate the saved packages that are available to load and execute. The [Application](#) class of the [Microsoft.SqlServer.Dts.Runtime](#) namespace provides a variety of methods to satisfy these requirements.

## Determining Whether a Package or Folder Exists

To determine programmatically whether a saved package exists, call one of the following methods before attempting to load and run it:

STORAGE LOCATION	METHOD TO CALL
SSIS Package Store	<a href="#">ExistsOnDtsServer</a>
SQL Server	<a href="#">ExistsOnSqlServer</a>

To determine programmatically whether a folder exists before attempting to list the packages stored in it, call one of the following methods:

STORAGE LOCATION	METHOD TO CALL
SSIS Package Store	<a href="#">FolderExistsOnDtsServer</a>
SQL Server	<a href="#">FolderExistsOnSqlServer</a>

[Back to top](#)

## Enumerating Available Packages

To obtain a list of saved packages programmatically, call one of the following methods:

STORAGE LOCATION	METHOD TO CALL
SSIS Package Store	<a href="#">GetDtsServerPackageInfos</a>
SQL Server	<a href="#">GetPackageInfos</a>

The following samples are console applications that demonstrate the use of these methods.

### Example (SSIS Package Store)

Use the [GetDtsServerPackageInfos](#) method to list packages stored in the SSIS Package Store. The default storage locations that are managed by the SSIS Package store are File System and MSDB. You can create additional logical folders within these locations.

```
Imports Microsoft.SqlServer.Dts.Runtime

Module Module1

    Sub Main()

        Dim sqlFolder As String
        Dim sqlServer As String

        Dim ssisApplication As Application
        Dim sqlPackages As PackageInfos
        Dim sqlPackage As PackageInfo

        sqlServer = "."

        ssisApplication = New Application()

        ' Get packages stored in MSDB.
        sqlFolder = "MSDB"
        sqlPackages = ssisApplication.GetDtsServerPackageInfos(sqlFolder, sqlServer)
        If sqlPackages.Count > 0 Then
            Console.WriteLine("Packages stored in MSDB:")
            For Each sqlPackage In sqlPackages
                Console.WriteLine(sqlPackage.Name)
            Next
            Console.WriteLine()
        End If

        ' Get packages stored in the File System.
        sqlFolder = "File System"
        sqlPackages = ssisApplication.GetDtsServerPackageInfos(sqlFolder, sqlServer)
        If sqlPackages.Count > 0 Then
            Console.WriteLine("Packages stored in the File System:")
            For Each sqlPackage In sqlPackages
                Console.WriteLine(sqlPackage.Name)
            Next
        End If

        Console.Read()

    End Sub

End Module
```

```

using System;
using Microsoft.SqlServer.Dts.Runtime;

namespace EnumeratePackagesSSIS_CS
{
    class Program
    {
        static void Main(string[] args)
        {

            string sqlFolder;
            string sqlServer;

            Application ssisApplication;
            PackageInfos sqlPackages;

            sqlServer = ".";

            ssisApplication = new Application();

            // Get packages stored in MSDB.
            sqlFolder = "MSDB";
            sqlPackages = ssisApplication.GetDtsServerPackageInfos(sqlFolder, sqlServer);
            if (sqlPackages.Count > 0)
            {
                Console.WriteLine("Packages stored in MSDB:");
                foreach (PackageInfo sqlPackage in sqlPackages)
                {
                    Console.WriteLine(sqlPackage.Name);
                }
                Console.WriteLine();
            }

            // Get packages stored in the File System.
            sqlFolder = "File System";
            sqlPackages = ssisApplication.GetDtsServerPackageInfos(sqlFolder, sqlServer);
            if (sqlPackages.Count > 0)
            {
                Console.WriteLine("Packages stored in the File System:");
                foreach (PackageInfo sqlPackage in sqlPackages)
                {
                    Console.WriteLine(sqlPackage.Name);
                }
            }

            Console.Read();

        }
    }
}

```

[Back to top](#)

### Example (SQL Server)

Use the [GetPackageInfos](#) method to list Integration Services packages that are stored in an instance of SQL Server.

```
Imports Microsoft.SqlServer.Dts.Runtime
```

```
Module Module1
```

```
    Sub Main()
```

```
        Dim sqlFolder As String
        Dim sqlServer As String
        Dim sqlUser As String
        Dim sqlPassword As String
```

```
        Dim ssisApplication As Application
        Dim sqlPackages As PackageInfos
        Dim sqlPackage As PackageInfo
```

```
        sqlFolder = String.Empty
        sqlServer = "(local)"
        sqlUser = String.Empty
        sqlPassword = String.Empty
```

```
        ssisApplication = New Application()
```

```
        sqlPackages = ssisApplication.GetPackageInfos(sqlFolder, sqlServer, sqlUser, sqlPassword)
```

```
        For Each sqlPackage In sqlPackages
            Console.WriteLine(sqlPackage.Name)
        Next
```

```
        Console.Read()
```

```
    End Sub
```

```
End Module
```

```

using System;
using Microsoft.SqlServer.Dts.Runtime;

namespace EnumeratePackagesSql_CS
{
    class Program
    {
        static void Main(string[] args)
        {

            string sqlFolder;
            string sqlServer;
            string sqlUser;
            string sqlPassword;

            Application ssisApplication;
            PackageInfos sqlPackages;

            sqlFolder = String.Empty;
            sqlServer = "(local)";
            sqlUser = String.Empty;
            sqlPassword = String.Empty;

            ssisApplication = new Application();

            sqlPackages = ssisApplication.GetPackageInfos(sqlFolder, sqlServer, sqlUser, sqlPassword);

            foreach (PackageInfo sqlPackage in sqlPackages)
            {
                Console.WriteLine(sqlPackage.Name);
            }

            Console.Read();

        }
    }
}

```

[Back to top](#)

## See Also

[Package Management \(SSIS Service\)](#)

# Integration Services Language Reference

3/24/2017 • 1 min to read • [Edit Online](#)

**THIS TOPIC APPLIES TO:**  SQL Server (starting with 2008)  Azure SQL Database  Azure SQL Data Warehouse  Parallel Data Warehouse

This section describes the Transact-SQL API for administering Integration Services projects that have been deployed to an instance of SQL Server.

Integration Services stores objects, settings, and operational data in a database referred to as the Integration Services catalog. The default name of the Integration Services catalog is SSISDB. The objects that are stored in the catalog include projects, packages, parameters, environments, and operational history.

The Integration Services catalog stores its data in internal tables that are not visible to users. However it exposes the information that you need through a set of public views that you can query. It also provides a set of stored procedures that you can use to perform common tasks on the catalog.

Typically you manage Integration Services objects in the catalog by opening SQL Server Management Studio. However you can also use the database views and stored procedures directly, or write custom code that calls the managed API. Management Studio and the managed API query the views and call the stored procedures that are described in this section to perform many of their tasks.

## In This Section

### [Views \(Integration Services Catalog\)](#)

Query the views to inspect Integration Services objects, settings, and operational data.

### [Stored Procedures \(Integration Services Catalog\)](#)

Call the stored procedures to add, remove, or modify Integration Services objects and settings.

### [Functions \(Integration Services Catalog\)](#)

Call the functions to administer Integration Services projects.



# Azure Feature Pack for Integration Services (SSIS)

3/24/2017 • 1 min to read • [Edit Online](#)

SQL Server Integration Services (SSIS) Feature Pack for Azure is an extension that provides the components listed on this page for SSIS to connect to Azure services, transfer data between Azure and on-premises data sources, and process data stored in Azure.



## Download

- [SSIS Feature Pack for Azure](#) for SQL Server 2016
- [SSIS Feature Pack for Azure](#) for SQL Server 2017

## Components in the Feature Pack

- Connection Managers
  - [Azure Storage Connection Manager](#)
  - [Azure Subscription Connection Manager](#)
  - [Azure Data Lake Store Connection Manager](#)
  - [Azure Resource Manager Connection Manager](#)
  - [Azure HDInsight Connection Manager](#)
- Tasks
  - [Azure Blob Upload Task](#)
  - [Azure Blob Download Task](#)
  - [Azure HDInsight Hive Task](#)
  - [Azure HDInsight Pig Task](#)
  - [Azure HDInsight Create Cluster Task](#)
  - [Azure HDInsight Delete Cluster Task](#)
  - [Azure SQL DW Upload Task](#)
- Data Flow Components
  - [Azure Blob Source](#)
  - [Azure Blob Destination](#)
  - [Azure Data Lake Store Source](#)
  - [Azure Data Lake Store Destination](#)
- Azure Blob Enumerator. See [Enumerator = Foreach Azure Blob Enumerator](#)

## Download the Feature Pack

Download the SQL Server Integration Services (SSIS) Feature Pack for Azure.

- [SSIS Feature Pack for Azure](#) for SQL Server 2016
- [SSIS Feature Pack for Azure](#) for SQL Server 2017

## Prerequisites

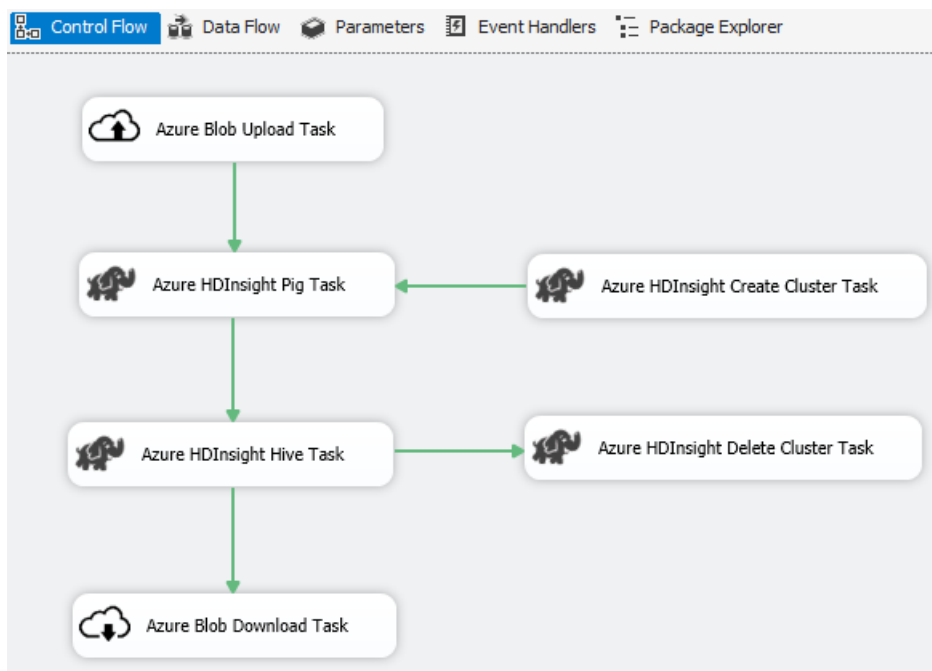
You must install the following prerequisites before installing this feature pack.

- SQL Server Integration Services
- .Net Framework 4.5

## Scenario: Processing big data

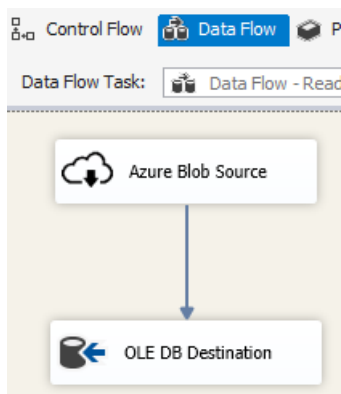
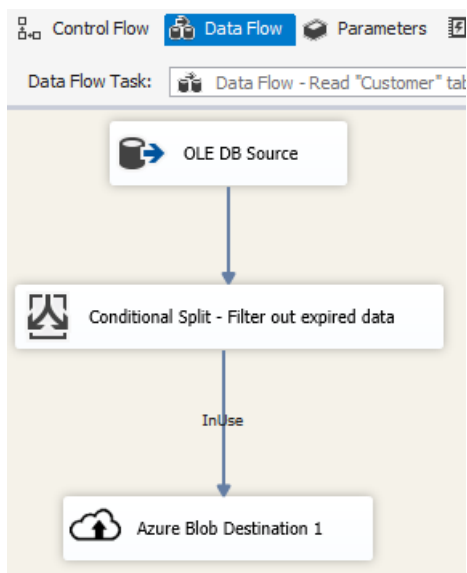
Use Azure Connector to complete following big data processing work:

1. Use the Azure Blob Upload Task to upload input data to Azure Blob Storage.
2. Use the Azure HDInsight Create Cluster Task to create an Azure HDInsight cluster. This step is optional if you want to use your own cluster.
3. Use the Azure HDInsight Hive Task or Azure HDInsight Pig Task to invoke a Pig or Hive job on the Azure HDInsight cluster.
4. Use the Azure HDInsight Delete Cluster Task to delete the HDInsight Cluster after use if you have created an on-demand HDInsight cluster in step #2.
5. Use the Azure HDInsight Blob Download Task to download the Pig/Hive output data from the Azure Blob Storage.

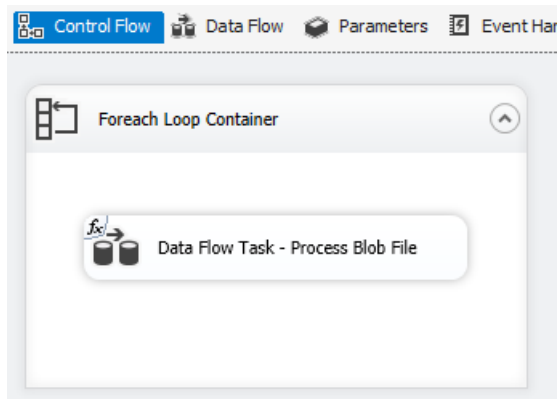


## Scenario: Managing data in the cloud

Use the Azure Blob Destination in an SSIS package to write output data to Azure Blob Storage, or use the Azure Blob Source to read data from an Azure Blob Storage.



Use the Foreach Loop Container with the Azure Blob Enumerator to process data in multiple blob files.



# Hadoop and HDFS Support in Integration Services (SSIS)

3/24/2017 • 1 min to read • [Edit Online](#)

SQL Server 2016 Integration Services (SSIS) includes the following components that provide support for Hadoop and HDFS on premises.

For info about the Integration Services components that support HDInsight and other features of Microsoft Azure, see [Azure Feature Pack for Integration Services \(SSIS\)](#).

- Connection manager
  - [Hadoop Connection Manager](#)
- Control flow - Tasks
  - [Hadoop File System Task](#)
  - [Hadoop Hive Task](#)
  - [Hadoop Pig Task](#)
- Data flow - Data source and destination
  - [HDFS File Source](#)
  - [HDFS File Destination](#)

# Microsoft Connectors for Oracle and Teradata by Attunity for Integration Services (SSIS)

5/19/2017 • 1 min to read • [Edit Online](#)

You can download connectors for Integration Services by Attunity that optimize performance when loading data to or from Oracle or Teradata in an SSIS package.

## Download the latest Attunity connectors

Get the latest version of the connectors here:

[Microsoft Connectors v5.0 for Oracle and Teradata](#)

## Issue - The Attunity connectors aren't visible in the SSIS Toolbox

To see the Attunity connectors in the SSIS Toolbox, you always have to install the version of the connectors that targets the same version of SQL Server as the version of SQL Server Data Tools (SSDT) installed on your computer. (You may also have earlier versions of the connectors installed.) This requirement is independent of the version of SQL Server that you want to target in your SSIS projects and packages.

For example, if you've installed the latest version of SSDT, you have version 17 of SSDT with a build number that starts with 14. This version of SSDT adds support for SQL Server 2017. To see and use the Attunity connectors in SSIS package development - even if you want to target an earlier version of SQL Server - you also have to install the latest version of the Attunity connectors, version 5.0. This version of the connectors also adds support for SQL Server 2017.

Check the installed version of SSDT in Visual Studio from **Help | About Microsoft Visual Studio**, or in **Programs and Features** in the Control Panel. Then install the corresponding version of the Attunity connectors from the following table.

SSDT VERSION	SSDT BUILD NUMBER	TARGET SQL SERVER VERSION	REQUIRED VERSION OF CONNECTORS
17	Starts with 14	SQL Server 2017	<a href="#">Microsoft Connectors v5.0 for Oracle and Teradata</a>
16	Starts with 13	SQL Server 2016	<a href="#">Microsoft Connectors v4.0 for Oracle and Teradata</a>

## Download the latest SQL Server Data Tools (SSDT)

Get the latest version of SSDT here:

[Download SQL Server Data Tools \(SSDT\)](#)

# Import and Export Data with the SQL Server Import and Export Wizard

5/11/2017 • 5 min to read • [Edit Online](#)

For content related to previous versions of SQL Server, see [Run the SQL Server Import and Export Wizard](#).

SQL Server Import and Export Wizard is a simple way to copy data from a source to a destination. This overview describes the data sources that the wizard can use as sources and destinations, as well as the permissions you need to run the wizard.

## Get the wizard

If you want to run the wizard, but you don't have Microsoft SQL Server installed on your computer, you can install the SQL Server Import and Export Wizard by installing SQL Server Data Tools (SSDT). For more info, see [Download SQL Server Data Tools \(SSDT\)](#).

## What happens when I run the wizard?

- **See the list of steps.** For a description of the steps in the wizard, see [Steps in the SQL Server Import and Export Wizard](#). There's also a separate page of documentation for each page of the wizard.  
- or -
- **See a simple example.** For a quick look at the several screens you see in a typical session, take a look at this simple end-to-end example on a single page - [Get started with this simple example of the Import and Export Wizard](#).

## What sources and destinations can I use?

The SQL Server Import and Export Wizard can copy data to and from the data sources listed in the following table. To connect to some of these data sources, you may have to download and install additional files.

DATA SOURCE	DO I HAVE TO DOWNLOAD ADDITIONAL FILES?
<b>Enterprise databases</b> SQL Server, Oracle, DB2, and others.	<p>SQL Server or SQL Server Data Tools (SSDT) installs the files that you need to connect to SQL Server. But SSDT doesn't install all the files that you need to connect to other enterprise databases such as Oracle or IBM DB2.</p> <p>To connect to an enterprise database, you typically have to have two things:</p> <ol style="list-style-type: none"><li>1. <b>Client software.</b> If you already have the client software installed for your enterprise database system, then you typically have what you need to make a connection. If you don't have the client software installed, ask the database administrator how to install a licensed copy.</li><li>2. <b>Drivers or providers.</b> Microsoft installs drivers and providers to connect to Oracle. To connect to IBM DB2, get the Microsoft® OLEDB Provider for DB2 v5.0 for Microsoft SQL Server from the <a href="#">Microsoft SQL Server 2016 Feature Pack</a>.</li></ol>

DATA SOURCE	DO I HAVE TO DOWNLOAD ADDITIONAL FILES?
<b>Text files</b> (flat files)	No additional files required.
<b>Microsoft Excel and Microsoft Access files</b>	<p>Microsoft Office doesn't install all the files that you need to connect to Excel and Access files as data sources. Get the following download - <a href="#">Microsoft Access 2016 Runtime</a>.</p> <p>For more info, see <a href="#">Connect to an Excel Data Source</a> or <a href="#">Connect to an Access Data Source</a>.</p>
<b>Azure data sources</b> Currently only Azure Blob Storage.	<p>SQL Server Data Tools don't install the files that you need to connect to Azure Blob Storage as a data source. Get the following download - <a href="#">Microsoft SQL Server 2016 Integration Services Feature Pack for Azure</a>.</p> <p>For more info, see <a href="#">Connect to Azure Blob Storage</a>.</p>
<b>Open source databases</b> PostgreSQL, MySql, and others.	<p>You have to download additional files to connect to these data sources.</p> <ul style="list-style-type: none"> <li>- For <b>PostgreSQL</b>, see <a href="#">Connect to a PostgreSQL Data Source</a>.</li> <li>- For <b>MySql</b>, see <a href="#">Connect to a MySQL Data Source</a>.</li> </ul>
<b>Any other data source for which a driver or provider is available</b>	<p>You typically have to download additional files to connect to the following types of data sources.</p> <ul style="list-style-type: none"> <li>- Any source for which an <b>ODBC driver</b> is available. For more info, see <a href="#">Connect to an ODBC Data Source</a>.</li> <li>- Any source for which a <b>.Net Framework Data Provider</b> is available.</li> <li>- Any source for which an <b>OLE DB Provider</b> is available.</li> </ul> <p>Third-party components that provide source and destination capabilities for other data sources are sometimes marketed as add-on products for SQL Server Integration Services (SSIS).</p>

## How do I connect to my data?

For info about how to connect to a commonly used data source, see one of the following pages.

- [Connect to SQL Server](#)
- [Connect to Oracle](#)
- [Connect to flat files \(text files\)](#)
- [Connect to Excel](#)
- [Connect to Access](#)
- [Connect to Azure Blob Storage](#)
- [Connect with ODBC](#)
- [Connect to PostgreSQL](#)
- [Connect to MySQL](#)

For info about how to connect to a data source that's not listed here, see [The Connection Strings Reference](#). This third-party site contains sample connection strings and more info about data providers and the connection info they require.

## What permissions do I need?

To run the SQL Server Import and Export Wizard successfully, you have to have at least the following permissions. If you already work with your data source and destination, you probably already have the permissions that you need.

YOU NEED PERMISSIONS TO DO THESE THINGS	IF YOU'RE CONNECTING TO SQL SERVER, YOU NEED THESE SPECIFIC PERMISSIONS
Connect to the source and destination databases or file shares.	Server and database login rights.
Export or read data from the source database or file.	SELECT permissions on the source tables and views.
Import or write data to the destination database or file.	INSERT permissions on the destination tables.
Create the destination database or file, if applicable.	CREATE DATABASE or CREATE TABLE permissions.
Save the SSIS package created by the wizard, if applicable.	If you want to save the package to SQL Server, permissions sufficient to save the package to the <b>msdb</b> database.

## Get help while the wizard is running

### TIP

Tap the F1 key from any page or dialog box of the wizard to see documentation for the current page.

## The wizard uses SQL Server Integration Services (SSIS)

The wizard uses SQL Server Integration Services (SSIS) to copy data. SSIS is a tool for extracting, transforming, and loading data (ETL). The pages of the wizard use some of the language of SSIS.

In SSIS, the basic unit is the **package**. The wizard creates an SSIS package in memory as you move through the pages of the wizard and specify options.

At the end of the wizard, if you have SQL Server Standard Edition or higher installed, you can optionally save the SSIS package. Later you can reuse the package and extend it by using SSIS Designer to add tasks, transformations, and event-driven logic. The SQL Server Import and Export Wizard is the simplest way to create a basic Integration Services package that copies data from a source to a destination.

For more info about SSIS, see [SQL Server Integration Services](#).

## What's next?

Start the wizard. For more info, see [Start the SQL Server Import and Export Wizard](#).

## See also

[Get started with this simple example of the Import and Export Wizard](#)  
[Data Type Mapping in the SQL Server Import and Export Wizard](#)



# Change Data Capture (SSIS)

3/24/2017 • 5 min to read • [Edit Online](#)

In SQL Server, change data capture offers an effective solution to the challenge of efficiently performing incremental loads from source tables to data marts and data warehouses.

## What is Change Data Capture?

Source tables change over time. A data mart or data warehouse that is based on those tables needs to reflect these changes. However, a process that periodically copies a snapshot of the entire source consumes too much time and resources. Alternate approaches that include timestamp columns, triggers, or complex queries often hurt performance and increase complexity. What is needed is a reliable stream of change data that is structured so that it can easily be applied by consumers to target representations of the data. Change data capture in SQL Server provides this solution.

The change data capture feature of the Database Engine captures insert, update, and delete activity applied to SQL Server tables, and makes the details of the changes available in an easily-consumed, relational format. The change tables used by change data capture contain columns that mirror the column structure of the tracked source tables, along with the metadata needed to understand the changes that have occurred on a row by row basis.

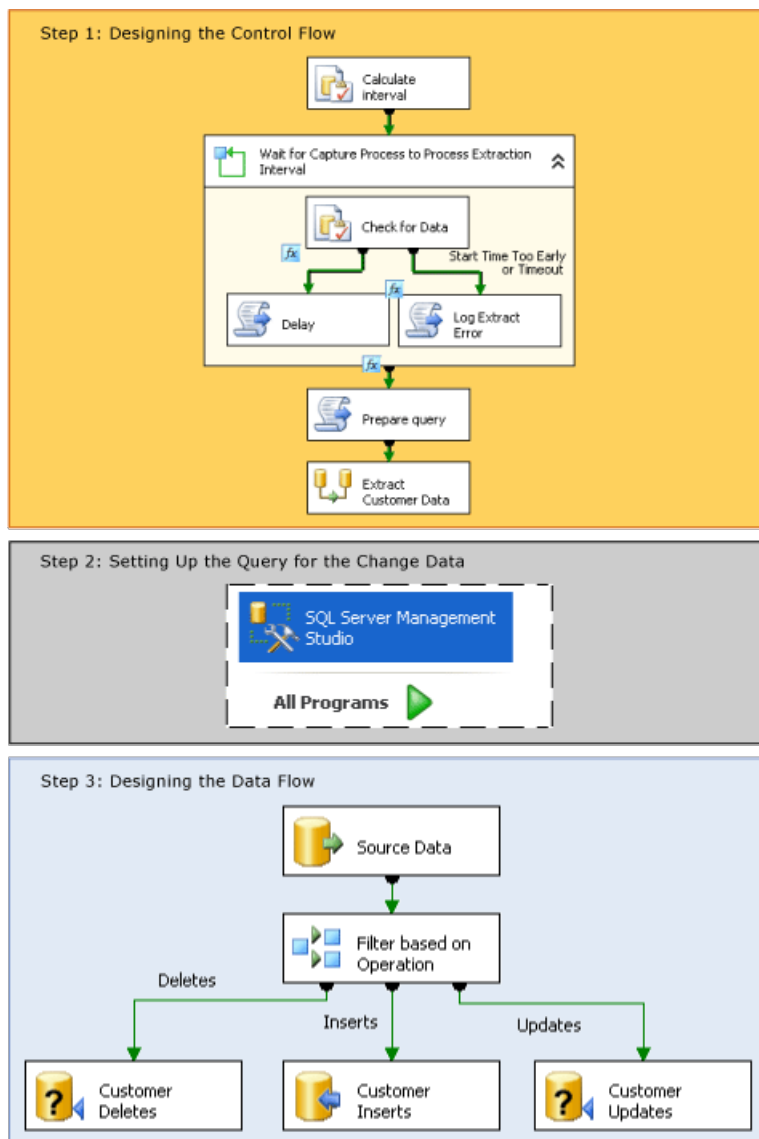
### NOTE

Change data capture is not available in every edition of Microsoft SQL Server. For a list of features that are supported by the editions of SQL Server, see [Features Supported by the Editions of SQL Server 2016](#).

## How Change Data Capture Works in Integration Services

An Integration Services package can easily harvest the change data in the SQL Server databases to perform efficient incremental loads to a data warehouse. However, before you can use Integration Services to load change data, an administrator must enable change data capture on the database and the tables from which you want to capture changes. For more information on how to configure change data capture on a database, see [Enable and Disable Change Data Capture \(SQL Server\)](#).

Once an administrator has enabled change data capture on the database, you can create a package that performs an incremental load of the change data. The following diagram shows the steps for creating such a package that performs an incremental load from a single table:



As shown in the previous diagram, creating a package that performs an incremental load of changed data involves the following steps:

### Step 1: Designing the Control Flow

In the control flow in the package, the following tasks need to be defined:

- Calculate the starting and ending **datetime** values for the interval of changes to the source data that you want to retrieve.

To calculate these values, use an Execute SQL task or Integration Services expressions with **datetime** functions. You then store these endpoints in package variables for use later in the package.

**For more information:** [Specify an Interval of Change Data](#)

- Determine whether the change data for the selected interval is ready. This step is necessary because the asynchronous capture process might not yet have reached the selected endpoint.

To determine whether the data is ready, start with a For Loop container to delay execution, if necessary, until the change data for the selected interval is ready. Inside the loop container, use an Execute SQL task to query the time mapping tables maintained by change data capture. Then, use a Script task that calls the **Thread.Sleep** method, or another Execute SQL task with a **WAITFOR** statement, to delay the execution of the package temporarily, if necessary. Optionally, use another Script task to log an error condition or a timeout.

**For more information:** [Determine Whether the Change Data Is Ready](#)

- Prepare the query string that will be used to query for the change data.

Use a Script task or an Execute SQL task to assemble the SQL statement that will be used to query for changes.

**For more information:** [Prepare to Query for the Change Data](#)

### Step 2: Setting Up the Query for Change Data

Create the table-valued function that will query for the data.

Use SQL Server Management Studio to develop and save the query.

**For more information:** [Retrieve and Understand the Change Data](#)

### Step 3: Designing the Data Flow

In the data flow of the package, the following tasks need to be defined:

- Retrieve the change data from the change tables.

To retrieve the data, use a source component to query the change tables for the changes that fall within the selected interval. The source calls a Transact-SQL table-valued function that you must have previously created.

**For more information:** [Retrieve and Understand the Change Data](#)

- Split the changes into inserts, updates, and deletes for processing.

To split the changes, use a Conditional Split transformation to direct inserts, updates, and deletes to different outputs for appropriate processing.

**For more information:** [Process Inserts, Updates, and Deletes](#)

- Apply the inserts, deletes, and updates to the destination.

To apply the changes to the destination, use a destination component to apply the inserts to the destination. Also, use OLE DB Command transformations with parameterized UPDATE and DELETE statements to apply updates and deletes to the destination. You can also apply updates and deletes by using destination components to save the rows to temporary tables. Then, use Execute SQL tasks to perform bulk update and bulk delete operations against the destination from the temporary tables.

**For more information:** [Apply the Changes to the Destination](#)

### Change Data from Multiple Tables

The process outlined in the previous diagram and steps involves an incremental load from a single table. When having to perform an incremental load from multiple tables, the overall process is the same. However, the design of the package needs to be changed to accommodate the processing of multiple tables. For more information on how to create a package that performs an incremental load from multiple tables, see [Perform an Incremental Load of Multiple Tables](#).

## Samples of Change Data Capture Packages

Integration Services provides two samples that demonstrate how to use change data capture in packages. For more information, see the following topics:

- [Readme\\_Change Data Capture for Specified Interval Package Sample](#)
- [Readme\\_Change Data Capture since Last Request Package Sample](#)

## Related Tasks

- [Specify an Interval of Change Data](#)
- [Determine Whether the Change Data Is Ready](#)
- [Prepare to Query for the Change Data](#)
- [Create the Function to Retrieve the Change Data](#)
- [Retrieve and Understand the Change Data](#)
- [Process Inserts, Updates, and Deletes](#)
- [Apply the Changes to the Destination](#)
- [Perform an Incremental Load of Multiple Tables](#)

## Related Content

Blog entry, [SSIS Design Pattern – Incremental Load](#), on [sqlblog.com](#)

# Microsoft Connector for SAP BW

3/24/2017 • 1 min to read • [Edit Online](#)

The Microsoft Connector for SAP BW consists of a set of three components that let you extract data from, or load data into, an SAP Netweaver BW version 7 system.

The Microsoft Connector for SAP BW for SQL Server 2016 is a component of the SQL Server 2016 Feature Pack. To install the Connector for SAP BW and its documentation, download and run the installer from the [SQL Server 2016 Feature Pack web page](#).

## IMPORTANT

The documentation for the Microsoft Connector for SAP BW assumes familiarity with the SAP Netweaver BW environment. For more information about SAP Netweaver BW, or for information about how to configure SAP Netweaver BW objects and processes, see your SAP documentation.

## IMPORTANT

Extracting data from SAP Netweaver BW requires additional SAP licensing. Check with SAP to verify these requirements.

## Components

The Microsoft Connector for SAP BW has the following components:

- **SAP BW Source**—The SAP BW source is a data flow source component that lets you extract data from an SAP Netweaver BW version 7 system.
- **SAP BW Destination**—The SAP BW destination is a data flow destination component that lets you load data into an SAP Netweaver BW version 7 system.
- **SAP BW Connection Manager**—The SAP BW connection manager connects either an SAP BW source or SAP BW destination to an SAP Netweaver BW version 7 system.

For a walkthrough that demonstrates how to configure and use the SAP BW connection manager, source, and destination, see the white paper, [Using SQL Server Integration Services with SAP BI 7.0](#). This white paper also shows how to configure the required objects in SAP BW.

## Documentation

This Help file for the Microsoft Connector for SAP BW contains the following topics and sections:

### [Installing the Microsoft Connector for SAP BW](#)

Describes the installation requirements for the Microsoft Connector for SAP BW.

### [Microsoft Connector for SAP BW Components](#)

Describes each component of the Microsoft Connector for SAP BW.

### [Microsoft Connector for SAP BW F1 Help](#)

Describes the user interface of each component of the Microsoft Connector for SAP BW.

# Installing the Microsoft Connector for SAP BW

3/24/2017 • 1 min to read • [Edit Online](#)

The Microsoft Connector for SAP BW for SQL Server 2016 is a component of the SQL Server 2016 Feature Pack. To install the Connector for SAP BW and its documentation, download and run the installer from the [SQL Server 2016 Feature Pack web page](#).

## IMPORTANT

The documentation for the Microsoft Connector for SAP BW assumes familiarity with the SAP Netweaver BW environment. For more information about SAP Netweaver BW, or for information about how to configure SAP Netweaver BW objects and processes, see your SAP documentation.

## IMPORTANT

Extracting data from SAP Netweaver BW requires additional SAP licensing. Check with SAP to verify these requirements.

## Required SAP Files

To use the Microsoft Connector for SAP BW, you do not have to install the SAP Front End software (SAP GUI) on the local computer.

However you must copy the SAP .NET connector file, `librfc32.dll`, into the system subfolder in the Windows folder. (Typically, this folder location is **C:\Windows\system32**.)

## Considerations for 64-bit Computers

The Microsoft Connector for SAP BW fully supports the 64-bit version of Microsoft Windows. On a 64-bit computer, the Microsoft Connector for SAP BW has the following additional requirements:

- To run packages in 64-bit mode on any 64-bit Windows operating system, copy the 64-bit version of the SAP GUI file, `librfc32.dll`, into the **system32** folder of the Windows folder. (Typically, this file location is **C:\Windows\system32**.)
- To run packages in 32-bit mode on any 64-bit Windows operating system, copy the SAP GUI file, `librfc32.dll`, into the **SysWow64** folder of the Windows folder. (Typically, this folder location is **C:\Windows\SysWow64**.)

# Microsoft Connector for SAP BW Components

3/24/2017 • 1 min to read • [Edit Online](#)

This section contains topics that describe the three components of the Microsoft Connector 1.1 for SAP BW:

- SAP BW connection manager
- SAP BW source
- SAP BW destination

## IMPORTANT

The documentation for the Microsoft Connector 1.1 for SAP BW assumes familiarity with the SAP Netweaver BW environment. For more information about SAP Netweaver BW, or for information about how to configure SAP Netweaver BW objects and processes, see your SAP documentation.

## In This Section

### [SAP BW Connection Manager](#)

Describes the SAP BW connection manager. The connection manager connects the SAP BW source or the SAP BW destination to an SAP Netweaver BW version 7 system.

### [SAP BW Source](#)

Describes the SAP BW source that lets you extract data from an SAP Netweaver BW system.

### [SAP BW Destination](#)

Describes the SAP BW destination that lets you load data into an SAP Netweaver BW system.

# Microsoft Connector for SAP BW F1 Help

3/24/2017 • 1 min to read • [Edit Online](#)

This section contains the F1 Help topics for the three components of the Microsoft Connector 1.1 for SAP BW. These topics are also available from the user interface by pressing the F1 key, or by clicking Help on wizard pages and dialog boxes.

## IMPORTANT

The documentation for the Microsoft Connector 1.1 for SAP BW assumes familiarity with the SAP Netweaver BW environment. For more information about SAP Netweaver BW, or for information about how to configure SAP Netweaver BW objects and processes, see your SAP documentation.

## In This Section

### SAP BW Connection Manager F1 Help

- [SAP BW Connection Manager Editor](#)

### SAP BW Source F1 Help

- [SAP BW Source Editor \(Connection Manager Page\)](#)
- [SAP BW Source Editor \(Columns Page\)](#)
- [SAP BW Source Editor \(Error Output Page\)](#)
- [SAP BW Source Editor \(Advanced Page\)](#)
- [Look Up RFC Destination](#)
- [Look Up Process Chain](#)
- [Request Log](#)
- [Preview](#)

### SAP BW Destination F1 Help

- [SAP BW Destination Editor \(Connection Manager Page\)](#)
- [SAP BW Destination Editor \(Mappings Page\)](#)
- [SAP BW Destination Editor \(Error Output Page\)](#)
- [SAP BW Destination Editor \(Advanced Page\)](#)
- [Look Up InfoPackage](#)
- [Create New InfoObject](#)
- [Create InfoCube for Transaction Data](#)
- [Look Up InfoObject](#)
- [Create InfoSource](#)
- [Create InfoSource for Transaction Data](#)



- [Create InfoSource for Master Data](#)
- [Create InfoPackage](#)

## See Also

[Microsoft Connector for SAP BW Components](#)

# Certification by SAP

3/24/2017 • 1 min to read • [Edit Online](#)

The Microsoft Connector 1.1 for SAP BW has received certification from SAP for integration with SAP NetWeaver.



The following table describes the details of the certification.

SAP INTERFACE	SAP RELEASE LEVELS	CERTIFICATION DATE	RELATED COMPONENT
BW_OHS 7.0 - SAP NetWeaver Business Intelligence - Open Hub Service 7.0	Business Intelligence 7.0	December 2012	Source
BW-STA 3.5 - Staging BAPIs for SAP BW 3.5	Business Intelligence 3.5, 7.0	December 2012	Destination

# Integration Services Tutorials

3/24/2017 • 1 min to read • [Edit Online](#)

This section contains tutorials Integration Services.

- [SSIS How to Create an ETL Package](#)
- [Deploy Packages with SSIS](#)

# SSIS How to Create an ETL Package

5/11/2017 • 3 min to read • [Edit Online](#)

For content related to previous versions of SQL Server, see [SSIS Tutorial: Creating a Simple ETL Package](#).

Microsoft SQL Server Integration Services (SSIS) is a platform for building high performance data integration solutions, including extraction, transformation, and load (ETL) packages for data warehousing. SSIS includes graphical tools and wizards for building and debugging packages; tasks for performing workflow functions such as FTP operations, executing SQL statements, and sending e-mail messages; 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 package execution and storage; and application programming interfaces (APIs) for programming the Integration Services object model.

In this tutorial, you will learn how to use SSIS Designer to create a simple Microsoft SQL Server Integration Services package. The package that you create takes data from a flat file, reformats the data, and then inserts the reformatted data into a fact table. In following lessons, the package is expanded to demonstrate looping, package configurations, logging and error flow.

When you install the sample data that the tutorial uses, you also install the completed versions of the packages that you will create in each lesson of the tutorial. By using the completed packages, you can skip ahead and begin the tutorial at a later lesson if you like. If this is your first time working with packages or the new development environment, we recommend that you begin with Lesson1.

## What You Will Learn

The best way to become acquainted with the new tools, controls and features available in Microsoft SQL Server Integration Services is to use them. This tutorial walks you through SSIS Designer to create a simple ETL package that includes looping, configurations, error flow logic and logging.

## Requirements

This tutorial is intended for users familiar with fundamental database operations, but who have limited exposure to the new features available in SQL Server Integration Services.

To use this tutorial, your system must have the following components installed:

- SQL Server with the **AdventureWorksDW2012** database. To enhance security, the sample databases are not installed by default. To download the **AdventureWorksDW2012** database, see [Adventure Works for SQL Server 2012](#).

### IMPORTANT

When you attach the database (\*.mdf file), SQL Server Management Studio will by default search for an .ldf file. You must manually remove the .ldf file before clicking OK in the **Attach Databases** dialog box.

For more information about attaching databases, see [Attach a Database](#).

- Sample data. The sample data is included with the SSIS lesson packages. To download the sample data and the lesson packages, do the following.

1. Navigate to [Integration Services Product Samples](#)

2. Click the **DOWNLOADS** tab.

3. Click the SQL2012.Integration\_Services.Create\_Simple\_ETL\_Tutorial.Sample.zip file.

## Lessons in This Tutorial

### [Lesson 1: Create a Project and Basic Package with SSIS](#)

In this lesson, you will create a simple ETL package that extracts data from a single flat file, transforms the data using lookup transformations and finally loads the result into a fact table destination.

### [Lesson 2: Adding Looping with SSIS](#)

In this lesson, you will expand the package you created in Lesson 1 to take advantage of new looping features to extract multiple flat files into a single data flow process.

### [Lesson 3: Add Logging with SSIS](#)

In this lesson, you will expand the package you created in Lesson 2 to take advantage of new logging features.

### [Lesson 4: Add Error Flow Redirection with SSIS](#)

In this lesson, you will expand the package you created in lesson 3 to take advantage of new error output configurations.

### [Lesson 5: Add SSIS Package Configurations for the Package Deployment Model](#)

In this lesson, you will expand the package you created in Lesson 4 to take advantage of new package configuration options.

### [Lesson 6: Using Parameters with the Project Deployment Model in SSIS](#)

In this lesson, you will expand the package you created in Lesson 5 to take advantage of using new parameters with the project deployment model.

# Lesson 1: Create a Project and Basic Package with SSIS

5/11/2017 • 3 min to read • [Edit Online](#)

For content related to previous versions of SQL Server, see [Lesson 1: Creating the Project and Basic Package](#).

In this lesson, you will create a simple ETL package that extracts data from a single flat file source, transforms the data using two lookup transformation components, and writes that data to the **FactCurrency** fact table in **AdventureWorksDW2012**. As part of this lesson, you will learn how to create new packages, add and configure data source and destination connections, and work with new control flow and data flow components.

## IMPORTANT

This tutorial requires the **AdventureWorksDW2012** sample database. For more information on installing and deploying **AdventureWorksDW2012**, see [Reporting Services Product Samples on CodePlex](#).

## Understanding the Package Requirements

This tutorial requires Microsoft SQL Server Data Tools.

For more information on installing the SQL Server Data Tools see [SQL Server Data Tools Download](#).

Before creating a package, you need a good understanding of the formatting used in both the source data and the destination. Once you understand both of these data formats, you will be ready to define the transformations necessary to map the source data to the destination.

### Looking at the Source

For this tutorial, the source data is a set of historical currency data contained in the flat file, SampleCurrencyData.txt. The source data has the following four columns: the average rate of the currency, a currency key, a date key, and the end-of-day rate.

Here is an example of the source data contained in the SampleCurrencyData.txt file:

```
1.00070049USD9/3/05 0:001.001201442
1.00020004USD9/4/05 0:001
1.00020004USD9/5/05 0:001.001201442
1.00020004USD9/6/05 0:001
1.00020004USD9/7/05 0:001.00070049
1.00070049USD9/8/05 0:000.99980004
1.00070049USD9/9/05 0:001.001502253
1.00070049USD9/10/05 0:000.99990001
1.00020004USD9/11/05 0:001.001101211
1.00020004USD9/12/05 0:000.99970009
```

When working with flat file source data, it is important to understand how the Flat File connection manager interprets the flat file data. If the flat file source is Unicode, the Flat File connection manager defines all columns as [DT\_WSTR] with a default column width of 50. If the flat file source is ANSI-encoded, the columns are defined as [DT\_STR] with a column width of 50. You will probably have to change these defaults to make the string column types more appropriate for your data. To do this, you will need to look at the data type of the destination where the data will be written to and then choose the correct type within the Flat File connection manager.

## Looking at the Destination

The ultimate destination for the source data is the **FactCurrency** fact table in **AdventureWorksDW**. The **FactCurrency** fact table has four columns, and has relationships to two dimension tables, as shown in the following table.

COLUMN NAME	DATA TYPE	LOOKUP TABLE	LOOKUP COLUMN
AverageRate	float	None	None
CurrencyKey	int (FK)	DimCurrency	CurrencyKey (PK)
DateKey	Int (FK)	DimDate	DateKey (PK)
EndOfDayRate	float	None	None

## Mapping Source Data to be Compatible with the Destination

Analysis of the source and destination data formats indicates that lookups will be necessary for the **CurrencyKey** and **DateKey** values. The transformations that will perform these lookups will obtain the **CurrencyKey** and **DateKey** values by using the alternate keys from **DimCurrency** and **DimDate** dimension tables.

FLAT FILE COLUMN	TABLE NAME	COLUMN NAME	DATA TYPE
0	FactCurrency	AverageRate	float
1	DimCurrency	CurrencyAlternateKey	nchar (3)
2	DimDate	FullDateAlternateKey	date
3	FactCurrency	EndOfDayRate	float

## Lesson Tasks

This lesson contains the following tasks:

- [Step 1: Creating a New Integration Services Project](#)
- [Step 2: Adding and Configuring a Flat File Connection Manager](#)
- [Step 3: Adding and Configuring an OLE DB Connection Manager](#)
- [Step 4: Adding a Data Flow Task to the Package](#)
- [Step 5: Adding and Configuring the Flat File Source](#)
- [Step 6: Adding and Configuring the Lookup Transformations](#)
- [Step 7: Adding and Configuring the OLE DB Destination](#)
- [Step 8: Making the Lesson 1 Package Easier to Understand](#)
- [Step 9: Testing the Lesson 1 Tutorial Package](#)

## Start the Lesson

[Step 1: Creating a New Integration Services Project](#)

# Lesson 1-1 - Creating a New Integration Services Project

3/24/2017 • 1 min to read • [Edit Online](#)

The first step in creating a package in Integration Services is to create an Integration Services project. This project includes the templates for the objects — data sources, data source views, and packages — that you use in a data transformation solution.

The packages that you will create in this Integration Services tutorial interpret the values of locale-sensitive data. If your computer is not configured to use the regional option English (United States), you need to set additional properties in the package. The packages that you use in lessons 2 through 5 are copied from the package created in lesson 1, and you need not update locale-sensitive properties in the copied packages.

## NOTE

This tutorial requires Microsoft SQL Server Data Tools.

For more information on installing the SQL Server Data Tools see [SQL Server Data Tools Download](#).

## To create a new Integration Services project

1. On the **Start** menu, point to **All Programs**, point to **Microsoft SQL Server**, and click **SQL Server Data Tools**.
2. On the **File** menu, point to **New**, and click **Project** to create a new Integration Services project.
3. In the **New Project** dialog box, expand the **Business Intelligence** node under **Installed Templates**, and select **Integration Services Project** in the **Templates** pane.
4. In the **Name** box, change the default name to **SSIS Tutorial**. Optionally, clear the **Create directory for solution** check box.
5. Accept the default location, or click **Browse** to browse to locate the folder you want to use. In the **Project Location** dialog box, click the folder and click **Select Folder**.
6. Click **OK**.

By default, an empty package, titled **Package.dtsx**, will be created and added to your project under SSIS Packages.

7. In **Solution Explorer** toolbar, right-click **Package.dtsx**, click **Rename**, and rename the default package to **Lesson 1.dtsx**.

## Next Task in Lesson

[Step 2: Adding and Configuring a Flat File Connection Manager](#)



# Lesson 1-2 - Adding and Configuring a Flat File Connection Manager

3/24/2017 • 4 min to read • [Edit Online](#)

In this task, you add a Flat File connection manager to the package that you just created. A Flat File connection manager enables a package to extract data from a flat file. Using the Flat File connection manager, you can specify the file name and location, the locale and code page, and the file format, including column delimiters, to apply when the package extracts data from the flat file. In addition, you can manually specify the data type for the individual columns, or use the **Suggest Column Types** dialog box to automatically map the columns of extracted data to Integration Services data types.

You must create a new Flat File connection manager for each file format that you work with. Because this tutorial extracts data from multiple flat files that have exactly the same data format, you will need to add and configure only one Flat File connection manager for your package.

For this tutorial, you will configure the following properties in your Flat File connection manager:

- **Column names:** Because the flat file does not have column names, the Flat File connection manager creates default column names. These default names are not useful for identifying what each column represents. To make these default names more useful, you need to change the default names to names that match the fact table into which the flat file data is to be loaded.
- **Data mappings:** The data type mappings that you specify for the Flat File connection manager will be used by all flat file data source components that reference the connection manager. You can either manually map the data types by using the Flat File connection manager, or you can use the **Suggest Column Types** dialog box. In this tutorial, you will view the mappings suggested in the **Suggest Column Types** dialog box and then manually make the necessary mappings in the **Flat File Connection Manager Editor** dialog box.

The Flat File connection manager provides locale information about the data file. If your computer is not configured to use the regional option English (United States), you must set additional properties in the **Flat File Connection Manager Editor** dialog box.

## To add a Flat File connection manager to the SSIS package

1. Right-click anywhere in the **Connection Managers** area, and then click **New Flat File Connection**.
2. In the **Flat File Connection Manager Editor** dialog box, for **Connection manager name**, type **Sample Flat File Source Data**.
3. Click **Browse**.
4. In the **Open** dialog box, locate the SampleCurrencyData.txt file on your machine.

The sample data is included with the SSIS lesson packages. To download the sample data and the lesson packages, do the following.

- a. Navigate to [Integration Services Product Samples](#)
- b. Click the **DOWNLOADS** tab.
- c. Click the SQL2012.Integration\_Services.Create\_Simple\_ETL\_Tutorial.Sample.zip file.

5. Clear the Column names in the first data row checkbox.

## To set locale sensitive properties

1. In the **Flat File Connection Manager Editor** dialog box, click **General**.
2. Set **Locale** to English (United States) and **Code page** to 1252.

#### To rename columns in the Flat File connection manager

1. In the **Flat File Connection Manager Editor** dialog box, click **Advanced**.
2. In the property pane, make the following changes:
  - Change the **Column 0** name property to **AverageRate**.
  - Change the **Column 1** name property to **CurrencyID**.
  - Change the **Column 2** name property to **CurrencyDate**.
  - Change the **Column 3** name property to **EndOfDayRate**.

#### NOTE

By default, all four of the columns are initially set to a string data type [DT\_STR] with an **OutputColumnWidth** of 50.

#### To remap column data types

1. In the **Flat File Connection Manager Editor** dialog box, click **Suggest Types**.

Integration Services automatically suggests the most appropriate data types based on the first 200 rows of data. You can also change these suggestion options to sample more or less data, to specify the default data type for integer or Boolean data, or to add spaces as padding to string columns.

For now, make no changes to the options in the **Suggest Column Types** dialog box, and click **OK** to have Integration Services suggest data types for columns. This returns you to the **Advanced** pane of the **Flat File Connection Manager Editor** dialog box, where you can view the column data types suggested by Integration Services. (If you click **Cancel**, no suggestions are made to column metadata and the default string (DT\_STR) data type is used.)

In this tutorial, Integration Services suggests the data types shown in the second column of the following table for the data from the SampleCurrencyData.txt file. However, the data types that are required for the columns in the destination, which will be defined in a later step, are shown in the last column of the following table.

FLAT FILE COLUMN	SUGGESTED TYPE	DESTINATION COLUMN	DESTINATION TYPE
AverageRate	float [DT_R4]	FactCurrency.AverageRate	float
CurrencyID	string [DT_STR]	DimCurrency.CurrencyAlternateKey	nchar(3)
CurrencyDate	date [DT_DATE]	DimDate.FullDateAlternateKey	date
EndOfDayRate	float [DT_R4]	FactCurrency.EndOfDayRate	float

The data type suggested for the **CurrencyID** column is incompatible with the data type of the field in the destination table. Because the data type of `DimCurrency.CurrencyAlternateKey` is nchar (3), **CurrencyID** must be changed from string [DT\_STR] to string [DT\_WSTR]. Additionally, the field `DimDate.FullDateAlternateKey` is defined as a date data type; therefore, **CurrencyDate** needs to be changed from date [DT\_Date] to database date [DT\_DBDATE].

2. In the list, select the CurrencyID column and in the property pane, change the Data Type of column **CurrencyID** from string [DT\_STR] to Unicode string [DT\_WSTR].
3. In the property pane, change the data type of column **CurrencyDate** from date [DT\_DATE] to database date [DT\_DBDATE].
4. Click **OK**.

## Next Task in Lesson

[Step 3: Adding and Configuring an OLE DB Connection Manager](#)

## See Also

[Flat File Connection Manager](#)

[Integration Services Data Types](#)

# Lesson 1-3 - Adding and Configuring an OLE DB Connection Manager

3/24/2017 • 1 min to read • [Edit Online](#)

After you have added a Flat File connection manager to connect to the data source, the next task is to add an OLE DB connection manager to connect to the destination. An OLE DB connection manager enables a package to extract data from or load data into any OLE DB–compliant data source. Using the OLE DB Connection manager, you can specify the server, the authentication method, and the default database for the connection.

In this lesson, you will create an OLE DB connection manager that uses Windows Authentication to connect to the local instance of **AdventureWorksDW2012**. The OLE DB connection manager that you create will also be referenced by other components that you will create later in this tutorial, such as the Lookup transformation and the OLE DB destination.

## Add and configure an OLE DB Connection Manager to the SSIS package

1. Right-click anywhere in the **Connection Managers** area, and then click **New OLE DB Connection**.
2. In the **Configure OLE DB Connection Manager** dialog box, click **New**.
3. For **Server name**, enter **localhost**.

When you specify localhost as the server name, the connection manager connects to the default instance of SQL Server on the local computer. To use a remote instance of SQL Server, replace localhost with the name of the server to which you want to connect.

4. In the **Log on to the server** group, verify that **Use Windows Authentication** is selected.
5. In the **Connect to a database** group, in the **Select or enter a database name** box, type or select **AdventureWorksDW2012**.
6. Click **Test Connection** to verify that the connection settings you have specified are valid.
7. Click **OK**.
8. Click **OK**.
9. In the **Data Connections** pane of the **Configure OLE DB Connection Manager** dialog box, verify that **localhost.AdventureWorksDW2012** is selected.
10. Click **OK**.

## Next Task in Lesson

[Step 4: Adding a Data Flow Task to the Package](#)

## See Also

[OLE DB Connection Manager](#)

# Lesson 1-4 - Adding a Data Flow Task to the Package

3/24/2017 • 1 min to read • [Edit Online](#)

After you have created the connection managers for the source and destination data, the next task is to add a Data Flow task to your package. The Data Flow task encapsulates the data flow engine that moves data between sources and destinations, and provides the functionality for transforming, cleaning, and modifying data as it is moved. The Data Flow task is where most of the work of an extract, transform, and load (ETL) process occurs.

## NOTE

SQL Server Integration Services separates data flow from control flow.

### To add a Data Flow task

1. Click the **Control Flow** tab.
2. In the **SSIS Toolbox**, expand **Favorites**, and drag a **Data Flow Task** onto the design surface of the **Control Flow** tab.

## NOTE

If the SSIS Toolbox isn't available, on the main menu select SSIS then SSIS Toolbox to display the SSIS Toolbox.

3. 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. Following these naming guidelines allows your Integration Services packages to be self-documenting. Another way to document your packages is by using annotations. For more information about annotations, see [Use Annotations in Packages](#).

4. Right-click the Data Flow task, click **Properties**, and in the Properties window, verify that the **LocaleID** property is set to **English (United States)**.

## Next Task in Lesson

[Step 5: Adding and Configuring the Flat File Source](#)

## See Also

[Data Flow Task](#)

# Lesson 1-5 - Adding and Configuring the Flat File Source

3/24/2017 • 1 min to read • [Edit Online](#)

In this task, you will add and configure a Flat File source to your package. A Flat File source is a data flow component that uses metadata defined by a Flat File connection manager to specify the format and structure of the data to be extracted from the flat file by a transform process. The Flat File source can be configured to extract data from a single flat file by using the file format definition provided by the Flat File connection manager.

For this tutorial, you will configure the Flat File source to use the **Sample Flat File Source Data** connection manager that you previously created.

## To add a Flat File Source component

1. Open the **Data Flow** designer, either by double-clicking the **Extract Sample Currency Data** data flow task or by clicking the **Data Flow** tab.
2. In the **SSIS Toolbox**, expand **OtherSources**, and then drag a **Flat File Source** onto the design surface of the **Data Flow** tab.
3. On the **Data Flow** design surface, right-click the newly added **Flat File Source**, click **Rename**, and change the name to **Extract Sample Currency Data**.
4. Double-click the Flat File source to open the Flat File Source Editor dialog box.
5. In the **Flat file connection manager** box, select **Sample Flat File Source Data**.
6. Click **Columns** and verify that the names of the columns are correct.
7. Click **OK**.
8. Right-click the Flat File source and click **Properties**.
9. In the Properties window, verify that the **LocaleID** property is set to **English (United States)**.

## Next Task in Lesson

[Step 6: Adding and Configuring the Lookup Transformations](#)

## See Also

[Flat File Source](#)

[Flat File Connection Manager Editor \(General Page\)](#)

# Lesson 1-6 - Adding and Configuring the Lookup Transformations

3/24/2017 • 3 min to read • [Edit Online](#)

After you have configured the Flat File source to extract data from the source file, the next task is to define the Lookup transformations needed to obtain the values for the **CurrencyKey** and **DateKey**. A Lookup transformation performs a lookup by joining data in the specified input column to a column in a reference dataset. The reference dataset can be an existing table or view, a new table, or the result of an SQL statement. In this tutorial, 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.

## NOTE

You can also configure the Lookup transformation to connect to a cache that contains the reference dataset. For more information, see [Lookup Transformation](#).

For this tutorial, you will add and configure the following two Lookup transformation components to the package:

- One transformation to perform a lookup of values from the **CurrencyKey** column of the **DimCurrency** dimension table based on matching **CurrencyID** column values from the flat file.
- One transformation to perform a lookup of values from the **DateKey** column of the **DimDate** dimension table based on matching **CurrencyDate** column values from the flat file.

In both cases, the Lookup transformation will utilize the OLE DB connection manager that you previously created.

## To add and configure the Lookup Currency Key transformation

1. In the **SSIS Toolbox**, expand **Common**, and then drag **Lookup** onto the design surface of the **Data Flow** tab. Place Lookup directly below the **Extract Sample Currency Data** source.
2. Click the **Extract Sample Currency Data** flat file source and drag the green arrow onto the newly added **Lookup** transformation to connect the two components.
3. On the **Data Flow** design surface, click **Lookup** in the **Lookup** transformation, and change the name to **Lookup Currency Key**.
4. Double-click the **Lookup CurrencyKey** transformation to display the Lookup Transformation Editor.
5. On the **General** page, make the following selections:
  - a. Select **Full cache**.
  - b. In the **Connection type** area, select **OLE DB connection manager**.
6. On the **Connection** page, make the following selections:
  - a. In the **OLE DB connection manager** dialog box, ensure that **localhost.AdventureWorksDW2012** is displayed.
  - b. Select **Use results of an SQL query**, and then type or copy the following SQL statement:

```

select * from (select * from [dbo].[DimCurrency]) as refTable
where [refTable].[CurrencyAlternateKey] = 'ARS'
OR
[refTable].[CurrencyAlternateKey] = 'AUD'
OR
[refTable].[CurrencyAlternateKey] = 'BRL'
OR
[refTable].[CurrencyAlternateKey] = 'CAD'
OR
[refTable].[CurrencyAlternateKey] = 'CNY'
OR
[refTable].[CurrencyAlternateKey] = 'DEM'
OR
[refTable].[CurrencyAlternateKey] = 'EUR'
OR
[refTable].[CurrencyAlternateKey] = 'FRF'
OR
[refTable].[CurrencyAlternateKey] = 'GBP'
OR
[refTable].[CurrencyAlternateKey] = 'JPY'
OR
[refTable].[CurrencyAlternateKey] = 'MXN'
OR
[refTable].[CurrencyAlternateKey] = 'SAR'
OR
[refTable].[CurrencyAlternateKey] = 'USD'
OR
[refTable].[CurrencyAlternateKey] = 'VEB'

```

7. On the **Columns** page, make the following selections:
  - a. In the **Available Input Columns** panel, drag **CurrencyID** to the **Available Lookup Columns** panel and drop it on **CurrencyAlternateKey**.
  - b. In the **Available Lookup Columns** list, select the check box to the left of **CurrencyKey**.
8. Click **OK** to return to the **Data Flow** design surface.
9. Right-click the Lookup Currency Key transformation, click **Properties**.
10. In the Properties window, verify that the **LocaleID** property is set to **English (United States)** and the **DefaultCodePage** property is set to **1252**.

#### To add and configure the Lookup DateKey transformation

1. In the **SSIS Toolbox**, drag **Lookup** onto the **Data Flow** design surface. Place Lookup directly below the **Lookup Currency Key** transformation.
2. Click the **Lookup Currency Key** transformation and drag the green arrow onto the newly added **Lookup** transformation to connect the two components.
3. In the **Input Output Selection** dialog box, click **Lookup Match Output** in the **Output** list box, and then click **OK**.
4. On the **Data Flow** design surface, click **Lookup** in the newly added **Lookup** transformation, and change the name to **Lookup Date Key**.
5. Double-click the **Lookup Date Key** transformation.
6. On the **General** page, select **Partial cache**.
7. On the **Connection** page, make the following selections:
  - a. In the **OleDb connection manager** dialog box, ensure that **localhost.AdventureWorksDW2012**



is displayed.

- b. In the **Use a table or view** box, type or select **[dbo].[DimDate]**.
8. On the **Columns** page, make the following selections:
  - a. In the **Available Input Columns** panel, drag **CurrencyDate** to the **Available Lookup Columns** panel and drop it on **FullDateAlternateKey**.
  - b. In the **Available Lookup Columns** list, select the check box to the left of **DateKey**.
9. On the **Advanced** page, review the caching options.
10. Click **OK** to return to the **Data Flow** design surface.
11. Right-click the Lookup Date Key transformation and click **Properties**.
12. In the Properties window, verify that the **LocaleID** property is set to **English (United States)** and the **DefaultCodePage** property is set to **1252**.

## Next Task in Lesson

[Step 7: Adding and Configuring the OLE DB Destination](#)

## See Also

[Lookup Transformation](#)

# Lesson 1-7 - Adding and Configuring the OLE DB Destination

3/24/2017 • 1 min to read • [Edit Online](#)

Your package now can extract data from the flat file source and transform that data into a format that is compatible with the destination. The next task is to actually load the transformed data into the destination. To load the data, you must add an OLE DB destination to the data flow. The OLE DB destination can use a database table, view, or an SQL command to load data into a variety of OLE DB-compliant databases.

In this procedure, you add and configure an OLE DB destination to use the OLE DB connection manager that you previously created.

## To add and configure the sample OLE DB destination

1. In the **SSIS Toolbox**, expand **Other Destinations**, and drag **OLE DB Destination** onto the design surface of the **Data Flow** tab. Place the OLE DB destination directly below the **Lookup Date Key** transformation.
2. Click the **Lookup Date Key** transformation and drag the green arrow over to the newly added **OLE DB Destination** to connect the two components together.
3. In the **Input Output Selection** dialog box, in the **Output** list box, click **Lookup Match Output**, and then click **OK**.
4. On the **Data Flow** design surface, click **OLE DB Destination** in the newly added **OLE DB Destination** component, and change the name to **Sample OLE DB Destination**.
5. Double-click **Sample OLE DB Destination**.
6. In the **OLE DB Destination Editor** dialog box, ensure that **localhost.AdventureWorksDW2012** is selected in the **OLE DB Connection manager** box.
7. In the **Name of the table or the view** box, type or select **[dbo].[FactCurrencyRate]**.
8. Click the **New** button to create a new table. Change the name of the table in the script to read **NewFactCurrencyRate**. Click **OK**.
9. Upon clicking **OK**, the dialog will close and the **Name of the table or the view** will automatically change to **NewFactCurrencyRate**.
10. Click **Mappings**.
11. Verify that the **AverageRate**, **CurrencyKey**, **EndOfDayRate**, and **DateKey** input columns are mapped correctly to the destination columns. If same-named columns are mapped, the mapping is correct.
12. Click **OK**.
13. Right-click the **Sample OLE DB Destination** destination and click **Properties**.
14. In the Properties window, verify that the **LocaleID** property is set to **English (United States)** and the **DefaultCodePage** property is set to **1252**.

## Next Task in Lesson

[Step 8: Making the Lesson 1 Package Easier to Understand](#)

## See Also

[OLE DB Destination](#)

# Lesson 1-8 - Making the Lesson 1 Package Easier to Understand

3/24/2017 • 1 min to read • [Edit Online](#)

Now that you have completed the configuration of the Lesson 1 package, it is a good idea to tidy up the package layout. If the shapes in the control and data flow layouts are random sizes, or if the shapes are not aligned or grouped, the functionality of package can be more difficult to understand.

SQL Server Data Tools provides tools that make it easy and quick to format the package layout. The formatting features include the ability to make shapes the same size, align shapes, and manipulate the horizontal and vertical spacing between shapes.

Another way to improve the understanding of package functionality is to add annotations that describe package functionality.

In this task, you will use the formatting features in SQL Server Data Tools to improve the layout of the data flow and also add an annotation to the data flow.

## To format the layout of the data flow

1. If the Lesson 1 package is not already open, double-click Lesson 1.dtsx in Solution Explorer.
2. Click the **Data Flow** tab.
3. Place the cursor to the top and to the right of the Extract Sample Currency transformation, click, and then drag the cursor across all the data flow components.
4. On the **Format** menu, point to **Make Same Size**, and then click **Both**.
5. With the data flow objects selected, on the **Format** menu, point to **Align**, and then click **Lefts**.

## To add an annotation to the data flow

1. Right-click anywhere in the background of the data flow design surface and then click **Add Annotation**.
2. Type or paste the following text in the annotation box.

**The data flow extracts data from a file, looks up values in the CurrencyKey column in the DimCurrency table and the DateKey column in the DimDate table, and writes the data to the NewFactCurrencyRate table.**

To wrap the text in the annotation box, place the cursor where you want to start a new line and press the Enter key.

If you do not add text to the annotation box, it disappears when you click outside the box.

## Next Steps

[Step 9: Testing the Lesson 1 Tutorial Package](#)

# Lesson 1-9 - Testing the Lesson 1 Tutorial Package

3/24/2017 • 1 min to read • [Edit Online](#)

In this lesson, you have done the following tasks:

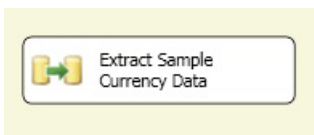
- Created a new SSIS project.
- Configured the connection managers that the package needs to connect to the source and destination data.
- Added a data flow that takes the data from a flat file source, performs the necessary Lookup transformations on the data, and configures the data for the destination.

Your package is now complete! It is time to test your package.

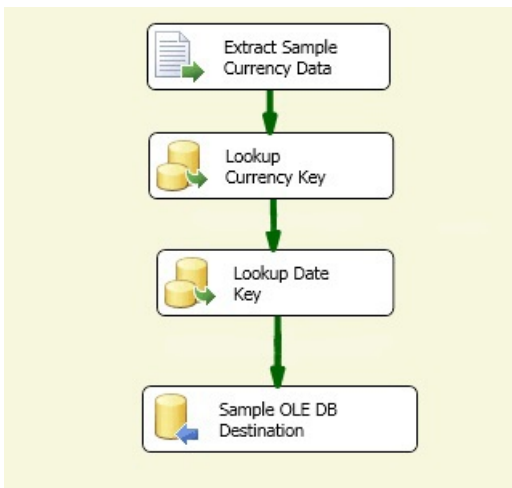
## Checking the Package Layout

Before you test the package you should verify that the control and data flows in the Lesson 1 package contain the objects shown in the following diagrams.

### Control Flow



### Data Flow



### To run the Lesson 1 tutorial package

1. On the **Debug** menu, click **Start Debugging**.

The package will run, resulting in 1097 rows successfully added into the **FactCurrency** fact table in **AdventureWorksDW2012**.

2. After the package has completed running, on the **Debug** menu, click **Stop Debugging**.

## Next Lesson

[Lesson 2: Adding Looping with SSIS](#)

## See Also

[Execution of Projects and Packages](#)

# Lesson 2: Adding Looping with SSIS

3/24/2017 • 1 min to read • [Edit Online](#)

In [Lesson 1: Create a Project and Basic Package with SSIS](#), you created a package that extracted data from a single flat file source, transformed the data using Lookup transformations, and finally loaded the data into the **FactCurrency** fact table of the **AdventureWorksDW2012** sample database.

However, it is rare for an extract, transform, and load (ETL) process to use a single flat file. A typical ETL process would extract data from multiple flat file sources. Extracting data from multiple sources requires an iterative control flow. One of the most anticipated features of Microsoft Integration Services is the ability to easily add iteration or looping to packages.

Integration Services provides two types of containers for looping through packages: the Foreach Loop container and the For Loop container. The Foreach Loop container uses an enumerator to perform the looping, whereas the For Loop container typically uses a variable expression. This lesson uses the Foreach Loop container.

The Foreach Loop container enables a package to repeat the control flow for each member of a specified enumerator. With the Foreach Loop container, you can enumerate:

- ADO recordset rows
- ADO .Net schema information
- File and directory structures
- System, package and user variables
- Enumerable objects contained in a variable
- Items in a collection
- Nodes in an XML Path Language (XPath) expression
- SQL Server Management Objects (SMO)

In this lesson, you will modify the simple ETL package created in Lesson 1 to take advantage of the Foreach Loop container. You will also set user-defined package variables to enable the tutorial package to iterate through all the flat files in the folder. If you have not completed the previous lesson, you can also copy the completed Lesson 1 package that is included with the tutorial.

In this lesson, you will not modify the data flow, only the control flow.

## IMPORTANT

This tutorial requires the **AdventureWorksDW2012** sample database. For more information about how to install and deploy **AdventureWorksDW2012**, see [Reporting Services Product Samples on CodePlex](#).

## Lesson Tasks

This lesson contains the following tasks:

- [Step 1: Copying the Lesson 1 Package](#)
- [Step 2: Adding and Configuring the Foreach Loop Container](#)

- [Step 3: Modifying the Flat File Connection Manager](#)
- [Step 4: Testing the Lesson 2 Tutorial Package](#)

## Start the Lesson

[Step 1: Copying the Lesson 1 Package](#)

## See Also

[For Loop Container](#)



# Lesson 2-1 - Copying the Lesson 1 Package

3/24/2017 • 1 min to read • [Edit Online](#)

In this task, you will create a copy of the Lesson 1.dtsx package that you created in Lesson 1. If you did not complete Lesson 1, you can add the completed lesson 1 package that is included with the tutorial to the project, and then copy it instead. You will use this new copy throughout the rest of Lesson 2.

## To create the Lesson 2 package

1. If SQL Server Data Tools is not already open, click **Start**, point to **All Programs**, point to **Microsoft SQL Server 2012**, and then click **SQL Server Data Tools**.
2. On the **File** menu, click **Open**, click **Project/Solution**, click the **SSIS Tutorial** folder and click **Open**, and then double-click **SSIS Tutorial.sln**.
3. In Solution Explorer, right-click **Lesson 1.dtsx**, and then click **Copy**.
4. In Solution Explorer, right-click **SSIS Packages**, and then click **Paste**.

By default, the copied package will be named Lesson 2.dtsx.

5. In Solution Explorer, double-click **Lesson 2.dtsx** to open the package
6. Right-click anywhere in the background of the **Control Flow** design surface and click **Properties**.
7. In the Properties window, update the **Name** property to **Lesson 2**.
8. Click the box for the **ID** property, click the dropdown arrow and then click .

## To add the completed Lesson 1 package

1. Open SQL Server Data Tools and open the SSIS Tutorial project.
2. In Solution Explorer, right-click **SSIS Packages**, and click **Add Existing Package**.
3. In the **Add Copy of Existing Package** dialog box, in **Package location**, select **File system**.
4. Click the browse (...) button, navigate to **Lesson 1.dtsx** on your machine, and then click **Open**.

To download all of the lesson packages for this tutorial, do the following.

- a. Navigate to [Integration Services Product Samples](#)
  - b. Click the **DOWNLOADS** tab.
  - c. Click the SQL2012.Integration\_Services.Create\_Simple\_ETL\_Tutorial.Sample.zip file.
5. Copy and paste the Lesson 1 package as described in steps 3-8 in the previous procedure.

## Next Task in Lesson

[Step 2: Adding and Configuring the Foreach Loop Container](#)

# Lesson 2-2 - Adding and Configuring the Foreach Loop Container

4/14/2017 • 3 min to read • [Edit Online](#)

In this task, you will add the ability to loop through a folder of flat files and apply the same data flow transformation used in Lesson 1 to each of those flat files. You do this by adding and configuring a Foreach Loop container to the control flow.

The Foreach Loop container that you add must be able to connect to each flat file in the folder. Because all the files in the folder have the same format, the Foreach Loop container can use the same Flat File connection manager to connect to each of these files. The Flat File connection manager that the container will use is the same Flat File connection manager that you created in Lesson 1.

Currently, the Flat File connection manager from Lesson 1 connects to only one, specific flat file. To iteratively connect to each flat file in the folder, you will have to configure both the Foreach Loop container and the Flat File connection manager as follows:

- **Foreach Loop container:** You will map the enumerated value of the container to a user-defined package variable. The container will then use this user-defined variable to dynamically modify the **ConnectionString** property of the Flat File connection manager and iteratively connect to each flat file in the folder.
- **Flat File connection manager:** You will modify the connection manager that was created in Lesson 1 by using a user-defined variable to populate the connection manager's **ConnectionString** property.

The procedures in this task show you how to create and modify the Foreach Loop container to use a user-defined package variable and to add the data flow task to the loop. You will learn how to modify the Flat File connection manager to use a user-defined variable in the next task.

After you have made these modifications to the package, when the package is run, the Foreach Loop Container will iterate through the collection of files in the Sample Data folder. Each time a file is found that matches the criteria, the Foreach Loop Container will populate the user-defined variable with the file name, map the user-defined variable to the **ConnectionString** property of the Sample Currency Data Flat File connection manager, and then run the data flow against that file. Therefore, in each iteration of the Foreach Loop the Data Flow task will consume a different flat file.

## NOTE

Because Microsoft Integration Services separates control flow from data flow, any looping that you add to the control flow will not require modification to the data flow. Therefore, the data flow that you created in Lesson 1 does not have to be changed.

### To add a Foreach Loop container

1. In **SQL Server Data Tools**, click the **Control Flow** tab.
2. In the **SSIS Toolbox**, expand **Containers**, and then drag a **Foreach Loop Container** onto the design surface of the **Control Flow** tab.
3. Right-click the newly added **Foreach Loop Container** and select **Edit**.
4. In the **Foreach Loop Editor** dialog box, on the **General** page, for **Name**, enter **Foreach File in Folder**.

Click **OK**.

5. Right-click the Foreach Loop container, click **Properties**, and in the Properties window, verify that the **LocaleID** property is set to **English (United States)**.

#### To configure the enumerator for the Foreach Loop container

1. Double-click Foreach File in Folder to reopen the **Foreach Loop Editor**.
2. Click **Collection**.
3. On the **Collection** page, select **Foreach File Enumerator**.
4. In the **Enumerator configuration** group, click **Browse**.
5. In the **Browse for Folder** dialog box, locate the folder on your machine that contains the Currency\_\*.txt files.

This sample data is included with the SSIS lesson packages. To download the sample data and the lesson packages, do the following.

- a. Navigate to [Integration Services Product Samples](#).
  - b. Click the **DOWNLOADS** tab.
  - c. Click the link for the [SQL2012.Integration\\_Services.Create\\_Simple\\_ETL\\_Tutorial.Sample.zip](#) file.
6. In the **Files** box, type **Currency\_\*.txt**.

#### To map the enumerator to a user-defined variable

1. Click **Variable Mappings**.
2. On the **Variable Mappings** page, in the **Variable** column, click the empty cell and select **<New Variable...>**.
3. In the **Add Variable** dialog box, for **Name**, type **varFileName**.

#### IMPORTANT

Variable names are case sensitive.

4. Click **OK**.
5. Click **OK** again to exit the **Foreach Loop Editor** dialog box.

#### To add the data flow task to the loop

- Drag the **Extract Sample Currency Data** data flow task onto the Foreach Loop container now renamed **Foreach File in Folder**.

## Next Lesson Task

[Step 3: Modifying the Flat File Connection Manager](#)

## See Also

[Configure a Foreach Loop Container](#)

[Use Variables in Packages](#)

# Lesson 2-3 - Modifying the Flat File Connection Manager

3/24/2017 • 1 min to read • [Edit Online](#)

In this task, you will modify the Flat File connection manager that you created and configured in Lesson 1. When originally created, the Flat File connection manager was configured to statically load a single file. To enable the Flat File connection manager to iteratively load files, you must modify the `ConnectionString` property of the connection manager to accept the user-defined variable `User:varFileName`, which contains the path of the file to be loaded at run time.

By modifying the connection manager to use the value of the user-defined variable, `User::varFileName`, to populate the `ConnectionString` property of the connection manager, the connection manager will be able to connect to different flat files. At run time, each iteration of the Foreach Loop container will dynamically update the `User::varFileName` variable. Updating the variable, in turn, causes the connection manager to connect to a different flat file, and the data flow task to process a different set of data.

## To configure the Flat File connection manager to use a variable for the connection string

1. In the **Connection Managers** pane, right-click **Sample Flat File Source Data**, and select **Properties**.
2. In the Properties window, for **Expressions**, click in the empty cell, and then click the ellipsis button (...).
3. In the **Property Expressions Editor** dialog box, in the **Property** column, type or select **ConnectionString**.
4. In the **Expression** column, click the ellipsis button (...) to open the **Expression Builder** dialog box.
5. In the **Expression Builder** dialog box, expand the **Variables** node.
6. Drag the variable, **User::varFileName**, into the **Expression** box.
7. Click **OK** to close the **Expression Builder** dialog box.
8. Click **OK** again to close the **Property Expressions Editor** dialog box.

## Next Lesson Task

[Step 4: Testing the Lesson 2 Tutorial Package](#)

# Lesson 2-4 - Testing the Lesson 2 Tutorial Package

3/24/2017 • 1 min to read • [Edit Online](#)

With the Foreach Loop container and the Flat File connection manager now configured, the Lesson 2 package can iterate through the collection of 14 flat files in the Sample Data folder. Each time a file name is found that matches the specified file name criteria, the Foreach Loop container populates the user-defined variable with the file name. This variable, in turn, updates the ConnectionString property of the Flat File connection manager, and a connection to the new flat file is made. The Foreach Loop container then runs the unmodified data flow task against the data in the new flat file before connecting to the next file in the folder.

Use the following procedure to test the new looping functionality that you have added to your package.

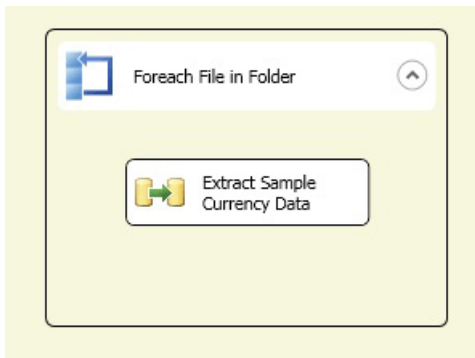
## NOTE

If you ran the package from Lesson 1, you will need to delete the records from dbo.FactCurrency in AdventureWorksDW2012 before you run the package from this lesson or the package will fail with errors indicating a Violation of Primary Key constraint. You will receive the same errors if you run the package by selecting Debug/Start Debugging (or press F5) because both Lesson 1 and Lesson 2 will run. Lesson 2 will attempt to insert records already inserted in Lesson 1.

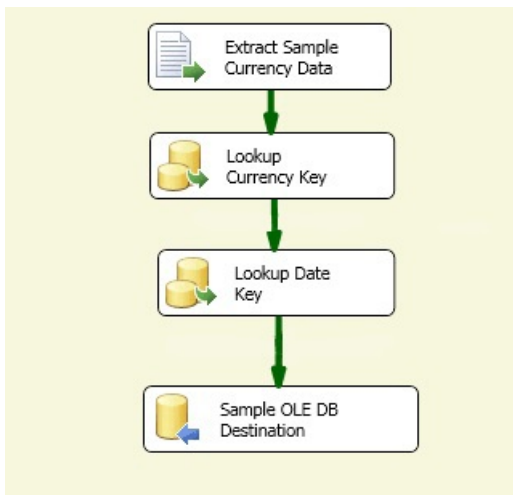
## Checking the Package Layout

Before you test the package you should verify that the control and data flows in the Lesson 2 package contains the objects shown in the following diagrams. The data flow should be identical to the data flow in lesson 1.

### Control Flow



### Data Flow



### To test the Lesson 2 tutorial package

1. In **Solution Explorer**, right-click **Lesson 2.dtsx** and click **Execute Package**.

The package will run. You can verify the status of each loop in the Output window, or by clicking on the **Progress** tab. For example, you can see that 1097 lines were added to the destination table from the file Currency\_VEB.txt.

2. After the package has completed running, on the **Debug** menu, click **Stop Debugging**.

## Next Lesson

[Lesson 5: Add SSIS Package Configurations for the Package Deployment Model](#)

## See Also

[Execution of Projects and Packages](#)

# Lesson 3: Add Logging with SSIS

3/24/2017 • 1 min to read • [Edit Online](#)

Microsoft Integration Services includes logging features that let you troubleshoot and monitor package execution by providing a trace of task and container events. The logging features are flexible, and can be enabled at the package level or on individual tasks and containers within the package. You can select which events you want to log, and create multiple logs against a single package.

Logging is provided by a log provider. Each log provider can write logging information to different formats and destination types. Integration Services provides the following log providers:

- Text file
- SQL Server Profiler
- Windows Event log
- SQL Server
- XML file

In this lesson, you will create a copy of the package that you created in [Lesson 2: Adding Looping with SSIS](#). Working with this new package, you will then add and configure logging to monitor specific events during package execution. If you have not completed any of the previous lessons, you can also copy the completed Lesson 2 package that is included with the tutorial.

## IMPORTANT

This tutorial requires the **AdventureWorksDW2012** sample database. For more information about how to install and deploy **AdventureWorksDW2012**, [Reporting Services Product Samples on CodePlex](#)

## Lesson Tasks

This lesson contains the following tasks:

- [Step 1: Copying the Lesson 2 Package](#)
- [Step 2: Adding and Configuring Logging](#)
- [Step 3: Testing the Lesson 3 Tutorial Package](#)

## Start the Lesson

[Step 1: Copying the Lesson 2 Package](#)

# Lesson 3-1 - Copying the Lesson 2 Package

3/24/2017 • 1 min to read • [Edit Online](#)

In this task, you will create a copy of the Lesson 2.dtsx package that you created in Lesson 2. Alternatively, you can add the completed lesson 2 package that is included with the tutorial to the project, and then copy it instead. You will use this new copy throughout the rest of Lesson 3.

## To create the Lesson 3 package

1. If SQL Server Data Tools is not already open, click **Start**, point to **All Programs**, point to **Microsoft SQL Server 2012**, and then click **SQL Server Data Tools**.
2. On the **File** menu, click **Open**, click **Project/Solution**, select **SSIS Tutorial** and click **Open**, and then double-click **SSIS Tutorial.sln**.
3. In Solution Explorer, right-click **Lesson 2.dtsx**, and then click **Copy**.
4. In Solution Explorer, right-click **SSIS Packages**, and then click **Paste**.

By default, the copied package is named Lesson 3.dtsx.

5. In Solution Explorer, double-click **Lesson 3.dtsx** to open the package.
6. Right-click anywhere in the background of the **Control Flow** tab and click **Properties**.
7. In the Properties window, update the **Name** property to **Lesson 3**.
8. Click the box for the **ID** property, and then in the list, click .

## To add the completed Lesson2 package

1. Open SQL Server Data Tools (SSDT) and open the SSIS Tutorial project.
2. In Solution Explorer, right-click **SSIS Packages**, and click **Add Existing Package**.
3. In the **Add Copy of Existing Package** dialog box, in **Package location**, select **File system**.
4. Click the browse (...) button, navigate to **Lesson 2.dtsx** on your machine, and then click **Open**.

To download all of the lesson packages for this tutorial, do the following.

- a. Navigate to [Integration Services Product Samples](#)
  - b. Click the **DOWNLOADS** tab.
  - c. Click the SQL2012.Integration\_Services.Create\_Simple\_ETL\_Tutorial.Sample.zip file.
5. Copy and paste the Lesson 3 package as described in steps 3-8 in the previous procedure.

## Next Task in Lesson

[Step 2: Adding and Configuring Logging](#)



# Lesson 3-2 - Adding and Configuring Logging

3/24/2017 • 2 min to read • [Edit Online](#)

In this task, you will enable logging for the data flow in the Lesson 3.dtsx package. Then, you will configure a Text File log provider to log the PipelineExecutionPlan and PipelineExecuteTrees events. The Text Files log provider creates logs that are easy to view and easily transportable. The simplicity of these log files makes these files especially useful during the basic testing phase of a package. You can also view the log entries in the Log Events window of SSIS Designer.

## To add logging to the package

1. On the **SSIS** menu, click **Logging**.
2. In the **Configure SSIS Logs** dialog box, in the **Containers** pane, make sure that the topmost object, which represents the Lesson 3 package, is selected.
3. On the **Providers and Logs** tab, in the **Provider type** box, select **SSIS log provider for Text files**, and then click **Add**.

Integration Services adds a new Text File log provider to the package with the default name **SSIS log provider for text files**. You can now configure the new log provider.

4. In the **Name** column, type **Lesson 3 Log File**.
5. Optionally, modify the **Description**.
6. In the **Configuration** column, click to specify the destination to which the log information is written.

In the **File Connection Manager Editor** dialog box, for **Usage type**, select **Create file**, and then click **Browse**. By default, the **Select File** dialog box opens the project folder, but you can save log information to any location.

7. In the **Select File** dialog box, in the **File name** box type **TutorialLog.log**, and click **Open**.
8. Click **OK** to close the **File Connection Manager Editor** dialog box.
9. In the **Containers** pane, expand all nodes of the package container hierarchy, and then clear all check boxes, including the **Extract Sample Currency Data** check box. Now select the check box for **Extract Sample Currency Data** to get only the events for this node.

### IMPORTANT

If the state of the **Extract Sample Currency Data** check box is dimmed instead of selected, the task uses the log settings of the parent container and you cannot enable the log events that are specific to the task.

10. On the **Details** tab, in the **Events** column, select the **PipelineExecutionPlan** and **PipelineExecutionTrees** events.
11. Click **Advanced** to review the details that the log provider will write to the log for each event. By default, all information categories are automatically selected for the events you specify.
12. Click **Basic** to hide the information categories.
13. On the **Provider and Logs** tab, in the **Name** column, select **Lesson 3 Log File**. Once you have created a log provider for your package, you can optionally deselect it to temporarily turn off logging, without having to

delete and re-create a log provider.

14. Click **OK**.

## Next Steps

[Step 3: Testing the Lesson 3 Tutorial Package](#)

# Lesson 3-3 - Testing the Lesson 3 Tutorial Package

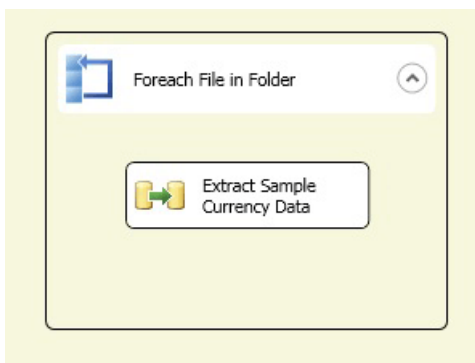
3/24/2017 • 1 min to read • [Edit Online](#)

In this task, you will run the Lesson 3.dtsx package. When the package runs, the Log Events window will list the log entries that are written to the log file. After the package finishes execution, you will then verify the contents of the log file that was generated by the log provider.

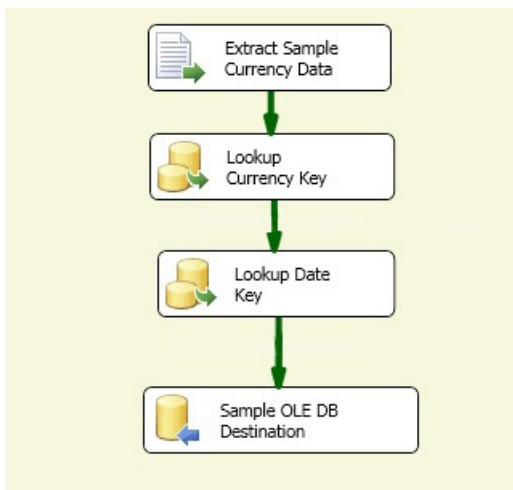
## Checking the Package Layout

Before you test the package you should verify that the control and data flows in the Lesson 3 package contain the objects shown in the following diagrams. The control flow should be identical to the control flow in lesson 2. The data flow should be identical to the data flow in lessons 1 and 2.

### Control Flow



### Data Flow



### To run the Lesson 4 tutorial package

1. On the SSIS menu, click Log Events.
2. On **Debug** menu, click **Start Debugging**.
3. After the package has completed running, on the **Debug** menu, click **Stop Debugging**.

### To examine the generated log file

- Using Notepad or any other text editor, open the TutorialLog.log file.
- Although the semantics of the information generated for the **PipelineExecutionPlan** and **PipelineExecutionTrees** events are beyond the scope of this tutorial, you can see that the first line lists the

information fields specified in the **Details** tab of the **Configure SSIS Logs** dialog box. Moreover, you can verify that the two events that you selected, PipelineExecutionPlan and PipelineExecutionTrees, have been logged for each iteration of the Foreach Loop.

## Next Lesson

[Lesson 4: Add Error Flow Redirection with SSIS](#)

# Lesson 4: Add Error Flow Redirection with SSIS

3/24/2017 • 1 min to read • [Edit Online](#)

To handle errors that may occur in the transformation process, Microsoft Integration Services gives you the ability to decide on a per component and per column basis how to handle data that cannot be transformed. You can choose to ignore a failure in certain columns, redirect the entire failed row, or just fail the component. By default, all components in Integration Services are configured to fail when errors occur. Failing a component, in turn, causes the package to fail and all subsequent processing to stop.

Instead of letting failures stop package execution, it is good practice to configure and handle potential processing errors as they occur within the transformation. While you might choose to ignore failures to ensure your package runs successfully, it is often better to redirect the failed row to another processing path where the data and the error can be persisted, examined and reprocessed at a later time.

In this lesson, you will create a copy of the package that you developed in [Lesson 3: Add Logging with SSIS](#). Working with this new package, you will create a corrupted version of one of the sample data files. The corrupted file will force a processing error to occur when you run the package.

To handle the error data, you will add and configure a Flat File destination that will write any rows that fail to locate a lookup value in the Lookup Currency Key transformation to a file.

Before the error data is written to the file, you will include a Script component that uses script to get error descriptions. You will then reconfigure the Lookup Currency Key transformation to redirect any data that could not be processed to the Script transformation.

## IMPORTANT

This tutorial requires the **AdventureWorksDW2012** sample database. For more information about how to install and deploy **AdventureWorksDW2012**, [Reporting Services Product Samples on CodePlex](#)

## Tasks in Lesson

This lesson contains the following tasks:

- [Step 1: Copying the Lesson 3 Package](#)
- [Step 2: Creating a Corrupted File](#)
- [Step 3: Adding Error Flow Redirection](#)
- [Step 4: Adding a Flat File Destination](#)
- [Step 5: Testing the Lesson 4 Tutorial Package](#)

## Start the Lesson

[Step 1: Copying the Lesson 3 Package](#)

# Lesson 4-1 - Copying the Lesson 3 Package

3/24/2017 • 1 min to read • [Edit Online](#)

In this task, you will create a copy of the Lesson 3.dtsx package that you created in Lesson 3. Alternatively, if you did not complete lesson 3, you can add the completed lesson 3 package that is included with the tutorial to the project, and then make a copy of it to work with. You will use this new copy throughout the rest of Lesson 4.

## To create the Lesson 4 package

1. If SQL Server Data Tools is not already open, click **Start**, point to **All Programs**, point to **Microsoft SQL Server**, and then click **SQL Server Data Tools**.
2. On the **File** menu, click **Open**, click **Project/Solution**, select **SSIS Tutorial** and click **Open**, and then double-click **SSIS Tutorial.sln**.
3. In Solution Explorer, right-click **Lesson 3.dtsx**, and then click **Copy**.
4. In Solution Explorer, right-click **SSIS Packages**, and then click **Paste**.

By default, the copied package is named Lesson 4.dtsx.

5. In Solution Explorer, double-click **Lesson 4.dtsx** to open the package.
6. Right-click anywhere in the background of the **Control Flow** tab and click **Properties**.
7. In the Properties window, update the **Name** property to **Lesson 4**.
8. Click the box for the **ID** property, and then in the list, click .

## To add the completed Lesson 3 package

1. Open SQL Server Data Tools (SSDT) and open the SSIS Tutorial project.
2. In Solution Explorer, right-click **SSIS Packages**, and click **Add Existing Package**.
3. In the **Add Copy of Existing Package** dialog box, in **Package location**, select **File system**.
4. Click the browse (...) button, navigate to Lesson 3.dtsx on your machine, and then click **Open**.

To download all of the lesson packages for this tutorial, do the following.

- a. Navigate to [Integration Services Product Samples](#)
  - b. Click the **DOWNLOADS** tab.
  - c. Click the SQL2012.Integration\_Services.Create\_Simple\_ETL\_Tutorial.Sample.zip file.
5. Copy and paste the Lesson 3 package as described in steps 3-8 in the previous procedure.

## Next Task in Lesson

[Step 2: Creating a Corrupted File](#)

# Lesson 4-2 - Creating a Corrupted File

3/24/2017 • 1 min to read • [Edit Online](#)

In order to demonstrate the configuration and handling of transformation errors, you will have to create a sample flat file that when processed causes a component to fail.

In this task, you will create a copy of an existing sample flat file. You will then open the file in Notepad and edit the **CurrencyID** column to ensure that it will fail to produce a match during the transformations lookup. When the new file is processed, the lookup failure will cause the Currency Key Lookup transformation to fail and therefore fail the rest of the package. After you have created the corrupted sample file, you will run the package to view the package failure.

## To create a corrupted sample flat file

1. In Notepad or any other text editor, open the Currency\_VEB.txt file.

The sample data is included with the SSIS Lesson packages. To download the sample data and the lesson packages, do the following.

- a. Navigate to [Integration Services Product Samples](#).
  - b. Click the **DOWNLOADS** tab.
  - c. Click the SQL2012.Integration\_Services.Create\_Simple\_ETL\_Tutorial.Sample.zip file.
2. Use the text editor's find and replace feature to find all instances of **VEB** and replace them with **BAD**.
  3. In the same folder as the other sample data files, save the modified file as **Currency\_BAD.txt**.

### IMPORTANT

Make sure that **Currency\_BAD.txt** is saved the same folder as the other sample data files.

4. Close your text editor.

## To verify that an error will occur during run time

1. On the **Debug** menu, click **Start Debugging**.

On the third iteration of the data flow, the Lookup Currency Key transformation tries to process the Currency\_BAD.txt file, and the transformation will fail. The failure of the transformation will cause the whole package to fail.

2. On the **Debug** menu, click **Stop Debugging**.
3. On the design surface, click the **Execution Results** tab.
4. Browse through the log and verify that the following unhandled error occurred:

```
[Lookup Currency Key[27]] Error: Row yielded no match during lookup.
```

### NOTE

The number 27 is the ID of the component. This value is assigned when you build the data flow, and the value in your package may be different.

# Next Steps

Step 3: Adding Error Flow Redirection



# Lesson 4-3 - Adding Error Flow Redirection

3/24/2017 • 2 min to read • [Edit Online](#)

As demonstrated in the previous task, the Lookup Currency Key transformation cannot generate a match when the transformation tries to process the corrupted sample flat file, which produced an error. Because the transformation uses the default settings for error output, any error causes the transformation to fail. When the transformation fails, the rest of the package also fails.

Instead of permitting the transformation to fail, you can configure the component to redirect the failed row to another processing path by using the error output. Use of a separate error processing path lets you do a number of things. For instance, you might try to clean the data and then reprocess the failed row. Or, you might save the failed row along with additional error information for later verification and reprocessing.

In this task, you will configure the Lookup Currency Key transformation to redirect any rows that fail to the error output. In the error branch of the data flow, these rows will be written to a file.

By default the two extra columns in an Integration Services error output, **ErrorCode** and **ErrorColumn**, contain only numeric codes that represent an error number, and the ID of the column in which the error occurred. These numeric values may be of limited use without the corresponding error description.

To enhance the usefulness of the error output, before the package writes the failed rows to the file, you will use a Script component to access the Integration Services API and get a description of the error.

## To configure an error output

1. In the **SSIS Toolbox**, expand **Common**, and then drag **Script Component** onto the design surface of the **Data Flow** tab. Place **Script** to the right of the **Lookup Currency Key** transformation.
2. In the **Select Script Component Type** dialog box, click **Transformation**, and click **OK**.
3. Click the **Lookup Currency Key** transformation and then drag the red arrow onto the newly added **Script** transformation to connect the two components.

The red arrow represents the error output of the **Lookup Currency Key** transformation. By using the red arrow to connect the transformation to the Script component, you can redirect any processing errors to the Script component, which then processes the errors and sends them to the destination.

4. In the **Configure Error Output** dialog box, in the **Error** column, select **Redirect row**, and then click **OK**.
5. On the **Data Flow** design surface, click **Script Component** in the newly added **ScriptComponent**, and change the name to **Get Error Description**.
6. Double-click the **Get Error Description** transformation.
7. In the **Script Transformation Editor** dialog box, on the **Input Columns** page, select the **ErrorCode** column.
8. On the **Inputs and Outputs** page, expand **Output 0**, click **Output Columns**, and then click **Add Column**.
9. In the **Name** property, type **ErrorDescription** and set the **DataType** property to **Unicode string [DT\_WSTR]**.
10. On the **Script** page, verify that the **LocaleID** property is set to **English (United States)**.
11. Click **Edit Script** to open Microsoft Visual Studio Tools for Applications (VSTA). In the **Input0\_ProcessInputRow** method, type or paste the following code.

[Visual Basic]

```
Row.ErrorDescription =  
    Me.ComponentMetaData.GetErrorDescription(Row.ErrorCode)
```

[Visual C#]

```
Row.ErrorDescription = this.ComponentMetaData.GetErrorDescription(Row.ErrorCode);
```

The completed subroutine will look like the following code.

[Visual Basic]

```
Public Overrides Sub Input0_ProcessInputRow(ByVal Row As Input0Buffer)  
  
    Row.ErrorDescription =  
        Me.ComponentMetaData.GetErrorDescription(Row.ErrorCode)  
  
End Sub
```

[Visual C#]

```
public override void Input0_ProcessInputRow(Input0Buffer Row)  
{  
  
    Row.ErrorDescription = this.ComponentMetaData.GetErrorDescription(Row.ErrorCode);  
  
}
```

12. On the **Build** menu, click **Build Solution** to build the script and save your changes, and then close VSTA.
13. Click **OK** to close the **Script Transformation Editor** dialog box.

## Next Steps

[Step 4: Adding a Flat File Destination](#)

# Lesson 4-4 - Adding a Flat File Destination

3/24/2017 • 2 min to read • [Edit Online](#)

The error output of the Lookup Currency Key transformation redirects to the Script transformation any data rows that failed the lookup operation. To enhance information about the errors that occurred, the Script transformation runs a script that gets the description of errors.

In this task, you will save all this information about the failed rows to a delimited file for later processing. To save the failed rows, you must add and configure a Flat File connection manager for the text file that will contain the error data and a Flat File destination. By setting properties on the Flat File connection manager that the Flat File destination uses, you can specify how the Flat File destination formats and writes the text file. For more information, see [Flat File Connection Manager](#) and [Flat File Destination](#).

## To add and configure a Flat File destination

1. Click the **Data Flow** tab.
2. In the **SSIS Toolbox**, expand **Other**, and drag **Flat File Destination** onto the data flow design surface. Put the **Flat File Destination** directly underneath the **Get Error Description** transformation.
3. Click the **Get Error Description** transformation, and then drag the green arrow onto the new **Flat File Destination**.
4. On the **Data Flow** design surface, click **Flat File Destination** in the newly added **Flat File Destination** transformation, and change the name to **Failed Rows**.
5. Right-click the **Failed Rows** transformation, click **Edit**, and then in the **Flat File Destination Editor**, click **New**.
6. In the **Flat File Format** dialog box, verify that **Delimited** is selected, and then click **OK**.
7. In the **Flat File Connection Manager Editor**, in the **Connection Manager Name** box type **Error Data**.
8. In the **Flat File Connection Manager Editor** dialog box, click **Browse**, and locate the folder in which to store the file.
9. In the **Open** dialog box, for **File name**, type **ErrorOutput.txt**, and then click **Open**.
10. In the **Flat File Connection Manager Editor** dialog box, verify that the **Locale** box contains English (United States) and **Code page** contains 1252 (ANSI -Latin I).
11. In the options pane, click **Columns**.

Notice that, in addition to the columns from the source data file, three new columns are present: **ErrorCode**, **ErrorColumn**, and **ErrorDescription**. These columns are generated by the error output of the Lookup Currency Key transformation and by the script in the Get Error Description transformation, and can be used to troubleshoot the cause of the failed row.

12. Click **OK**.
13. In the **Flat File Destination Editor**, clear the **Overwrite data in the file** check box.

Clearing this check box persists the errors over multiple package executions.

14. In the **Flat File Destination Editor**, click **Mappings** to verify that all the columns are correct. Optionally, you can rename the columns in the destination.

15. Click **OK**.

## Next Steps

[Step 5: Testing the Lesson 4 Tutorial Package](#)

# Lesson 4-5 - Testing the Lesson 4 Tutorial Package

3/24/2017 • 1 min to read • [Edit Online](#)

At run time, the corrupted file, Currency\_BAD.txt, will fail to generate a match within the Currency Key Lookup transformation. Because the error output of Currency Key Lookup has now been configured to redirect failed rows to the new Failed Rows destination, the component does not fail, and the package runs successfully. All failed error rows are written to ErrorOutput.txt.

In this task, you will test the revised error output configuration by running the package. Upon successful package execution, you will then view the contents of the ErrorOutput.txt file.

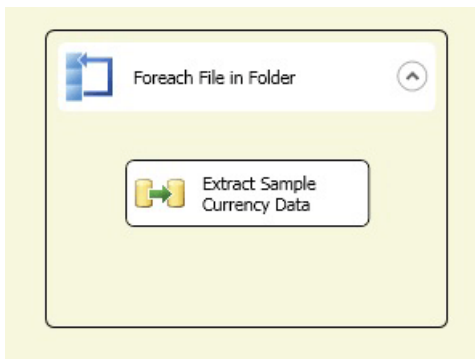
## NOTE

If you do not want to accumulate error rows in the ErrorOutput.txt file, you should manually delete the file content between package runs.

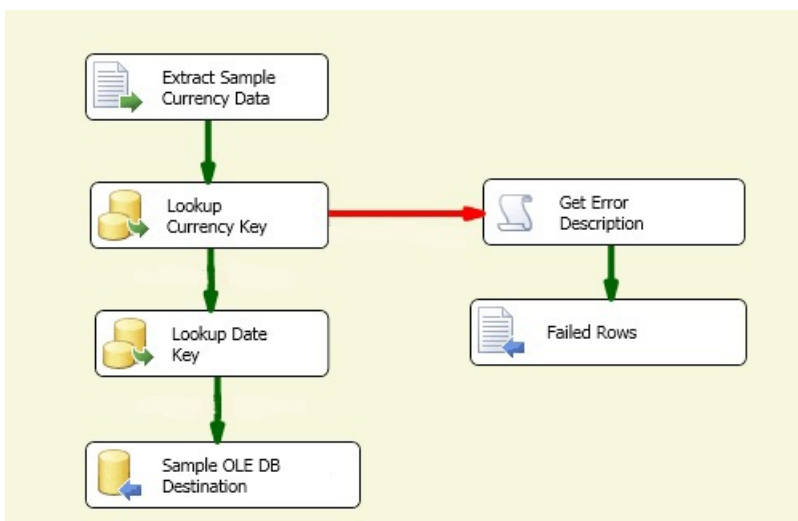
## Checking the Package layout

Before you test the package you should verify that the control flow and the data flow in the Lesson 4 package contain the objects shown in the following diagrams. The control flow should be identical to the control flow in lessons 2 - 4.

### Control Flow



### Data Flow



### To run the Lesson 4 tutorial package

1. On the **Debug** menu, click **Start Debugging**.
2. After the package has completed running, on the **Debug** menu, click **Stop Debugging**.

### To verify the contents of the ErrorOutput.txt file

- In Notepad or any other text editor, open the ErrorOutput.txt file. The default column order is: AverageRate, CurrencyID, CurrencyDate, EndOfDateRate, ErrorCode, ErrorColumn, ErrorDescription.

Notice that all the rows in the file contain the unmatched CurrencyID value of BAD, the ErrorCode value of -1071607778, the ErrorColumn value of 0, and the ErrorDescription value "Row yielded no match during lookup". The value of the ErrorColumn is set to 0 because the error is not column specific. It is the lookup operation that failed. .

# Lesson 5: Add SSIS Package Configurations for the Package Deployment Model

3/24/2017 • 1 min to read • [Edit Online](#)

Package configurations let you set run-time properties and variables from outside of the development environment. Configurations allow you to develop packages that are flexible and easy to both deploy and distribute. Microsoft Integration Services offers the following configuration types:

- XML configuration file
- Environment variable
- Registry entry
- Parent package variable
- SQL Server table

In this lesson, you will modify the simple Integration Services package that you created in [Lesson 4: Add Error Flow Redirection with SSIS](#) to use the Package Deployment Model and take advantage of package configurations. You can also copy the completed Lesson 4 package that is included with the tutorial. Using the Package Configuration Wizard, you will create an XML configuration that updates the **Directory** property of the Foreach Loop container by using a package-level variable mapped to the Directory property. Once you have created the configuration file, you will modify the value of the variable from outside of the development environment and point the modified property to a new sample data folder. When you run the package again, the configuration file populates the value of the variable, and the variable in turn updates the **Directory** property. As a result, the package iterates through the files in the new data folder, rather than iterating through the files in the original folder that was hard-coded in the package.

## IMPORTANT

This tutorial requires the **AdventureWorksDW2012** sample database. For more information about how to install and deploy **AdventureWorksDW2012**, see [Reporting Services Product Samples on CodePlex](#).

## Lesson Tasks

This lesson contains the following tasks:

- [Step 1: Copying the Lesson 4 Package](#)
- [Step 2: Enabling and Configuring Package Configurations](#)
- [Step 3: Modifying the Directory Property Configuration Value](#)
- [Step 4: Testing the Lesson 5 Tutorial Package](#)

## Start the Lesson

- [Step 1: Copying the Lesson 4 Package](#)

# Lesson 5-1 - Copying the Lesson 4 Package

3/24/2017 • 1 min to read • [Edit Online](#)

In this task, you will create a copy of the Lesson 4.dtsx package that you created in Lesson 4. Alternatively, you can add the completed lesson 4 package that is included with the tutorial to the project, and then copy it instead. You will use this new copy throughout the rest of Lesson 5.

## To copy the Lesson 4 package

1. If SQL Server Data Tools is not already open, click **Start**, point to **All Programs**, point to **Microsoft SQL Server 2012**, and then click **SQL Server Data Tools**.
2. On the **File** menu, click **Open**, click **Project/Solution**, select **SSIS Tutorial** and click **Open**, and then double-click **SSIS Tutorial.sln**.
3. In Solution Explorer, right-click **Lesson 4.dtsx**, and then click **Copy**.
4. In Solution Explorer, right-click **SSIS Packages**, and then click **Paste**.

By default, the copied package is named Lesson 5.dtsx.

5. In the Solution Explorer, double-click **Lesson 5.dtsx** to open the package.
6. Right-click anywhere in the background of the **Control Flow** tab then click **Properties**.
7. In the Properties window, update the **Name** property to **Lesson 5**.
8. Click the box for the **ID** property, then click the dropdown arrow, and then click .

## To add the completed Lesson 4 package

1. Open SQL Server Data Tools and open the SSIS Tutorial project.
2. In Solution Explorer, right-click **SSIS Packages**, and click **Add Existing Package**.
3. In the **Add Copy of Existing Package** dialog box, in **Package location**, select **File system**.
4. Click the browse (...) button, navigate to **Lesson 4.dtsx** on your machine, and then click **Open**.

To download all of the lesson packages for this tutorial, do the following.

- a. Navigate to [Integration Services Product Samples](#)
  - b. Click the **DOWNLOADS** tab.
  - c. Click the SQL2012.Integration\_Services.Create\_Simple\_ETL\_Tutorial.Sample.zip file.
5. Copy and paste the Lesson 4 package as described in steps 3-8 in the previous procedure.

## Next Task in Lesson

[Step 2: Enabling and Configuring Package Configurations](#)



# Lesson 5-2 - Enabling and Configuring Package Configurations

3/24/2017 • 2 min to read • [Edit Online](#)

In this task, you will convert the project to the Package Deployment Model and enable package configurations using the Package Configuration Wizard. You will use this wizard to generate an XML configuration file that contains configuration settings for the **Directory** property of the Foreach Loop container. The value of the Directory property is supplied by a new package-level variable that you can update at run time. Additionally, you will populate a new sample data folder to use during testing.

## To create a new package-level variable mapped to the Directory property

1. Click the background of the **Control Flow** tab in SSIS Designer. This sets the scope for the variable you will create to the package.
2. On the SSIS menu, select **Variables**.
3. In the **Variables** window, click the Add Variable icon.
4. In the **Name** box, type **varFolderName**.

### IMPORTANT

Variable names are case sensitive.

5. Verify that the **Scope** box shows the name of the package, Lesson 5.
6. Set the value of the **Data Type** box of the `varFolderName` variable to **String**.
7. Return to the **Control Flow** tab and double-click the **Foreach File in Folder** container.
8. On the **Collection** page of the **Foreach Loop Editor**, click **Expressions**, and then click the ellipsis button (...).
9. In the **Property Expressions Editor**, click in the **Property** list, and select **Directory**.
10. In the **Expression** box, click the ellipsis button(...).
11. In the **Expression Builder**, expand the Variables folder, and drag the variable **User::varFolderName** to the **Expression** box.
12. Click **OK** to exit the **Expression Builder**.
13. Click **OK** to exit the **Property Expressions Editor**.
14. Click **OK** to exit the **Foreach Loop Editor**.

## To enable package configurations

1. On the **Project Menu**, click **Convert to Package Deployment Model**.
2. Click **OK** on the warning prompt and, once the conversion is complete, click **OK** on the **Convert to Package Deployment Model** dialog.
3. Click the background of the **Control Flow** tab in SSIS Designer.

4. On the **SSIS** menu, click **Package Configurations**.
5. In the **Package Configurations Organizer** dialog box, select **Enable Package Configurations**, and then click **Add**.
6. On the welcome page of the Package Configuration Wizard, click **Next**.
7. On the **Select Configuration Type** page, verify that the **Configuration type** is set to **XML configuration file**.
8. On the **Select Configuration Type** page, click **Browse**.
9. By default, the **Select Configuration File Location** dialog box will open to the project folder.
10. In the **Select Configuration File Location** dialog box, for **File name** type **SSISTutorial**, and then click **Save**.
11. On the **Select Configuration Type** page, click **Next**.
12. On the **Select Properties to Export** page, in the **Objects** pane, expand **Variables**, expand **varFolderName**, expand **Properties**, and then select **Value**.
13. On the **Select Properties to Export** page, click **Next**.
14. On the **Completing the Wizard** page, type a configuration name for the configuration, such as **SSIS Tutorial Directory configuration**. This is the configuration name that is displayed in the **Package Configuration Organizer** dialog box.
15. Click **Finish**.
16. Click **Close**.
17. The wizard creates a configuration file, named SSISTutorial.dtsConfig, that contains configuration settings for the **value** of the variable that in turn sets the **Directory** property of the enumerator.

#### NOTE

A configuration file typically contains complex information about the package properties, but for this tutorial the only configuration information should be

```
<\ConfiguredValue>  
<\Configuration>.
```

#### To create and populate a new sample data folder

1. In Windows Explorer, at the root level of your drive (for example, C:\), create a new folder named **New Sample Data**.
2. Locate the sample files on your computer and copy three of the files from the folder.
3. In the **New Sample Data** folder, paste the copied files.

## Next Task in Lesson

[Step 3: Modifying the Directory Property Configuration Value](#)

# Lesson 5-3 - Modifying the Directory Property Configuration Value

3/24/2017 • 1 min to read • [Edit Online](#)

In this task, you will modify the configuration setting, stored in the SSISTutorial.dtsConfig file, for the Value property of the package-level variable `User::varFolderName`. The variable updates the Directory property of the Foreach Loop container. The modified value will point to the **New Sample Data** folder that you created in the previous task. After you modify the configuration setting and run the package, the Directory property will be updated by the variable, using the value populated from the configuration file instead of the directory value originally configured in the package.

## To modify the configuration setting of the Directory property

1. In Notepad or any other text editor, locate and open the SSISTutorial.dtsConfig configuration file that you created by using the Package Configuration Wizard in the previous task.
2. Change the value of the **ConfiguredValue** element to match the path of the **New Sample Data** folder that you created in the previous task. Do not surround the path in quotes. If the **New Sample Data** folder is at the root level of your drive (for example, C:\), the updated XML should be similar to the following sample:

```
<?xml version="1.0"?><DTSConfiguration><DTSConfigurationHeading><DTSConfigurationFileInfo  
GeneratedBy="DOMAIN\UserName" GeneratedFromPackageName="Lesson 5" GeneratedFromPackageID="{F4475E73-59E3-  
478F-8EB2-B10AFA61D3FA}" GeneratedDate="6/10/2012 8:16:50 AM"/></DTSConfigurationHeading><Configuration  
ConfiguredType="Property" Path="\Package.Variables[User::varFolderName].Properties[Value]"  
ValueType="String"><ConfiguredValue></ConfiguredValue></Configuration></DTSConfiguration>
```

The heading information, **GeneratedBy**, **GeneratedFromPackageID**, and **GeneratedDate** will be different in your file, of course. The element to note is the **Configuration** element. The **Value** property of the variable, `User::varFolderName`, now contains C:\New Sample Data.

3. Save the change, and then close the text editor.

## Next Task in Lesson

[Step 4: Testing the Lesson 5 Tutorial Package](#)

# Lesson 5-4 - Testing the Lesson 5 Tutorial Package

3/24/2017 • 1 min to read • [Edit Online](#)

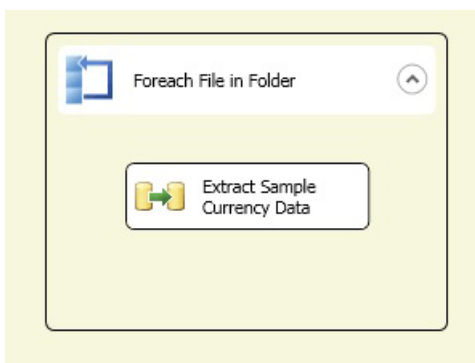
At run time, your package will obtain the value for the **Directory** property from a variable updated at run time, rather than using the original directory name that you specified when you created the package. The value of the variable is populated by the SSISTutorial.dtsConfig file.

To verify that the package updates the Directory property with the new value during run time, simply execute the package. Because only three sample data files were copied to the new directory, the data flow will run only three times, rather than iterate through the 14 files in the original folder.

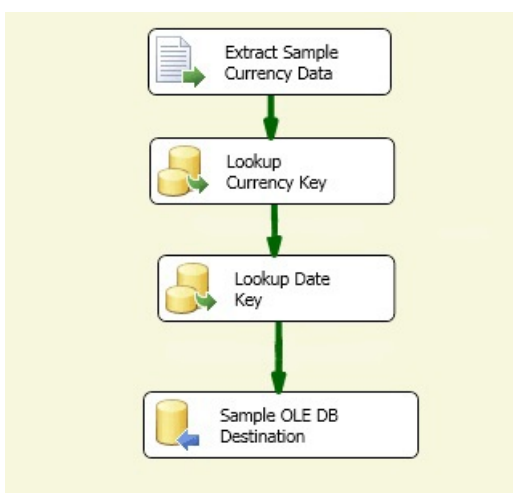
## Checking the Package Layout

Before you test the package you should verify that the control and data flows in the Lesson 5 package contains the objects shown in the following diagrams. The control flow should be identical to the control flow in lesson 4. The data flow should be identical to the data flow in lessons 4.

### Control Flow



### Data Flow



### To test the Lesson 5 tutorial package

1. On the **Debug** menu, click **Start Debugging**.
2. After the package has completed running, on the **Debug** menu, and then click **Stop Debugging**.

## Next Lesson



# Lesson 6: Using Parameters with the Project Deployment Model in SSIS

3/24/2017 • 1 min to read • [Edit Online](#)

SQL Server 2012 introduces a new deployment model where you can deploy your projects to the Integration Services server. The Integration Services server enables you to manage and run packages, and to configure runtime values for packages.

In this lesson, you will modify the package that you created in [Lesson 5: Add SSIS Package Configurations for the Package Deployment Model](#) to use the Project Deployment Model. You replace the configuration value with a parameter to specify the sample data location. You can also copy the completed Lesson 5 package that is included with the tutorial.

Using the Integration Services Project Configuration Wizard, you will convert the project to the Project Deployment Model and use a Parameter rather than a configuration value to set the Directory property. This lesson partially covers the steps you would follow to convert existing SSIS packages to the new Project Deployment Model.

When you run the package again, the Integration Services service uses the parameter to populate the value of the variable, and the variable in turn updates the Directory property. As a result, the package iterates through the files in the new data folder specified by the parameter value rather than the folder that was set in the package configuration file.

## IMPORTANT

This tutorial requires the **AdventureWorksDW2012** sample database. For more information about how to install and deploy **AdventureWorksDW2012**, see [Considerations for Installing SQL Server Samples and Sample Databases](#).

## Lesson Tasks

This lesson contains the following tasks:

1. [Step 1: Copying the Lesson 5 Package](#)
2. [Step 2: Converting the Project to the Project Deployment Model](#)
3. [Step 3: Testing the Lesson 6 Package](#)
4. [Step 4: Deploying the Lesson 6 Package](#)

## Start the Lesson

[Step 1: Copying the Lesson 5 Package](#)

# Lesson 6-1 - Copying the Lesson 5 Package

3/24/2017 • 1 min to read • [Edit Online](#)

In this task, you will create a copy of the Lesson 5.dtsx package that you created in Lesson 5. Alternatively, you can add the completed lesson 5 package that is included with the tutorial to the project, and then copy it instead. You will use this new copy throughout the rest of Lesson 6.

## To copy the Lesson 5 package

1. If SQL Server Data Tools is not already open, click Start, point to All Programs, point to Microsoft SQL Server 2012, and then click SQL Server Data Tools.
2. On the File menu, click Open, click Project/Solution, select SSIS Tutorial and click Open, and then double-click SSIS Tutorial.sln.
3. In Solution Explorer, right-click Lesson 5.dtsx, and then click Copy.
4. In Solution Explorer, right-click SSIS Packages, and then click Paste.

By default, the copied package is named Lesson 6.dtsx.

5. In the Solution Explorer, double-click Lesson 6.dtsx to open the package.
6. Right-click anywhere in the background of the Control Flow tab then click Properties.
7. In the Properties window, update the Name property to Lesson 6.
8. Click the box for the ID property, then click the dropdown arrow, and then click .

## To add the completed Lesson 5 package

1. Open SQL Server Data Tools and open the SSIS Tutorial project.
2. In Solution Explorer, right-click SSIS Packages, and click Add Existing Package.
3. In the Add Copy of Existing Package dialog box, in Package location, select File system.
4. Click the browse (...) button, Lesson 5.dtsx on your machine, and then click **Open**.

To download all of the lesson packages for this tutorial, do the following.

- a. Navigate to [Integration Services Product Samples](#)
  - b. Click the **DOWNLOADS** tab.
  - c. Click the SQL2012.Integration\_Services.Create\_Simple\_ETL\_Tutorial.Sample.zip file.
5. Copy and paste the Lesson 5 package as described in steps 3-8 in the previous procedure.

After copying the Lesson 5 package, if you currently have the packages from the previous lessons in your solution, right-click each package from lessons 1-5 and click Exclude From Project. When done you should have only Lesson 6.dtsx in your solution.

## Next Task in Lesson

[Step 2: Converting the Project to the Project Deployment Model](#)

# Lesson 6-2 - Converting the Project to the Project Deployment Model

3/24/2017 • 1 min to read • [Edit Online](#)

In this task, you will use the Integration Services Project Conversion Wizard to convert the project to the Project Deployment Model.

## Converting the Project to the Project Deployment Model

1. On the Project Menu, click Convert to Project Deployment Model.
2. On the Integration Services Project Conversion Wizard Introduction page, review the steps then click Next.
3. On the Select Packages page, in the Packages list, clear all checkboxes except Lesson 6.dtsx then click Next.
4. On the Specify Project Properties page, click Next.
5. On the Update Execute Package Task page click Next.
6. On the Select Configurations page, make sure the Lesson 6.dtsx package is selected in the Configurations list, then click Next.
7. On the Create Parameters page make sure the Lesson 6.dtsx package is selected, and the Scope is set to Package, in the Configuration Properties List, then Click Next.
8. On the Configure Parameters page verify that the values for Name and Value are the same name and value specified in Lesson 5 for the variable and configuration value, then click Next.
9. On the Review page, in the Summary pane, notice that the wizard has used the information from the configuration file to set the Properties to be converted.
10. Click Convert.

When the conversion completes a message is displayed warning that the changes are not saved until the project is saved in Visual Studio. Click OK on the warning dialog.

11. On the Integration Services Project Conversion Wizard click Close.
12. In SQL Server Data Tools, click the File menu, then click Save to save the converted package.
13. Click the Parameters Tab and verify that the package now contains a parameter for VarFolderName and that the value is the same path specified for the New Sample Data folder from the Lesson 5 configuration file.

## Next Task in Lesson

[Step 3: Testing the Lesson 6 Package](#)



# Lesson 6-3 - Testing the Lesson 6 Package

3/24/2017 • 1 min to read • [Edit Online](#)

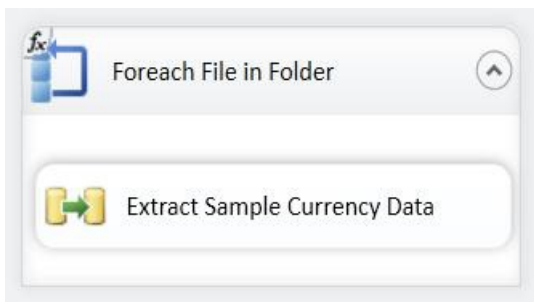
At run time, your package will obtain the value for the Directory property from the VarFolderName parameter.

To verify that the package updates the Directory property with the new value during run time, simply execute the package. Because only three sample data files were copied to the new directory, the data flow will run only three times, rather than iterate through the 14 files in the original folder.

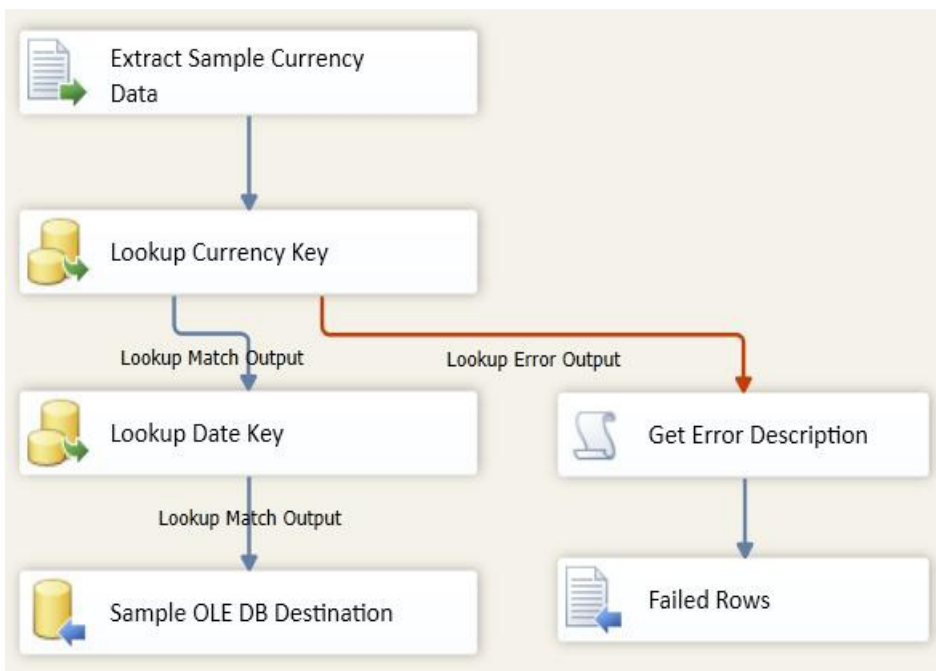
## Checking the Package Layout

Before you test the package you should verify that the control and data flows in the Lesson 6 package contains the objects shown in the following diagrams. The control flow should be identical to the control flow in lesson 5. The data flow should be identical to the data flow in lesson 5.

### Control Flow



### Data Flow



### TO test the Lesson 6 tutorial package

1. On the Debug menu, click Start Debugging.
2. After the package has completed running, on the Debug menu, and then click Stop Debugging.

## Next Task in Lesson

[Step 4: Deploying the Lesson 6 Package](#)

# Lesson 6-4 - Deploying the Lesson 6 Package

3/24/2017 • 4 min to read • [Edit Online](#)

Deploying the package involves adding the package to the SSISDB catalog in Integration Services on an instance of SQL Server. In this lesson you will add the Lesson 6 package to the SSISDB catalog, set the parameter, and execute the package. For this lesson you will use SQL Server Management Studio to add the Lesson 6 package to the SSISDB catalog, and deploy the package. After deploying the package you will modify the parameter to point to a new location then execute the package.

In this lesson you will:

- Add the package to the SSISDB catalog in the SSIS node in SQL Server.
- Deploy the package.
- Set the package parameter value.
- Execute the package in SSMS.

## **To Locate or add the the SSISDB catalog**

1. Click Start, point to All Programs, point to Microsoft SQL Server 2012, and then click SQL Management Studio.
2. On the Connect to Server dialog box, verify the default settings, and then click Connect. To connect, the Server name box must contain the name of the computer where SQL Server is installed. If the Database Engine is a named instance, the Server name box should also contain the instance name in the format \.
3. In Object Explorer expand Integration Services Catalogs.
4. If there are no catalogs listed under Integration Services Catalogs then add the SSISDB catalog.
5. To Add the SSISDB catalog, right-click Integration Services Catalogs and click Create Catalog.
6. On the Create Catalog dialog box select Enable CLR Integration.
7. In the Password box, type a new password then type it again in the Retype Password box. Be sure to remember the password you type.
8. Click OK to add the SSISDB catalog.

## **To add the package to the SSISDB catalog**

1. In Object Explorer, right-click SSISDB and click Create Folder.
2. In the Create Folder dialog box type SSIS Tutorial in the Folder name box and click OK.
3. Expand the SSIS Tutorial folder, right-click Projects, and click Import Packages.
4. On the Integration Services Project Conversion Wizard Introduction page click Next.
5. On the Locate Packages page, ensure that File system is selected in the Source list, then click Browse.
6. On the Browse For Folder dialog box, browse to the folder containing the SSIS Tutorial project, then click OK.
7. Click Next.
8. On the Select Packages page you should see all six packages from the SSIS Tutorial. In the Packages list, select Lesson 6.dtsx, then click Next.

9. On the Select Destination page, type SSIS Tutorial Deployment in the Project Name box then click Next.
10. Click Next on each of the remaining wizard pages until you get to the Review page.
11. On the Review page, click Convert.
12. When the conversion completes, click Close.

When you close the Integration Services Project Conversion Wizard, SSIS displays the Integration Services Deployment Wizard. You will use this wizard now to deploy the Lesson 6 package.

1. On the Integration Services Deployment Wizard Introduction page, review the steps for deploying the project, then click Next.
2. On the Select Destination page verify that the server name is the instance of SQL Server containing the SSISDB catalog and that the path shows SSIS Tutorial Deployment, then click Next.
3. On the Review page, review the Summary then click Deploy.
4. When the deployment completes, click Close.
5. In Object Explorer, right-click Integration Services Catalogs and click Refresh.
6. Expand Integration Services Catalogs then expand SSISDB. Continue to Expand the tree under SSIS Tutorial until you have completely expanded the project. You should see Lesson 6.dtsx under the Packages node of the SSIS Tutorial Deployment node.

To verify that the package is complete, right-click Lesson 6.dtsx and click Configure. On the Configure dialog box, select Parameters and verify that there is an entry with Lesson 6.dtsx as the Container, VarFolderName as the Name and the path to New Sample Data as the value, then click Close.

Before continuing create a new sample data folder, name it Sample Data Two, and copy any three of the original sample files into it.

#### **To create and populate a new sample data folder**

1. In Windows Explorer, at the root level of your drive (for example, C:\), create a new folder named Sample Data Two.
2. Open the c:\Program Files\Microsoft SQL Server\110\Samples\Integration Services\Tutorial\Creating a Simple ETL Package\Sample Data folder and then copy any three of the sample files from the folder.
3. In the New Sample Data folder, paste the copied files.

#### **To change the package parameter to point to the new sample data**

1. In Object Explorer, right click Lesson 6.dtsx and click Configure.
2. On the Configure dialog box, change the parameter value to the path to Sample Data Two. For example C:\Sample Data Two if you placed the new folder in the root folder on the C drive.
3. Click OK to close the Configure dialog box.

#### **To test the Lesson 6 package deployment**

1. In Object Explorer, right click Lesson 6.dtsx and click Execute.
2. On the Execute Package dialog box, click OK.
3. On the message dialog box click Yes to open Overview Report.

The Overview report for the package is displayed showing the name of the package and a status summary. The Execution Overview section shows the result from each task in the package and the Parameters Used section shows the names and values of all parameters used in the package execution, including VarFolderName.

# Deploy Packages with SSIS

3/24/2017 • 4 min to read • [Edit Online](#)

Microsoft SQL Server Integration Services provides tools that make it easy to deploy packages to another computer. The deployment tools also manage any dependencies, such as configurations and files that the package needs. In this tutorial, you will learn how to use these tools to install packages and their dependencies on a target computer.

First, you will perform tasks to prepare for deployment. You will create a new Integration Services project in SQL Server Data Tools (SSDT) and add existing packages and data files to the project. You will not create any new packages from scratch; instead, you will work only with completed packages that were created just for this tutorial. You will not modify the functionality of the packages in this tutorial; however, after you have added the packages to the project, you might find it useful to open the packages in SSIS Designer and review the contents of each package. By examining the packages, you will learn about package dependencies such as log files and about other interesting features of the packages.

In preparation for deployment, you will also update the packages to use configurations. Configurations make the properties of packages and package objects updatable at run time. In this tutorial, you will use configurations to update the connection strings of log and text files and the locations of the XML and XSD files that the package uses. For more information, see [Package Configurations](#) and [Create Package Configurations](#).

After you have verified that the packages run successfully in SQL Server Data Tools (SSDT), you will create the deployment bundle to use to install the packages. The deployment bundle will consist of the package files and other items that you added to the Integration Services project, the package dependencies that Integration Services automatically includes, and the deployment utility that you built. For more information, see [Create a Deployment Utility](#).

You will then copy the deployment bundle to the target computer and run the Package Installation Wizard to install the packages and package dependencies. The packages will be installed in the msdb SQL Server database, and the supporting and ancillary files will be installed in the file system. Because the deployed packages use configurations, you will update the configuration to use new values that enable packages to run successfully in the new environment.

Finally, you will run the packages in SQL Server Management Studio by using the Execute Package Utility.

It is the goal of this tutorial to simulate the complexity of real-life deployment issues that you may encounter. However, if it is not possible for you to deploy the packages to a different computer, you can still do this tutorial by installing the packages in the msdb database on a local instance of SQL Server, and then running the packages from SQL Server Management Studio on the same instance.

## What You Will Learn

The best way to become acquainted with the new tools, controls, and features available in Microsoft SQL Server Integration Services is to use them. This tutorial walks you through the steps to create an Integration Services project and then add the packages and other necessary files to the project. After the project is complete, you will create a deployment bundle, copy the bundle to the destination computer, and then install the packages on the destination computer.

## Requirements

This tutorial is intended for users who are already familiar with fundamental file system operations, but who have

limited exposure to the new features available in SQL Server Integration Services. To better understand basic Integration Services concepts that you will put to use in this tutorial, you might find it useful to first complete the following Integration Services tutorial: [SSIS How to Create an ETL Package](#).

**Source computer.** The computer on which you will create the deployment bundle **must have the following components installed:**

- SQL Server
- Sample data, completed packages, configurations, and a Readme. These files are installed together if you download the [Adventure Works 2014 Sample Databases](#).

**Note!** Make sure you have permission to create and drop tables in AdventureWorks or other data you use.

- [SQL Server Data Tools \(SSDT\)](#).

**Destination computer.** The computer to which you deploy packages **must have the following components installed:**

- SQL Server
- Sample data, completed packages, configurations, and a Readme. These files are installed together if you download the [Adventure Works 2014 Sample Databases](#).
- [SQL Server Management Studio](#).
- SQL Server Integration Services.
- You must have permission to create and drop tables in AdventureWorks and to run packages in SQL Server Management Studio.
- You must have read and write permission on the sysssispackages table in the msdb SQL Server system database.

If you plan to deploy packages to the same computer as the one on which you create the deployment bundle, that computer must meet requirements for both the source and destination computers.

**Estimated time to complete this tutorial:** 2 hours

## Lessons in This Tutorial

### [Lesson 1: Preparing to Create the Deployment Bundle](#)

In this lesson, you will get ready to deploy an ETL solution by creating a new Integration Services project and adding the packages and other required files to the project.

### [Lesson 2: Create the Deployment Bundle in SSIS](#)

In this lesson, you will build a deployment utility and verify that the deployment bundle includes the necessary files.

### [Lesson 3: Install SSIS Packages](#)

In this lesson, you will copy the deployment bundle to the target computer, install the packages, and then run the packages.

# Lesson 1: Preparing to Create the Deployment Bundle

3/24/2017 • 1 min to read • [Edit Online](#)

In this lesson, you will create the working folders and environment variables that support the tutorial, create an Integration Services project, add several packages and their supporting files to the project, and implement configurations in packages.

Integration Services deploys packages on a project basis; therefore, as the first step in creating the deployment bundle, you must collect all the packages and package dependencies into one Integration Services project. Frequently it is useful to include other information with the deployed packages: for example you will also add to the project a Readme file that provides basic documentation for this group of packages.

After you have added the packages and files, you will add configurations to packages that do not already use configurations. The configurations update properties of packages and package objects at run time. In a later lesson, you will modify the values of these configurations during package deployment to support the packages in the deployed-to environment.

After you have added the configurations, you should open the packages in SSIS Designer, the Integration Services graphical tool for building ETL packages, and examine the properties of packages and package elements as well as the package configurations to better understand the issues that the deployment needs to address. For example, one of the packages extracts data from text files, so the location of the data files must be updated before the deployed packages will run successfully.

**Estimated time to complete this lesson:** 1 hour

## Lesson Tasks

This lesson contains the following tasks:

- [Step 1: Creating Working Folders and Environment Variables](#)
- [Step 2: Creating the Deployment Project](#)
- [Step 3: Adding Packages and Other Files](#)
- [Step 4: Adding Package Configurations](#)
- [Step 5: Testing the Updated Packages](#)

## Start the Lesson

[Step 1: Creating Working Folders and Environment Variables](#)

# Lesson 1-1 - Creating Working Folders and Environment Variables

3/24/2017 • 3 min to read • [Edit Online](#)

In this task, you will create the working folder (C:\DeploymentTutorial) and the new system environment variables ( `DataTransfer` and `LoadXMLData` ) that you will use in later tutorial tasks.

The working folder is at the root of the C drive. If you must use a different drive or location, you can do that. However, you need to note this location and then use it wherever the tutorial refers to the location of the DeploymentTutorial working folder.

In a later lesson, you will deploy packages that are saved to the file system to the sysssispackages table in the msdb SQL Server database. Ideally you will deploy the Integration Services packages to a different computer. If that is not possible, you can still learn a lot from doing this tutorial by deploying the packages to an instance of SQL Server that is on the local computer. The environment variables that are used on the local and destination computers have the same variable names, but different values are stored in the variables. For example, on the local computer, the value of the environment variable `DataTransfer` references the C:\DeploymentTutorial folder, whereas on the target computer the environment variable `DataTransfer` references the C:\DeploymentTutorialInstall folder.

If you plan to deploy to the local computer, you need to create only one set of environment variables; however, you will need to update the value of the environment variables to an appropriate value before you do the local deployment.

If you plan to deploy the packages to a different computer, you must create two sets of environment variables: one set for the local computer, and one set for the destination computer. You can create only the variables for the source computer now, and create the variables for the destination computer later, but you must have both the folder and environment variables available on the destination computer before you can install the packages on that computer.

## To create the local working folder

1. Right-click the Start menu, and click Explore.
2. Click **Local Disk (C:)**.
3. On the **File** menu, point to **New**, and then click **Folder**.
4. Rename New Folder to **DeploymentTutorial**.

## To create local environment variables

1. On the **Start** menu, click **Control Panel**.
2. In Control Panel, double-click **System**.
3. In the **System Properties** dialog box, click the **Advanced** tab, and then click **Environment Variables**.
4. In the **Environment Variables** dialog box, in the **System variables** frame, click **New**.
5. In the **New System Variable** dialog box, type **DataTransfer** in the **Variable name** box, and **C:\DeploymentTutorial\datatransferconfig.dtsconfig** in the **Variable value** box.
6. Click **OK**.
7. Click **New** again, and type **LoadXMLData** in the **Variable name** box, and **C:\DeploymentTutorial\loadxmldataconfig.dtsconfig** in the **Variable value** box.



8. Click **OK** to exit the **Environment Variables** dialog box.
9. Click **OK** to exit the **System Properties** dialog box.\
10. Optionally, restart your computer. If you do not restart the computer, the name of the new variable will not be displayed in the Package Configuration Wizard, but you can still use it.

#### **To create destination environment variables**

1. On the **Start** menu, click **Control Panel**.
2. In Control Panel, double-click **System**.
3. In the **System Properties** dialog box, click the **Advanced** tab, and then click **Environment Variables**.
4. In the **Environment Variables** dialog box, in **System variables** frame, click **New**.
5. In the **New System Variables** dialog box, type **DataTransfer** in the **Variable name** box, and **C:\DeploymentTutorialInstall\datatransferconfig.dtsconfig** in the **Variable value** box.
6. Click **OK**.
7. Click **New** again, and type **LoadXMLData** in the **Variable name** box, and **C:\DeploymentTutorialInstall\loadxmldataconfig.dtsconfig** in the **Variable value** box.
8. Click **OK** to exit the **Environment Variables** dialog box.
9. Click **OK** to exit the **System Properties** dialog box.\
10. Optionally, restart your computer.

## Next Task in Lesson

[Step 2: Creating the Deployment Project](#)

# Lesson 1-2 - Creating the Deployment Project

3/24/2017 • 1 min to read • [Edit Online](#)

In Integration Services, the deployable unit is an Integration Services project. Before you can deploy packages, you must create a new Integration Services project and add all the packages and any ancillary files that you want to deploy with the packages to that project.

## To create the Integration Services project

1. Click **Start**, point to **All Programs**, point to **Microsoft SQL Server**, and then click **SQL ServerSQL Server Data Tools**.
2. On the **File** menu, point to **New**, and then click **Project** to create a new Integration Services project.
3. In the **New Project** dialog box, select **Integration Services Project** in the **Templates** pane.
4. In the **Name** box, change the default name to **Deployment Tutorial**. Optionally, clear the **Create directory for solution** check box.
5. Accept the default location, or click **Browse** to locate the folder you want to use.
6. In the **Project Location** dialog box, click the folder, and then click **Open**.
7. Click **OK**.
8. By default, an empty package, named Package.dtsx, is created and added to your project. However, you will not use this package; instead, you will add existing packages to the project. Because all the packages in a project will be included in the deployment, you should delete Package.dtsx. To delete it, right-click it, and then click **Delete**.

## Next Task in Lesson

[Step 3: Adding Packages and Other Files](#)

## See Also

[Integration Services \(SSIS\) Projects](#)

# Lesson 1-3 - Adding Packages and Other Files

3/24/2017 • 2 min to read • [Edit Online](#)

In this task, you will add existing packages, ancillary files that support individual packages, and a Readme to the Deployment Tutorial project that you created in the previous task. For example, you will add an XML data file that contains the data for a package and a text file that provides Readme information about all the packages in the project.

When you deploy packages to a test or production environment, you typically do not include the data files in the deployment, but instead use configurations to update the paths of the data sources to access test or production versions of the data files or databases. For instructional purposes, this tutorial includes data files in the package deployment.

The packages and the sample data that the packages use are installed when you install the SQL Server samples. You will add the following packages to the Deployment Tutorial project:

- **DataTransfer.** Basic package that extracts data from a flat file, evaluates column values to conditionally keep rows in the dataset, and loads data into a table in the AdventureWorks database.
- **LoadXMLData.** Data-transfer package that extracts data from an XML data file, evaluates and aggregates column values, and loads data into a table in the AdventureWorks database.

To support the deployment of these packages, you will add the following ancillary files to the Deployment Tutorial project.

PACKAGE	FILE
DataTransfer	NewCustomers.txt
LoadXMLData	orders.xml and orders.xsd

You will also add a Readme, which is a text file that provides information about the Deployment Tutorial project.

The paths used in the following procedures assume that the SQL Server samples were installed in the default location, C:\Program Files\Microsoft SQL Server\120\Samples\Integration Services\. If you installed the samples to a different location, you should use that location instead in the procedures.

In the next task, you will add configurations to the DataTransfer and LoadXMLData packages. All configurations are stored in XML files, and you will use a system environment variable to specify the location of the files. After you create the configuration files, you will add them to the project.

## To add packages to the Deployment Tutorial project

1. If SQL Server Data Tools (SSDT) is not already open, click **Start**, point to **All Programs**, point to **Microsoft SQL Server**, and then click **SQL Server Data Tools**.
2. On the **File** menu, click **Open**, click **Project/Solution**, click the **Deployment Tutorial** folder and click **Open**, and then double-click **Deployment Tutorial.sln**.
3. In Solution Explorer, right-click Deployment Tutorial, click **Add**, and then click **Existing Package**.
4. In the **Add Copy of Existing Package** dialog box, in **Package location**, select **File System**.
5. Click the browse (...) button, navigate to C:\Program Files\Microsoft SQL Server\100\Samples\Integration ServicesTutorial\Deploying Packages\Completed Packages, select **DataTransfer.dtsx**, and then click **Open**.

6. Click **OK**.
7. Repeat steps 3 - 6, and this time add LoadXMLData.dtsx, which is found in C:\Program Files\Microsoft SQL Server\100\Samples\Integration Services\Tutorial\Deploying Packages\Completed Packages.

#### **To add ancillary files to the Deployment Tutorial project**

1. In Solution Explorer, right-click Deployment Tutorial, click **Add**, and then click **Existing Item**.
2. In the **Add Existing Item - Deployment Tutorial** dialog box, navigate to C:\Program Files\Microsoft SQL Server\100\Samples\Integration Services\Tutorial\Deployment Packages\Sample Data, select orders.xml, orders.xsd, and NewCustomers.txt, and then click **Add**.
3. In the **Add Existing Item - Deployment Tutorial** dialog box, navigate to C:\Program Files\Microsoft SQL Server\100\Samples\Integration Services\Tutorial\Deployment Packages\, select Readme.txt and click **Add**.
4. On the File menu, click **Save All**.

## Next Task in Lesson

[Step 4: Adding Package Configurations](#)

# Lesson 1-4 - Adding Package Configurations

4/19/2017 • 5 min to read • [Edit Online](#)

In this task, you will add a configuration to each package. Configurations update the values of package properties and package objects at run time.

Integration Services provides a variety of configuration types. You can store configurations in environment variables, registry entries, user-defined variables, SQL Server tables, and XML files. To provide additional flexibility, Integration Services supports the use of indirect configurations. This means that you use an environment variable to specify the location of the configuration, which in turn specifies the actual values. The packages in the Deployment Tutorial project use a combination of XML configuration files and indirect configurations. An XML configuration file can include configurations for multiple properties, and when appropriate, can be referenced by multiple packages. In this tutorial, you will use a separate configuration file for each package.

Configuration files frequently contain sensitive information such as connection strings. Therefore, you should use an access control list (ACL) to restrict access to the location or folder where you store the files, and give access only to users or accounts that are permitted to run packages. For more information, see [Access to Files Used by Packages](#).

The packages (DataTransfer and LoadXMLData) that you added to the Deployment Tutorial project in the previous task need configurations to run successfully after they are deployed to the target server. To implement configurations, you will first create the indirect configurations for the XML configuration files, and then you will create the XML configuration files.

You will create two configuration files, DataTransferConfig.dtsConfig and LoadXMLData.dtsConfig. These files contain name-value pairs that update the properties in packages that specify the location of the data and log files used by the package. Later, as a step in the deployment process, you will update the values in the configuration files to reflect the new location of the files on the destination computer.

Integration Services recognizes that the DataTransferConfig.dtsConfig and LoadXMLData.dtsConfig are dependencies of the DataTransfer and LoadXMLData packages, and automatically includes the configuration files when you create the deployment bundle in the next lesson.

## To create indirect configuration for the DataTransfer package

1. In Solution Explorer, double-click DataTransfer.dtsx.
2. In SSIS Designer, click anywhere in the background of the control flow design surface.
3. On the **SSIS** menu, click **Package Configurations**.
4. In the **Package Configuration Organizer** dialog box, select **Enable package configurations** if it is not already selected, and then click **Add**.
5. On the welcome page of the Package Configuration Wizard, click **Next**.
6. On the Select Configuration Type page, select **XML configuration file** in the **Configuration type** list, select the **Configuration location is stored in an environment variable** option, and type **DataTransfer**, or select the **DataTransfer** environment variable in the list.

### NOTE

To make the environment variable available in the list, you may have to restart your computer after adding the variable. If you do not want to restart the computer, you can type the name of the environment variable.

7. Click **Next**.
8. On the Completing the Wizard page, type **DataTransfer EV Configuration** in the **Configuration name** box, review the configuration contents in the **Preview** pane, and then click **Finish**.
9. Close the **Package Configuration Organizer** dialog box.

#### To create the XML configuration for the DataTransfer package

1. In Solution Explorer, double-click DataTransfer.dtsx.
2. In SSIS Designer, click anywhere in the background of the control flow design surface.
3. On the **SSIS** menu, click **Package Configurations**.
4. In the Package Configuration Organizer dialog box, select the **Enable Package Configurations** check-box, and then click **Add**.
5. On the welcome page of the Package Configuration Wizard, click **Next**.
6. On the Select Configuration Type page, select **XML configuration file** in the **Configuration type** list and then click **Browse**.
7. In **Select Configuration File Location** dialog box, navigate to C:\DeploymentTutorial and type **DataTransferConfig** in the **File name** box, and then click **Save**.
8. On the Select Configuration Type page, click **Next**.
9. On the Select Properties to Export page, expand DataTransfer, Connection Managers, Deployment Tutorial Log, and Properties, and then select the **Connection String** check-box.
10. Within Connection Managers, expand NewCustomers, and then select the **Connection String** check-box.
11. Click **Next**.
12. On the Completing the Wizard page, type **DataTransfer Configuration** in the **Configuration name** box, review the content of the configuration, and then click **Finish**.
13. In the **Package Configuration Organizer** dialog box, verify that DataTransfer EV Configuration is listed first, and DataTransfer Configuration is listed second, and then click **Close**.

#### To create indirect configuration for the LoadXMLData package

1. In Solution Explorer, double-click LoadXMLData.dtsx.
2. In SSIS Designer, click anywhere in the background of the control flow design surface.
3. On the **SSIS** menu, click **Package Configurations**.
4. In the **Package Configuration Organizer** dialog box, Click **Add**.
5. On the welcome page of the Package Configuration Wizard, click **Next**.
6. On the Select Configuration Type page, select **XML configuration file** in the **Configuration type** list, select the **Configuration location is stored in an environment variable** option, type **LoadXMLData** or select the **LoadXMLData** environment variable in the list.

#### NOTE

To make the environment variable available in the list, you may have to restart your computer after adding the variable.

7. Click **Next**.

8. On the Completing the Wizard page, type **LoadXMLData EV Configuration** in the **Configuration name** box, review the content of the configuration, and then click **Finish**.

#### To create the XML configuration for the LoadXMLData package

1. In Solution Explorer, double-click LoadXMLData.dtsx.
2. In SSIS Designer, click anywhere in the background of the control flow design surface.
3. On the **SSIS** menu, click **Package Configurations**.
4. In the Package Configuration Organizer dialog box, select the **Enable Package Configurations** check-box, and click **Add**.
5. On the welcome page of the Package Configuration Wizard, click **Next**.
6. On the Select Configuration Type page, select **XML configuration file** in the **Configuration type** list and click **Browse**.
7. In **Select Configuration File Location** dialog box, navigate to C:\DeploymentTutorial and type **LoadXMLDataConfig** in the **File name** box, and then click **Save**.
8. On the Select Configuration Type page, click **Next**.
9. On the Select Properties to Export page, expand LoadXMLData, Executables, Load XML Data, and Properties, and then select the **[XMLSource].[XMLData]** and **[XMLSource].[XMLSchemaDefinition]** check boxes.
10. Click **Next**.
11. On the Completing the Wizard page, type **LoadXMLData Configuration** in the **Configuration name** box, review the content of the configuration, and then click **Finish**.
12. In the **Package Configuration Organizer** dialog box, verify that the LoadXMLData EV Configuration is listed first, and the LoadXMLData Configuration is listed second, and then click **Close**.

## Next Task in Lesson

[Step 5: Testing the Updated Packages](#)

## See Also

[Package Configurations](#)

[Create Package Configurations](#)

[Access to Files Used by Packages](#)

# Lesson 1-5 - Testing the Updated Packages

3/24/2017 • 1 min to read • [Edit Online](#)

Before you go on to the next lesson, in which you will create the deployment bundle to use to install the tutorial packages on the destination computer, you should test the packages. In this task, you will run the packages, DataTransfer.dtsx and LoadXMLData, that you added to the Deployment Tutorial project and then extended with configurations.

When the packages run, each executable in the package becomes a green color as it completes successfully. When all executables are green, the package has completed successfully. You can also view the package execution progress on the **Progress** tab.

If the packages do not run successfully, you must fix them before you go on to the next lesson.

## To run the DataTransfer package

1. In Solution Explorer, click DataTransfer.dtsx.
2. On **Debug** menu, click **Start Debugging**.
3. After the package has completed running, on the **Debug** menu, click **Stop Debugging**.

## To run the LoadXMLData package

1. In Solution Explorer, click LoadXMLData.dtsx.
2. On **Debug** menu, click **Start Debugging**.
3. After the package has completed running, on the **Debug** menu, click **Stop Debugging**.

## Next Lesson

[Lesson 2: Create the Deployment Bundle in SSIS](#)



# Lesson 2: Create the Deployment Bundle in SSIS

3/24/2017 • 1 min to read • [Edit Online](#)

In [Lesson 1: Preparing to Create the Deployment Bundle](#), you created the Integration Services project named Deployment Tutorial, added the packages and supporting files to the project, and implemented configurations in packages.

In this lesson, you will create the deployment bundle, which is a folder that contains the items that you need to install packages on another computer. The deployment bundle will include a deployment manifest, copies of the packages, and copies of the supporting files from the Deployment Tutorial project. The deployment manifest lists the packages, miscellaneous files, and configurations in the deployment bundle.

You will also verify the file list in the deployment bundle and examine the contents of the manifest.

**Estimated time to complete this lesson:** 30 minutes

## Lesson Tasks

This lesson contains the following tasks:

- [Step 1: Building the Deployment Utility](#)
- [Step 2: Verifying the Deployment Bundle](#)

## Start the Lesson

[Step 1: Building the Deployment Utility](#)

# Lesson 2-1 - Building the Deployment Utility

3/24/2017 • 1 min to read • [Edit Online](#)

In this task, you will configure and build a deployment utility for the Deployment Tutorial project.

Before you can build the deployment utility, you must modify the properties of the Deployment Tutorial project. You will use the **Deployment Tutorial Property Pages** dialog box to configure these properties. In this dialog box, you must enable the ability to update configurations during deployment and specify that the build process generates a deployment utility. After you set the properties, you will build the project.

SQL Server Data Tools (SSDT) provides a set of windows that you can use to debug packages. You can view error, warning, and information messages, track about the status of packages at run time, or view the progress and results of build processes. For this lesson, you will use the Output window to view the progress and results of building the deployment utility.

## To set the deployment utility properties

1. If SQL Server Data Tools (SSDT) is not already open, click **Start**, point to **All Programs**, point to **Microsoft SQL Server**, and then click **Business Intelligence Development Studio**.
2. On the **File** menu, click **Open**, click **Project/Solution**, click the **Deployment Tutorial** folder and click **Open**, and then double-click **Deployment Tutorial.sln**.
3. In Solution Explorer, right-click Deployment Tutorial and click **Properties**.
4. In the **Deployment Tutorial Property Pages** dialog box, expand Configuration Properties, and click Deployment Utility.
5. In the right pane of the **Deployment Tutorial Property Pages** dialog box, verify that **AllowConfigurationChanges** is set to **true**, set **CreateDeploymentUtility** to **true**, and optionally update the default value of **DeploymentOutputPath**.
6. Click **OK**.

## To build the deployment utility

1. In Solution Explorer, click **Deployment Tutorial**.
2. On the **View** menu, click **Output**. By default, the Output window is located in the bottom left corner of SQL Server Data Tools (SSDT).
3. On the **Build** menu, click **Build Deployment Tutorial**.
4. In the Output window, verify the following information:

Build started: SQL Integration Services project: Incremental ...

Creating deployment utility...

Deployment Utility created.

Build complete -- 0 errors, 0 warnings

===== Build: 0 succeeded, 0 failed, 1 up-to-date, 0 skipped =====

5. On the **File** menu, click **Exit**. If prompted to save changes to Deployment Tutorial items, click **Yes**.

## Next Task in Lesson

[Step 2: Verifying the Deployment Bundle](#)

## See Also

[Create a Deployment Utility](#)

# Lesson 2-2 - Verifying the Deployment Bundle

3/24/2017 • 1 min to read • [Edit Online](#)

In lesson 1, you created the Deployment Tutorial project and added packages and ancillary files to the project; in the previous task you built a deployment utility for the project.

In this task, you will verify the contents of the deployment bundle. The deployment bundle is the folder that you will copy to the destination computer and use to install packages. If you used the default value—bin\Deployment—as the location of the deployment utility, the deployment bundle is the Bin\Deployment folder within the Deployment Tutorial folder in the Integration Services project.

## To verify the content of deployment bundle

1. Locate the bin\Deployment folder on your computer.
2. Verify that the following files are present:
  - DataTransfer.dtsx
  - datatransferconfig.dtsconfig
  - Deployment Tutorial.SSISDeploymentManifest
  - LoadXMLData.dtsx
  - loadxmldataconfig.dtsconfig
  - NewCustomers.txt
  - orders.xml
  - orders.xsd
  - Readme.txt
3. Right-click Deployment Tutorial.SSISDeploymentManifest, point **to Open With**, and then click **Internet Explorer**. You can also open the file in a text editor such as Notepad. The XML code of the file should be the following:

```
<?xml version="1.0"?><DTSDeploymentManifest GeneratedBy="Domain\UserName"
GeneratedFromProjectName="Deployment Tutorial" GeneratedDate="2006-02-24T13:29:02.6537669-08:00"
AllowConfigurationChanges="true"><Package>DataTransfer.dtsx</Package><Package>LoadXMLData.dtsx</Package>
<ConfigurationFile>datatransferconfig.dtsconfig</ConfigurationFile>
<ConfigurationFile>loadxmldataconfig.dtsconfig</ConfigurationFile>
<MiscellaneousFile>Readme.txt</MiscellaneousFile><MiscellaneousFile>orders.xml</MiscellaneousFile>
<MiscellaneousFile>NewCustomers.txt</MiscellaneousFile><MiscellaneousFile>orders.xsd</MiscellaneousFile>
</DTSDeploymentManifest>
```
4. Verify that the value of the **AllowConfigurationChanges** attribute is **true** and the XML includes a **Package** element for each of the two packages, a **MiscellaneousFile** element for each of the four non-package files, and a **ConfigurationFile** element for each of the two XML configuration files.
5. Exit Internet Explorer or the text editor.

## Next Lesson

[Lesson 3: Install SSIS Packages](#)

# Lesson 3: Install SSIS Packages

3/24/2017 • 1 min to read • [Edit Online](#)

In [Lesson 2: Create the Deployment Bundle in SSIS](#), you built a deployment utility and created the deployment bundle that contains the items that you must install packages on another computer. You also verified the file list in the deployment bundle and examined the contents of the manifest file that was created when you built the deployment utility.

In this lesson, you will copy the deployment bundle to the destination computer and then run the Package Installation Wizard to install the packages, package dependencies, and ancillary files on that computer. The packages will be installed in the **msdb** SQL Server database and the other items will be installed in the file system. After you complete the package installation, you will test the deployment by running the packages from SQL Server Management Studio using the Execute Package Utility.

**Estimated time to complete this lesson:** 30 minutes

## Lesson Tasks

This lesson contains the following tasks:

- [Step 1: Copying the Deployment Bundle](#)
- [Step 2: Running the Package Installation Wizard](#)
- [Step 3: Testing the Deployed Packages](#)

## Start the Lesson

[Step 1: Copying the Deployment Bundle](#)

# Lesson 3-1 - Copying the Deployment Bundle

3/24/2017 • 1 min to read • [Edit Online](#)

In this task, you will copy the deployment bundle to the destination computer.

The easiest way to copy the deployment bundle to the destination computer is to first create a public share on the destination computer, map a drive to the public share, and then copy the deployment bundle to the share. If you do not know how to create and configure public folders or map drives, see the Windows documentation.

## To copy the deployment bundle

1. Locate the deployment bundle on your computer.

If you used the default location, the deployment bundle is the Bin\Deployment folder within the Deployment Tutorial folder.

2. Right-click the Deployment folder and click **Copy**.
3. Locate the public share to which you want to copy the folder on the target computer and click **Paste**.

## Next Task in Lesson

[Step 2: Running the Package Installation Wizard](#)

# Lesson 3-2 - Running the Package Installation Wizard

4/10/2017 • 3 min to read • [Edit Online](#)

In this task, you will run the Package Installation Wizard to deploy the packages from the Deployment Tutorial project to an instance of SQL Server. Only packages can be installed in the sysssispackages table in the msdb SQL Server database, the supporting files that the deployment bundle includes will be deployed to the file system.

The Package Installation Wizard will guide you through the steps to install and configure the packages. You will install the packages to an instance of SQL Server on the destination computer (the computer to which you copied the deployment bundle). You will also create a folder, C:\DeploymentTutorialInstall, in which the wizard will install the non-package files.

In an earlier lesson, you modified the packages in the tutorial to use configurations. Using the Package Installation Wizard, you will edit these configurations to enable packages to run successfully in the installed-to environment.

## To install the packages

1. On the destination computer, locate the deployment bundle.

If you used the default value—bin\Deployment—as the location of the deployment utility, the deployment bundle is the Deployment folder in the Deployment Tutorial project.

2. In the Deployment folder, double-click the manifest file, Deployment Tutorial.SSISDeploymentManifest.
3. On the Welcome page of the Package Installation Wizard, click **Next**.
4. On the Deploy SSIS Packages page, select the **SQL Server deployment** option, select the **Validate packages after installation** check box, and then click **Next**.
5. On the Specify Target SQL Server page, specify **(local)**, in the **Server name** box.
6. If the instance of SQL Server supports Windows Authentication, select **Use Windows Authentication**; otherwise, select **Use SQL Server Authentication** and provide a user name and a password.
7. Verify that the **Rely on server storage for encryption** check box is cleared.
8. Click **Next**.
9. On the Select Installation Folder page, click **Browse**.
10. In the **Browse For Folder** dialog box, expand **My Computer** and then click **Local Disk (C:)**.
11. Click **Make New Folder** and replace the default name of the new folder, **New Folder**, with **DeploymentTutorialInstall**.

### IMPORTANT

This name is referenced in the value of the environment variables that configurations use. The name of the folder and the reference must match or the package cannot run.

12. Click **OK**.
13. On the Select Installation Folder page, verify that the Folder box contains **C:\DeploymentTutorialInstall** and then click **Next**.
14. On the Confirm Installation page, click **Next**.

The wizard installs the packages. After installation is completed, the Configure Packages page opens.

15. On the Configure Packages page, verify that the **Configuration file** box lists `datatransferconfig.dtsconfig` and `loadxmldataconfig.dtsconfig`.
16. In the **Configuration file** list, click `datatransferconfig.dtsconfig`, expand Property in the **Path** column of the **Configurations** box, and update the **Value** column with the following values:

PROPERTY	VALUE	UPDATED VALUE
\Package.Connections[Deployment Tutorial Log].Properties[ConnectionString]	C:\Program Files\Microsoft SQL Server\100\Samples\Integration Services\Tutorial\Deploying Packages\Completed Packages\Deployment Tutorial Log	C:\DeploymentTutorialInstall\Deployment Tutorial Log
\Package.Connections[NewCustomers].Properties[ConnectionString]	C:\Program Files\Microsoft SQL Server\100\Samples\Integration Services\Tutorial\Deploying Packages\Sample Data\NewCustomers.txt	C:\DeploymentTutorialInstall\NewCustomers.txt

17. In the **Configuration file** list, click `loadxmldataconfig.dtsconfig`, expand Property in the **Path** column of the **Configurations** box, and update the **Value** column with the following values:

PROPERTY	VALUE	UPDATED VALUE
\Package.LoadXMLData.Properties[[XML Source].[XMLData]]	C:\Program Files\Microsoft SQL Server\100\Samples\Integration Services\Tutorial\Deploying Packages\Sample Data\orders.xml	C:\DeploymentTutorialInstall\orders.xml
\Package.LoadXMLData.Properties[[XML Source].[XMLSchemaDefinition]]	C:\Program Files\Microsoft SQL Server\100\Samples\Integration Services\Tutorial\Deploying Packages\Sample Data\orders.xsd	C:\DeploymentTutorialInstall\orders.xsd

18. On the Package Validation page, view the validation results of each package installed and then click **Next**.

Because the values of the environment variables on the destination computer differ from the values of the environment variables on the development computer, several warnings appear on the Package Validation page. You should expect four warnings:

- The configuration file: "C:\DeploymentTutorial\DataTransferConfig.dtsConfig" is not valid. Check the configuration file name.
- Failed to load at least one of the configuration entries in the package. Check configuration entries and previous warnings to see description of which configuration failed.
- The configuration file: "C:\DeploymentTutorial\LoadXMLDataConfig.dtsConfig" is not valid. Check the configuration file name.
- Failed to load at least one of the configuration entries in the package. Check configuration entries and previous warnings to see description of which configuration failed.

These warnings do not affect package installation.

If you did not select the **Validate packages after installation** option on the Deploy SSIS Packages page, the Package Validation pages does not open and the wizard does not display post-installation information



about validation.

19. On the Finish the Package Installation Wizard page, read the installation summary and then click **Finish**.

**NOTE**

A temporary log file is created to use in the package validation. This file is not used when the package runs.

## Next Task in Lesson

[Step 3: Testing the Deployed Packages](#)

## See Also

[Integration Services Service \(SSIS Service\)](#)

# Lesson 3-3 - Testing the Deployed Packages

4/25/2017 • 5 min to read • [Edit Online](#)

In this task, you will test the packages that you deployed to an instance of SQL Server.

In other Integration Services tutorials, you ran packages in SQL Server Data Tools (SSDT), the development environment for Integration Services, using the **Start Debugging** option on the **Debug** menu. This time you will run the packages differently.

Integration Services provides several tools that you can use to run packages in the test and production environment: the command prompt utility **dtexec** and the Execute Package Utility. The Execute Package Utility is a graphical tool that is built on **dtexec**. Both of these tools execute the package immediately. In addition, SQL Server provides a subsystem of SQL Server Agent that is especially designed for scheduling package execution as a step in a SQL Server Agent job.

You will use the Execute Package Utility to run the deployed packages. The packages will be used as is; therefore, you do not have to update information on any pages in the dialog box. You will run the packages from the General page, which is the first page in the Execute Package Utility. If you like, you can click the other pages too see the information that they contain for each package.

## NOTE

To ensure that the packages run successfully in the context of this tutorial, you should not modify any options.

Before you run packages in SQL Server Management Studio by using the Execute Package Utility, ensure that the Integration Services service is running. The Integration Services service provides support for package storage and execution. If the service is stopped, you cannot connect to Integration Services and SQL Server Management Studio does not list the packages to run. You also must have permissions to run the package on the instance where the package has been deployed. For more information, see [Integration Services Roles \(SSIS Service\)](#).

The top-level folders within the Stored Packages folder are the user-defined folders that Integration Services service monitors. You can specify as many or few folders in the MsDtsSrvr.ini.xml file as you want. This tutorial assumes that you are using the default MsDtsSrvr.ini.xml file, and that the names of the top-level folders within Stored Packages are File System and MSDB.

## To connect to Integration Services in SQL Server Management Studio

1. Click **Start**, point to **All Programs**, point to **Microsoft SQL Server**, and then click **SQL Server Management Studio**.
2. In the **Connect to Server** dialog box, select **Integration Services** in the **Server type** list, provide a server name in the **Server name** box, and click **Connect**.

## IMPORTANT

If you cannot connect to Integration Services, the Integration Services service is likely not running. To learn the status of the service, click **Start**, point to **All Programs**, point to **Microsoft SQL Server**, point to **Configuration Tools**, and then click **SQL Server Configuration Manager**. In the left pane, click **SQL Server Services**. In the right pane, find the Integration Services service. Start the service if it is not already running.

SQL Server Management Studio opens. By default the Object Explorer window is open and placed in the upper right corner of the studio. If Object Explorer is not open, click **Object Explorer** on the **View** menu.

### To run the packages using the Execute Package Utility

1. In Object Explorer, expand the Stored Packages folder.
2. Expand the MSDB folder. Because you deployed the packages to SQL Server, all the deployed packages are stored in the msdb SQL Server database, and all deployed packages appear in the MSDB folder. The File System folder is empty, unless you have deployed packages to the file system outside of the Deployment Tutorial.
3. Starting at the top of the package list, right-click DataTransfer, and click **Run Package**.
4. In the **Execute Package Utility** dialog box, click **Execute**.
5. In the **Execute Package Utility** dialog box, view the progress and execution results of the package. When the **Stop** button becomes unavailable, which indicates that the package has completed, click **Close**.

#### IMPORTANT

If you click **Stop** while the package is running, the package will not finish.

6. In the **Execute Package Utility** dialog box, click **Close**.
7. Repeat steps 3 - 6 for the LoadXML package.
8. On the **File** menu, click **Exit**.

### To verify the results of the DataTransfer package

1. On the toolbar in SQL Server Management Studio, click **New Query**.
2. In the **Connect to Server** dialog box, select **Database Engine** in the **Server type** list, provide the name of the server name on which you installed the tutorial packages or type (local) in the **Server name** box, and select an authentication mode. If using SQL Server Authentication, provide a user name and password.
3. Click **Connect**.
4. In the query window, type or paste the following SQL statement:

```
USE AdventureWorks
```

```
SELECT * FROM HighIncomeCustomers
```

5. Press **F5** or click the Execute icon on the toolbar.

The query returns 31 rows of data. The return result contains any rows from the text file, Customers.txt, that have values larger than 100000 in the YearlyIncome column.

6. Locate the DeploymentTutorial folder, right-click the log XML file, Deployment Tutorial Log, and then click **Open**. You can open the file by using Notepad or the text/XML editor of choice.

### To verify the results of the LoadXMLData package

1. On the toolbar in SQL Server Management Studio, click **New Query**.
2. If prompted to connect again, in the **Connect to Server** dialog box, select **Database Engine** in the **Server type** list, provide the name of the server on which you installed the tutorial packages or enter (local) in the **Server name** box, and select an authentication mode. If using SQL Server Authentication, provide a user name and password.
3. Click **Connect**.
4. In the query window, type or paste the following SQL statement:

```
USE AdventureWorks
```

```
SELECT * FROM OrderDatesByCountryRegion
```

5. Press **F5** or click the Execute icon on the toolbar.

The query returns 21 rows of data. The return result consists of the rows from the XML data file, orders.xml. Each row is a summary by country/region; the row lists the name of a country/region, the number of orders for each country/region and the dates of the newest and oldest orders.

## See Also

[dtexec Utility](#)