Tech Note 883

Using SQL Server Integration Services (SSIS) to Automate Work with Wonderware DBs

---

Topic#: 002687
Created: October 2012

Introduction

Microsoft defines SQL Server Integration Services (**SSIS**) as a platform for building high performance data integration solutions, including extraction, transformation, and load (ETL- **E**xtract, **T**ransform, **L**oad) packages for data warehousing.

A simpler way to think of SSIS is that it's the solution for automating SQL Server.SSIS provides a way to build packages made up of tasks that can move data around from place to place and alter it on the way. There are visual designers (hosted within Business Intelligence Development Studio) to help you build these packages as well as an API for programming SSIS objects from other applications.

This *Tech Note* uses the Alarm DB for an example.

Application Versions

- Microsoft SQL Server 2005 and later
- SQL Server Intelligence Development Studio

Assumptions

- This Tech Note assumes you are familiar with SQL Server Setup and SQL Server Intelligence Development Studio.

Task Overview

1. **Installing SQL Server Integration Servic**e
2. **Using SSIS**
3. **Manually Executing a SQL Server SSIS Package**
4. **Scheduling the SSIS Job**

Installing SQL Server Integration Service

Check integration services from the features

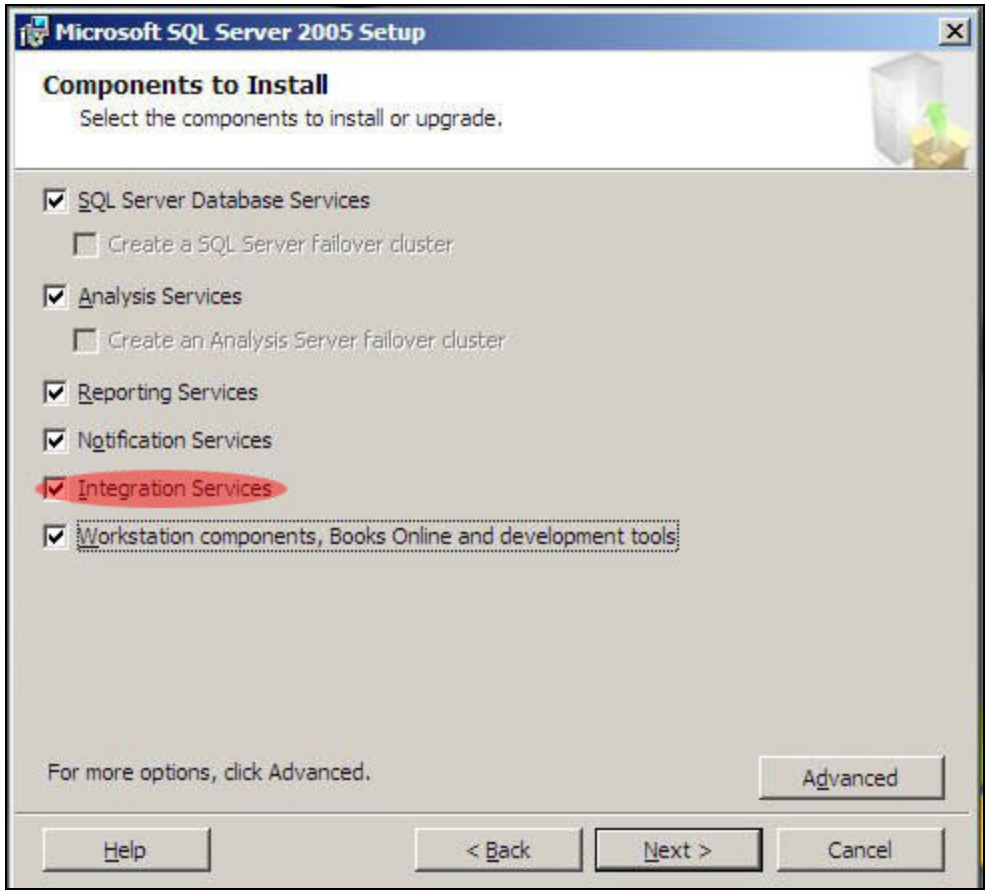

FIGURE 1: SQL SERVER 2005 SETUP

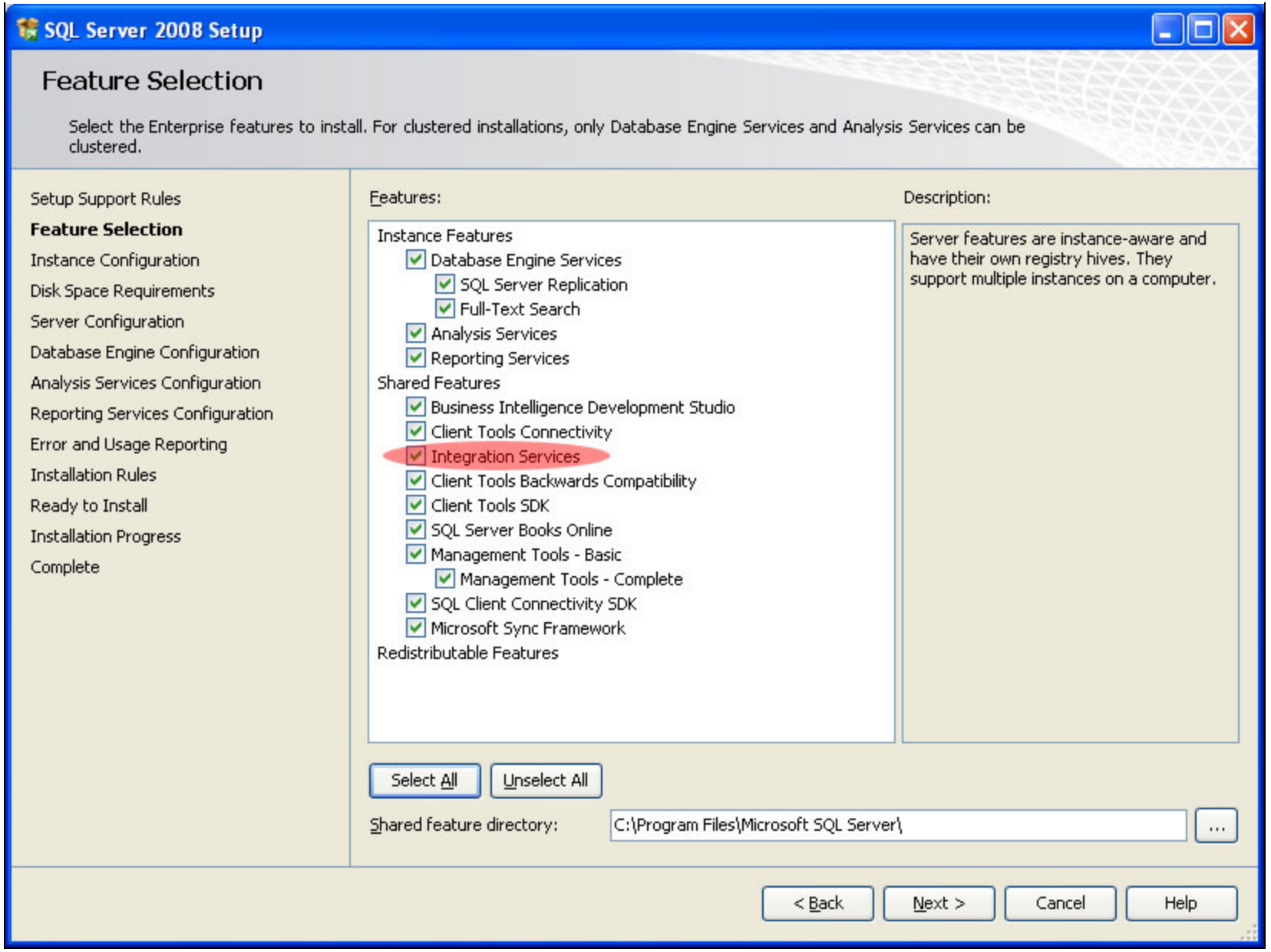

FIGURE 2: SQL SERVER 2008 SETUP

Using SSIS

In this section we will explain in details with example how to benefit SSIS to perform set of tasks on your DB and we will use WWAlarmDB as an example.

Example 1: Extract AlarmMaster Table to a File.

1. Open SQL Server Intelligence Development Studio and create a new **Integration Services Project**.

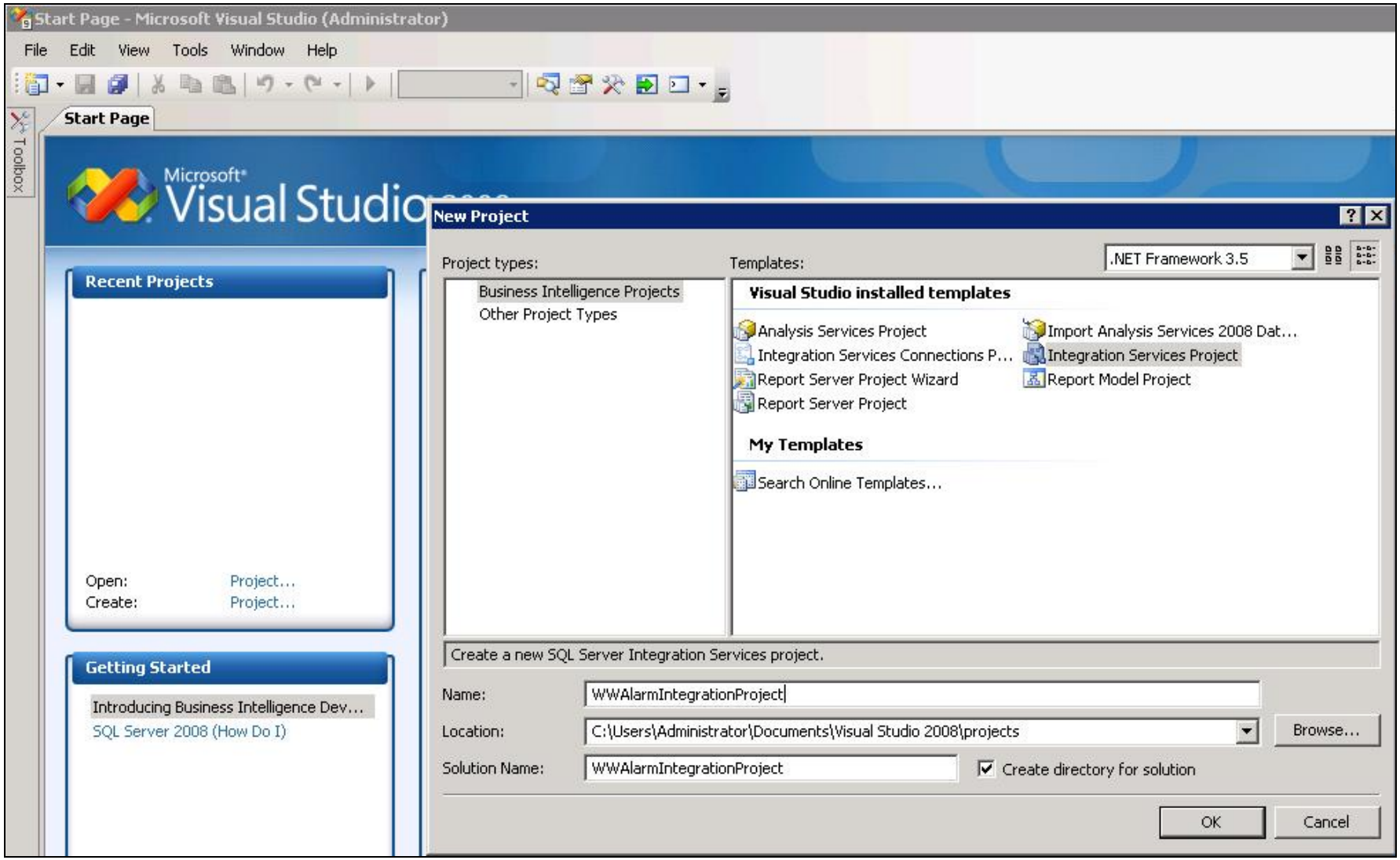

FIGURE 3: SQL SERVER INTELLIGENCE DEVELOPMENT STUDIO: NEW PROJECT

2. Right-click the **Connection Manager** pane to create a new OLE DB Connection (Figure 4 below).

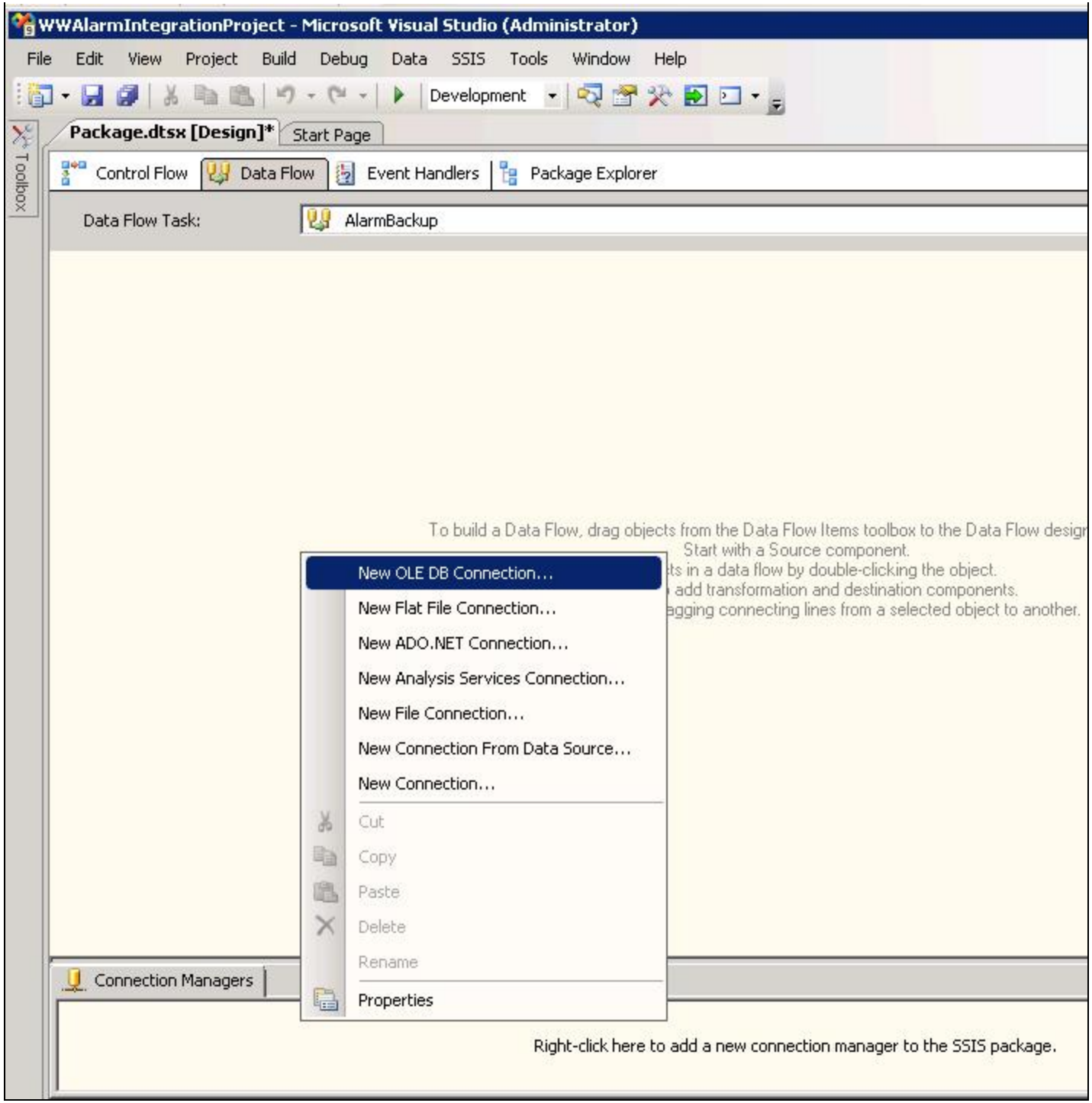

FIGURE 4: CONNECTION MANAGER - NEW OLE DB CONNECTION

3. The **Configure OLE DB Connection Manager** appears (Figure 5 below).

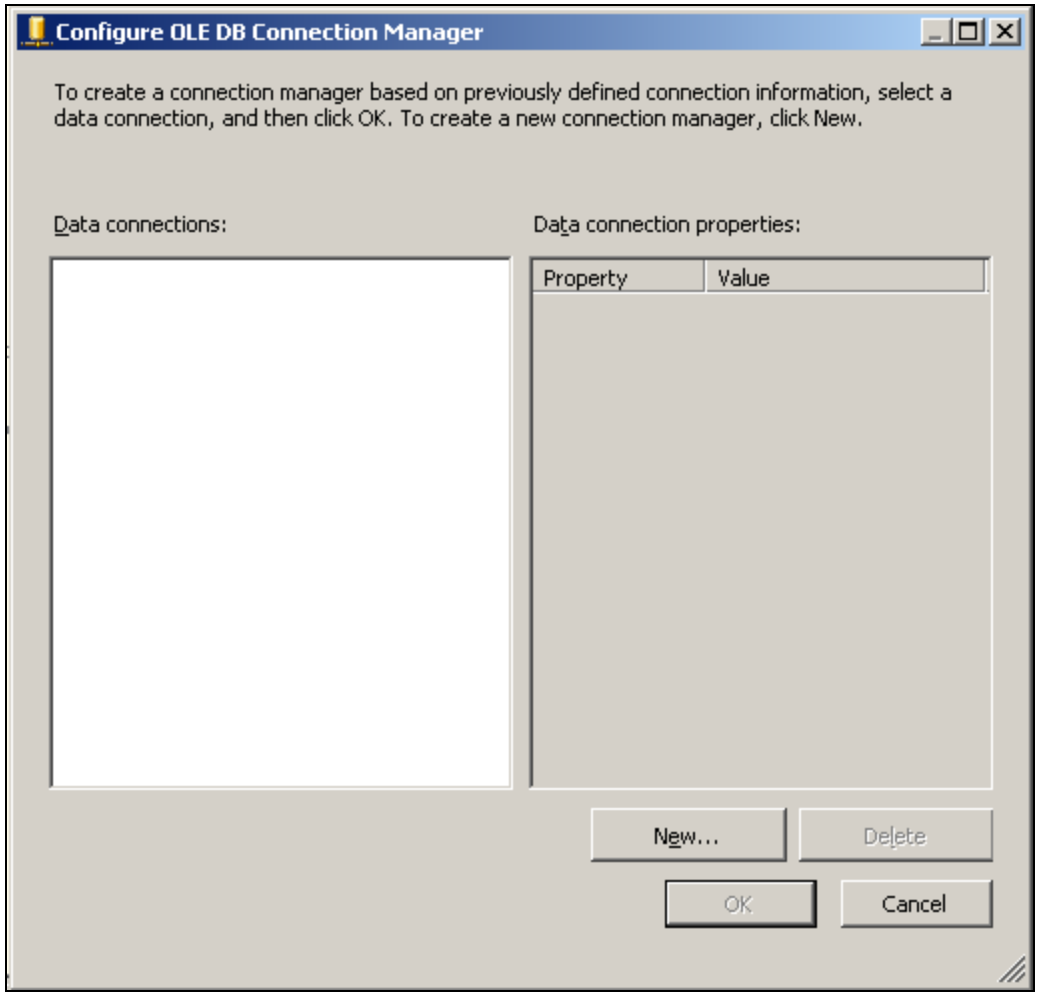

FIGURE 5: CONFIGURE OLE DB CONNECTION MANAGER WINDOW

4. Click the New button to add the connection and configure it (Figure 6 below).

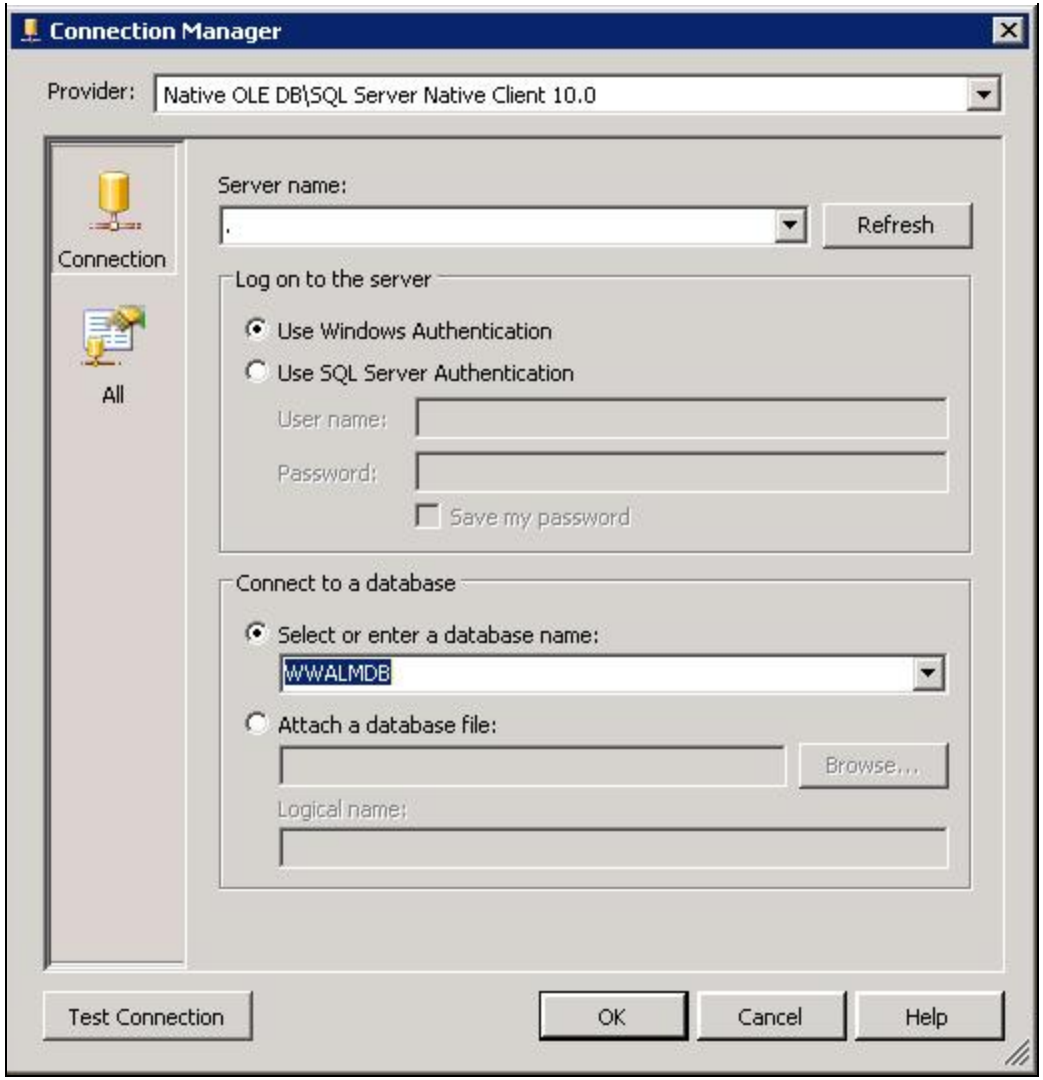

FIGURE 6: CONNECTION MANAGER WINDOW

For this example, the Dot in the Server name field refers to the local machine. Wonderware recommends using the actual network name of the machine for the local connection. If you need to connect with a remote DB, use the IP address for the machine where the DB exists.

5. Click the **Test Connection** button to test the connection (Figure 7 below).

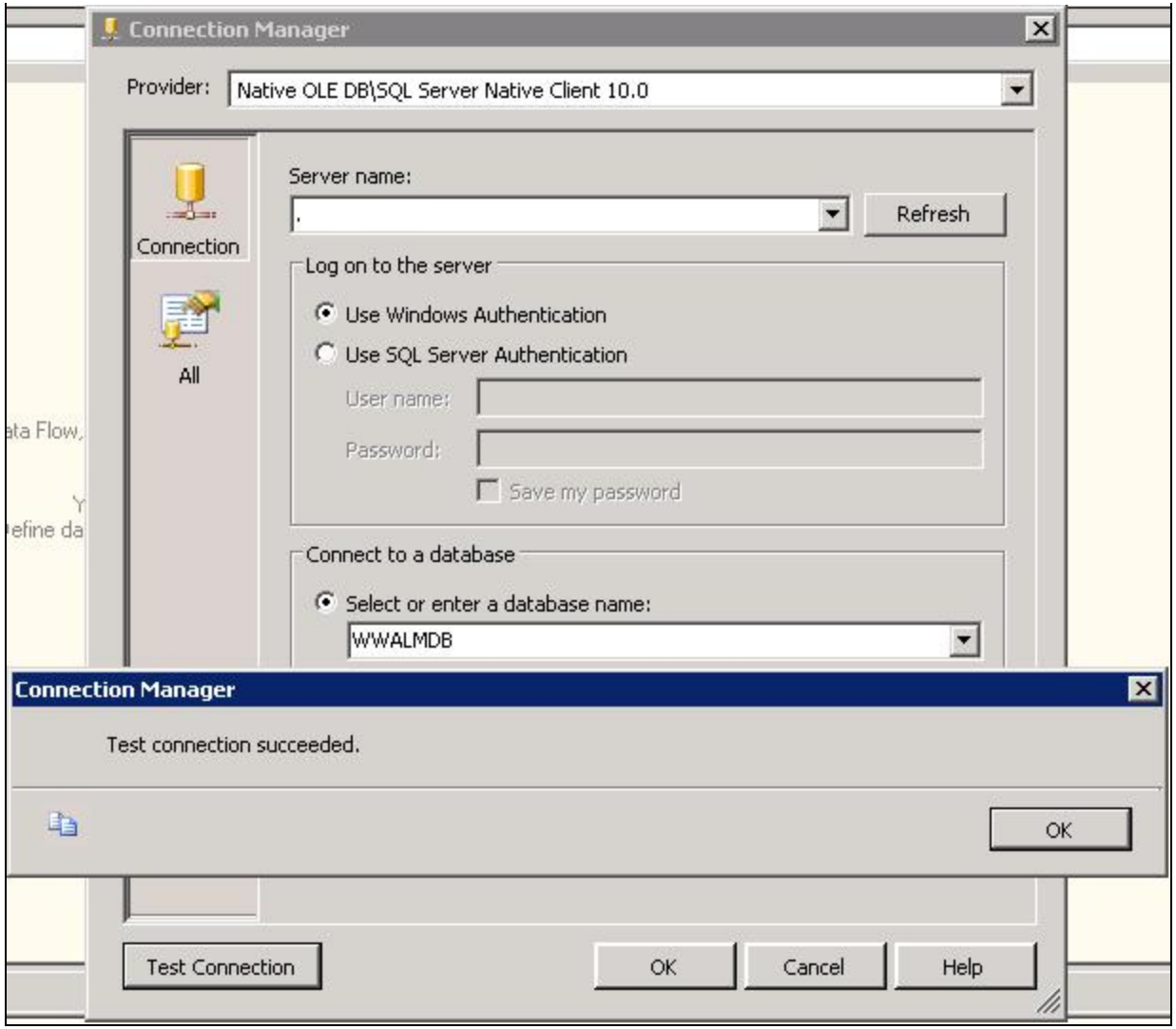

FIGURE 7: TEST CONNECTION SUCCEEDED MESSAGE

6. Click **OK**, then **OK** again to close the Connection Manager window.
7. From the connection managers tab, rename the connection (Figure 8 below). In this example, it is **AlarmDBConnection**.

> **Note**: The **Connection Managers** tab lists all configured connections in your project. To modify any connection, just locate it and double-click on it to re-configure it using the **Connection Manager** window.

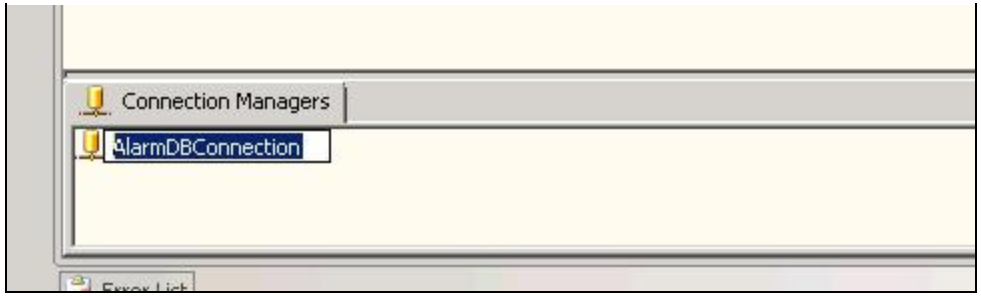

FIGURE 8: RENAME THE CONNECTION MANAGER

8. Click the **Control Flow** tab.
9. Click the **Toolbox** tab (at left) and under **Control Flow** items drag-and-drop **Data Flow Task** to the main pane (Figure 9 below).

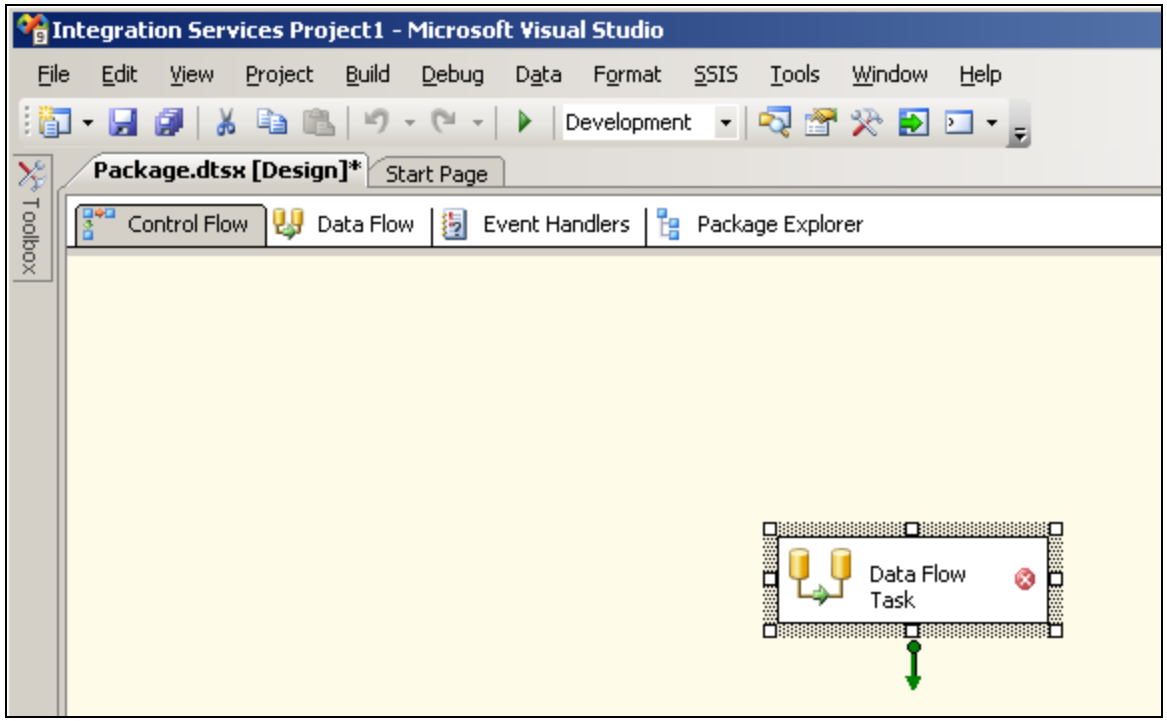

FIGURE 9: NEW DATA FLOW TASK

10. Rename the data flow task **AlarmDBToFileTask** (Figure 10 below).

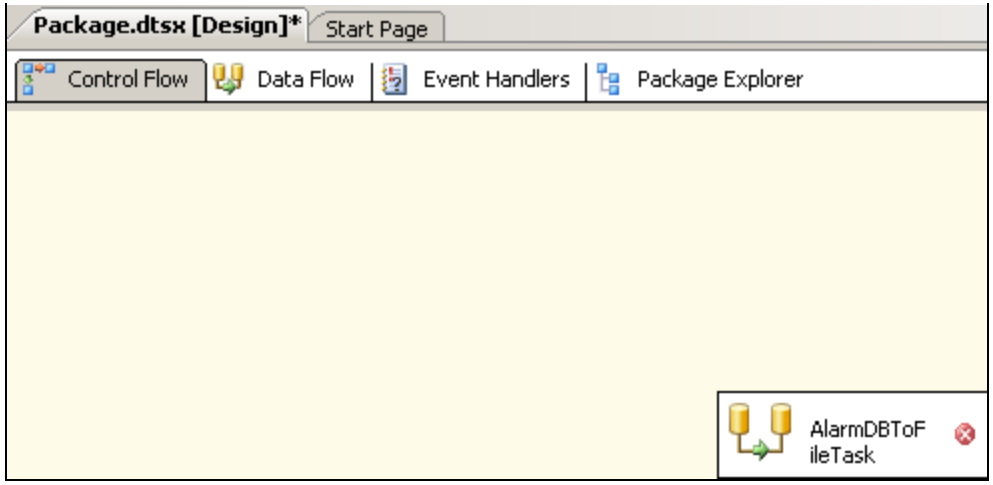

FIGURE 10: RENAME THE DATA FLOW TASK

11. Click the **Data Flow Sources** tab, then drag-and-drop **OLE DB Source** (Figure 11 below) to the main panel. The red x indicates that the source is not configured yet.

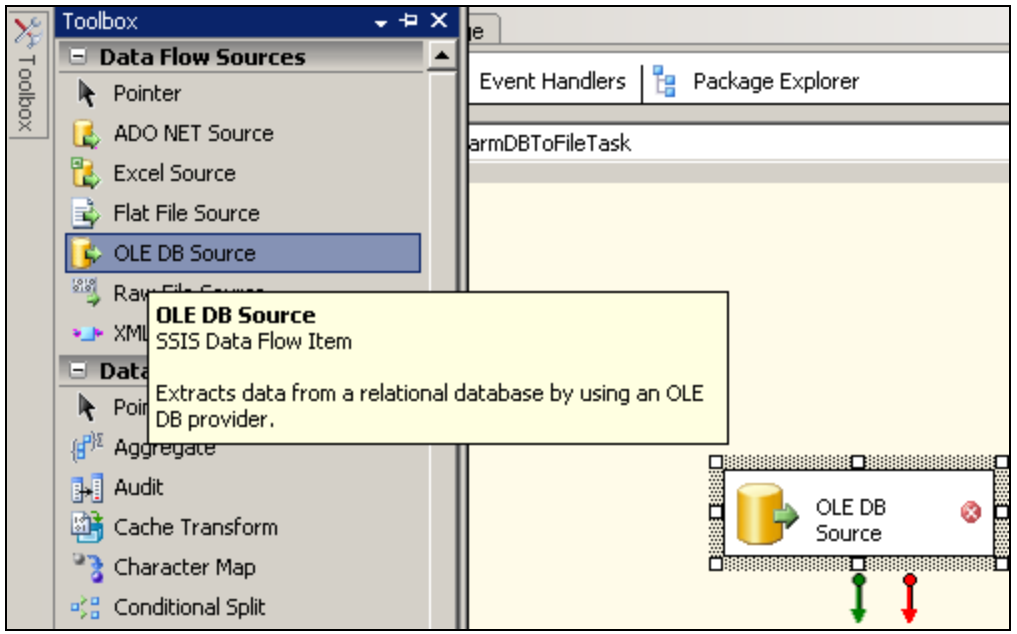

FIGURE 11: OLE DB SOURCE

12. Rename the OLE DB Source. For this example, it is **AlarmDBAlarmMasterTable** (Figure 12 below).

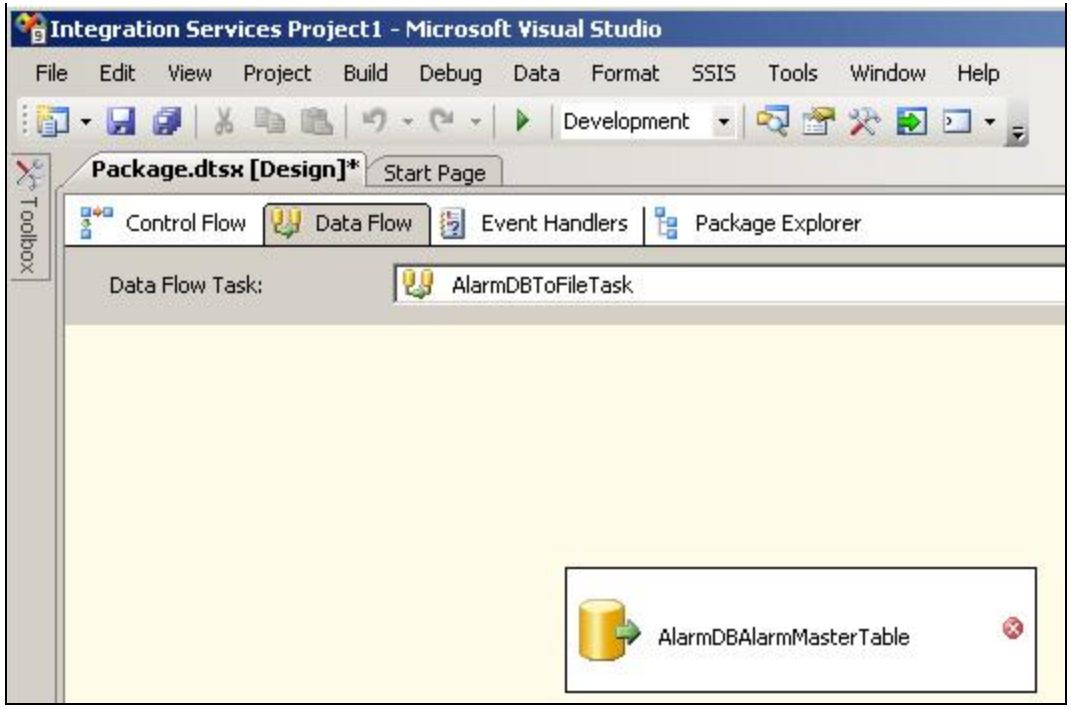

FIGURE 12: RENAMING THE OLE DB SOURCE

13. Double-click the OLE DB Source. All connections are listed in the **OLE DB Source Editor** (Figure 13 below).

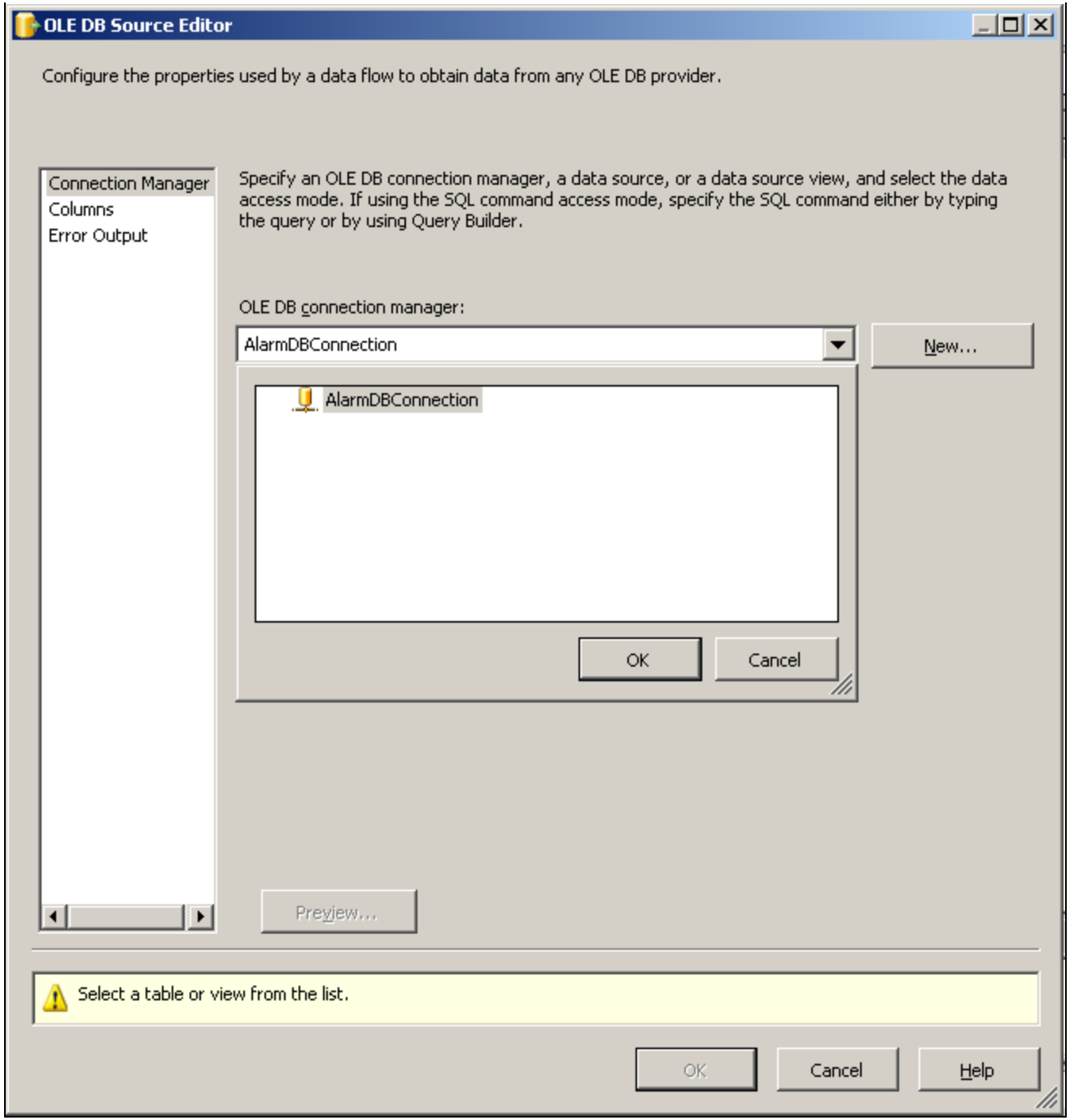

FIGURE 13: OLE DB SOURCE EDITOR

14. Select **AlarmDBConnection**, then select the **dbo.AlarmMaster** table from the **Name of the table or the view list** (Figure 14 below).

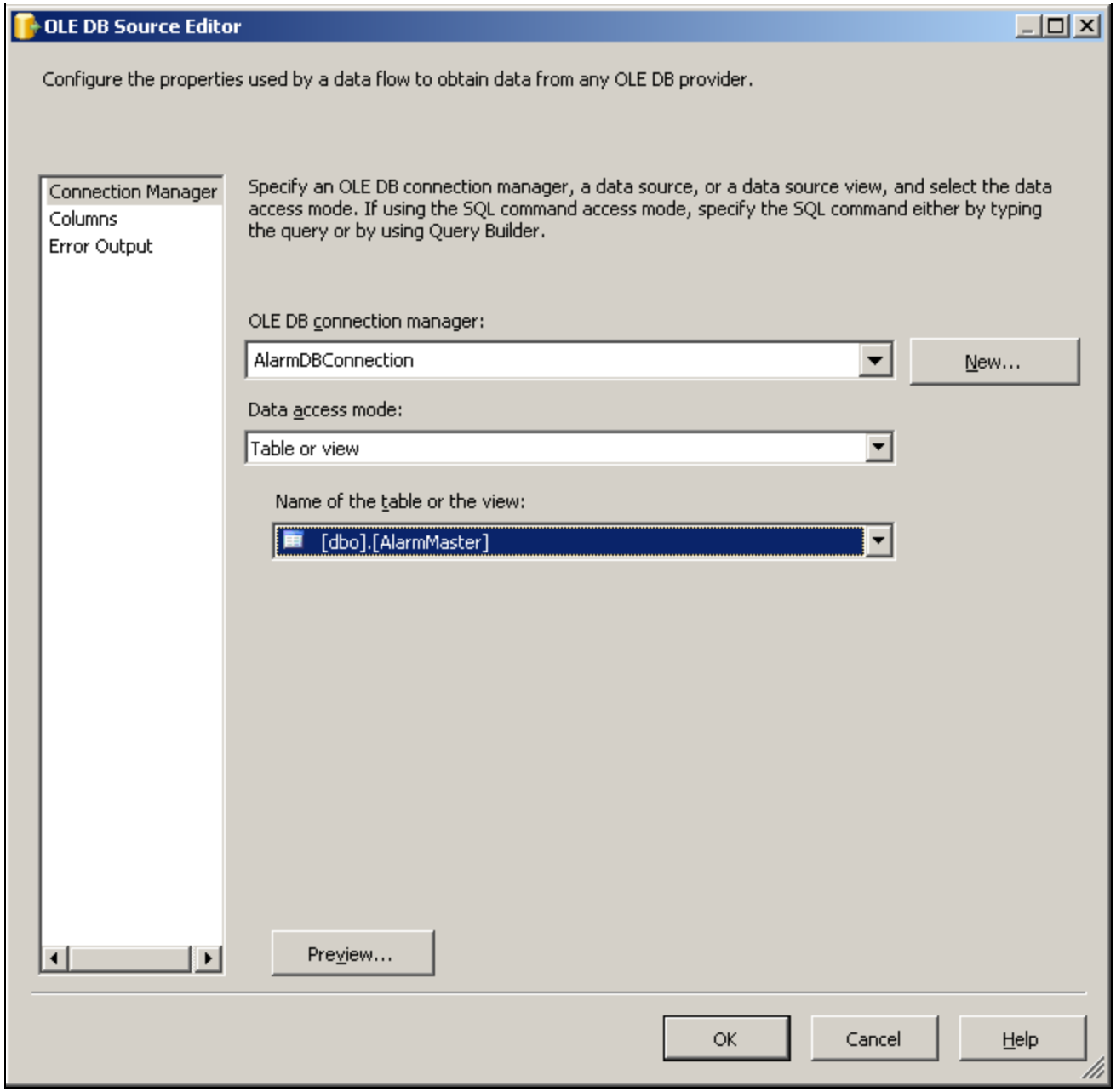

FIGURE 14: SELECT DBO.ALARMMASTER TABLE

15. Click **Preview** to display the table content (Figure 15 below)

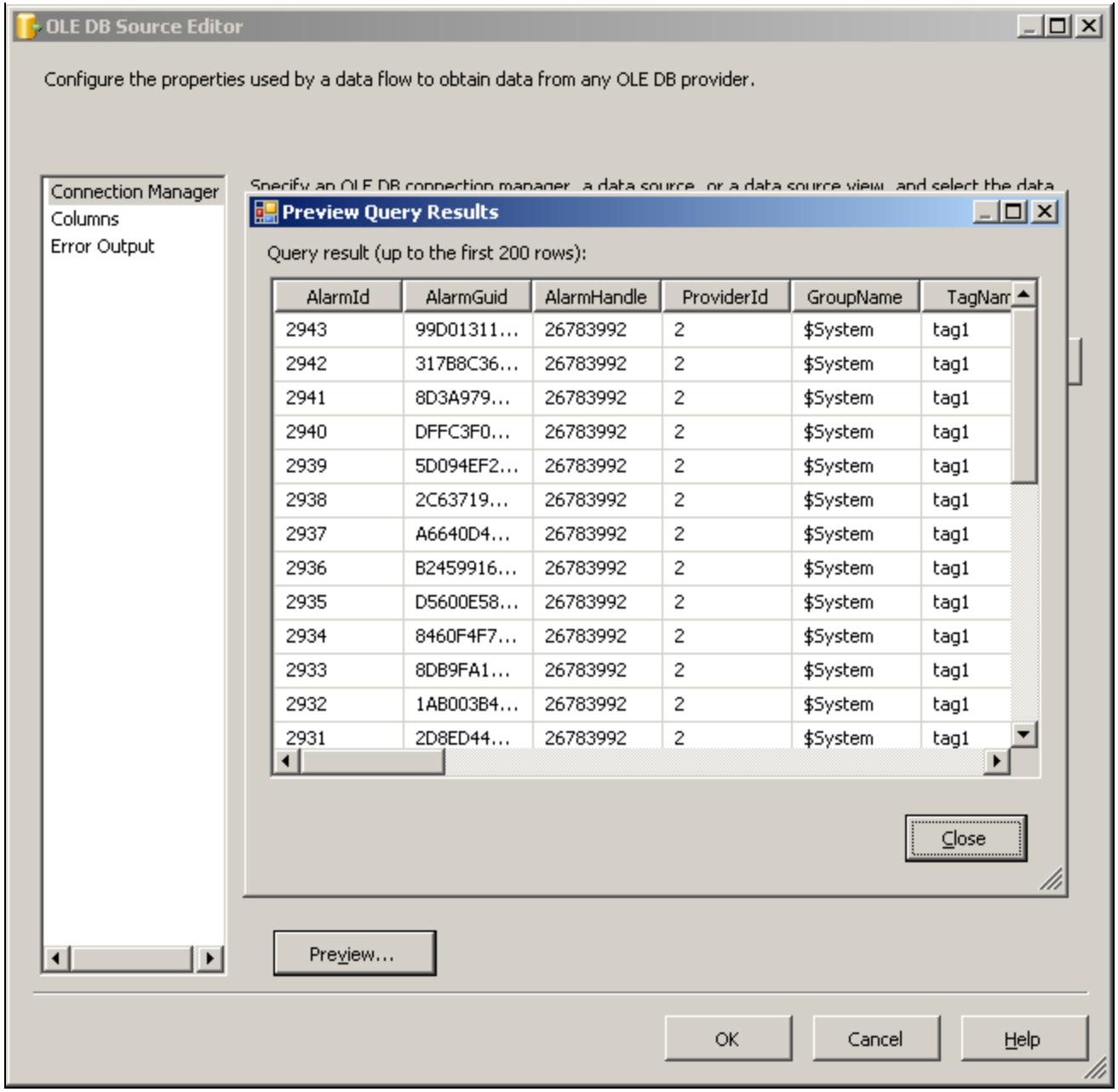

FIGURE 15: PREVIEW QUERY RESULTS

16. Click **Close**.

17. In OLE DB Source Editor, click **Columns**. The table's columns appear. You can uncheck specific columns to be discarded from the results (Figure 16 below).

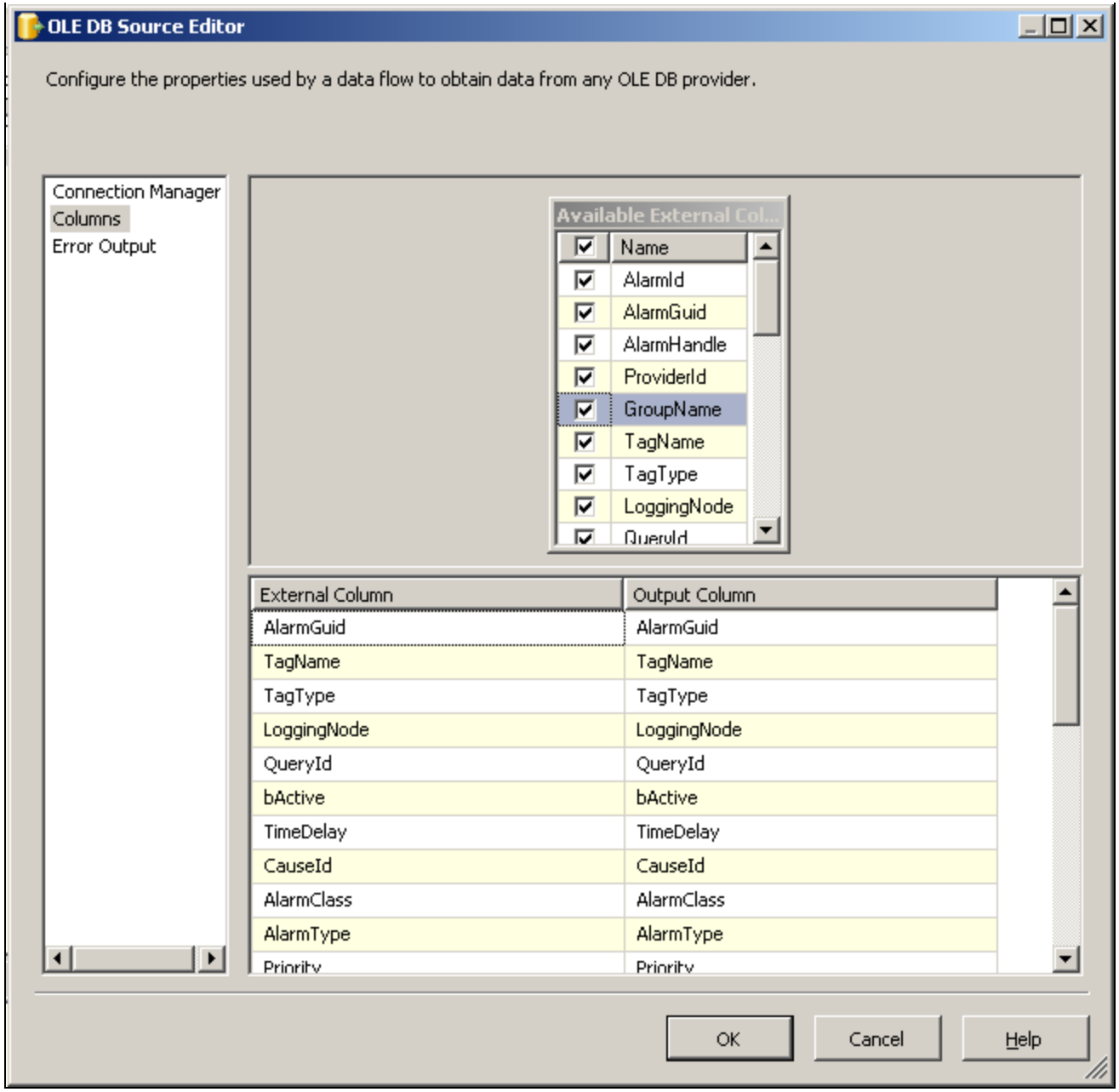

FIGURE 16: SELECT COLUMNS

18. Click **OK**. The Red x disappears, which means that the source is configured properly without errors (Figure 17 below).

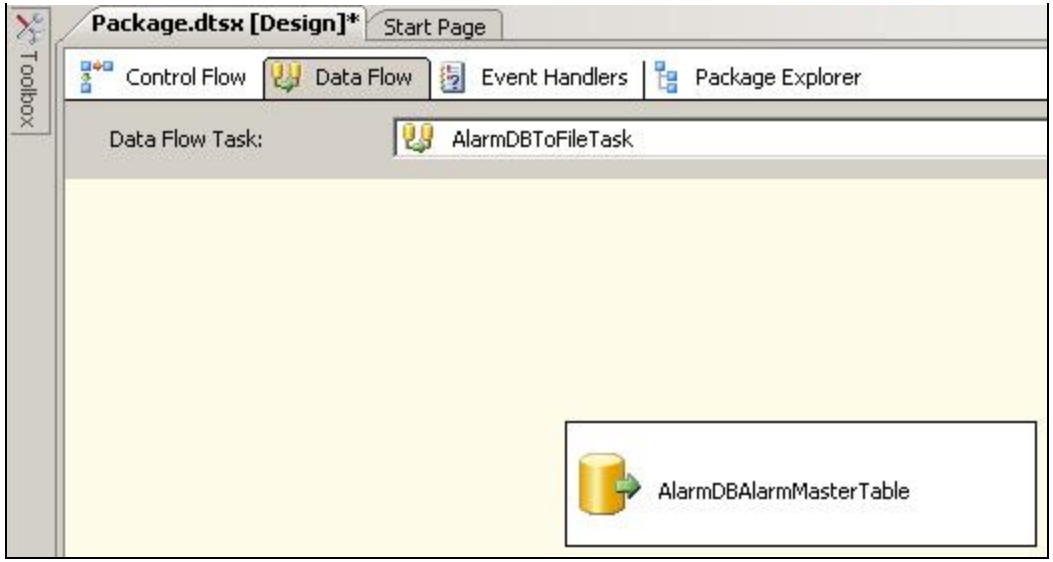

FIGURE 17: ALARMDBALARMMASTERTABLE SOURCE CONFIGURED CORRECTLY

19. Click the Toolbox, and under **Data Flow Destinations** drag-and-drop **Flat File Destination** (Figure 18 below). The red x indicates that the destination is not configured yet.

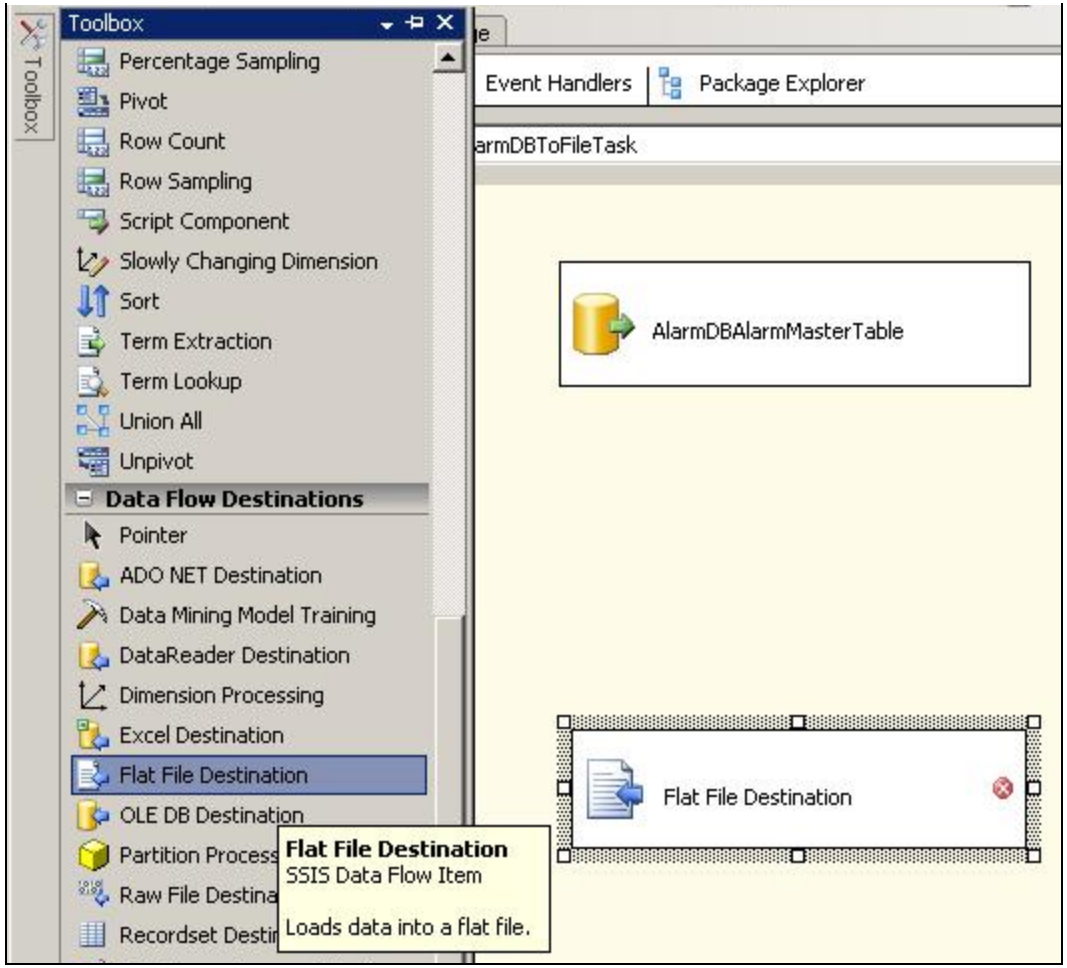

FIGURE 18: FLAT FILE DESTINATION

20. Rename the Flat File Destination. In this example, it is **AlarmMasterFile** (Figure 19 below).

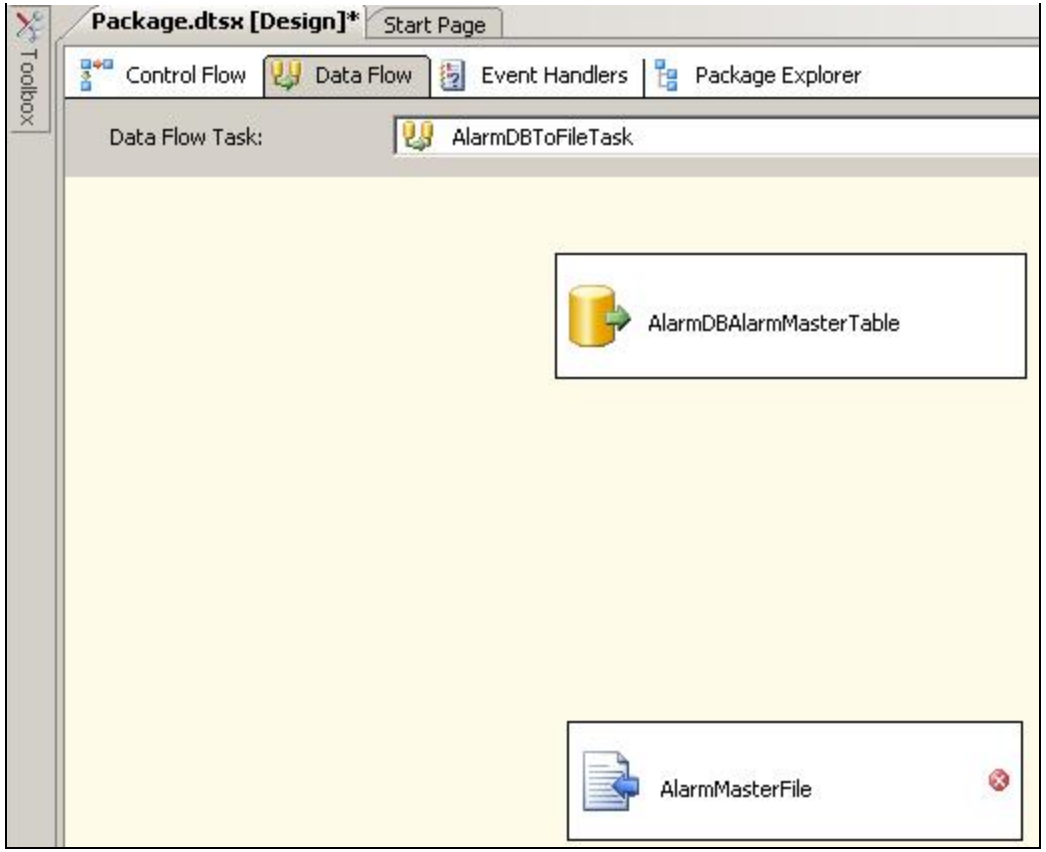

FIGURE 19: RENAME THE FLAT FILE DESTINATION

21. Drag the green arrow from **AlarmDBAlarmMasterTable** source to **AlarmMasterFile** destination (Figure 20 below)

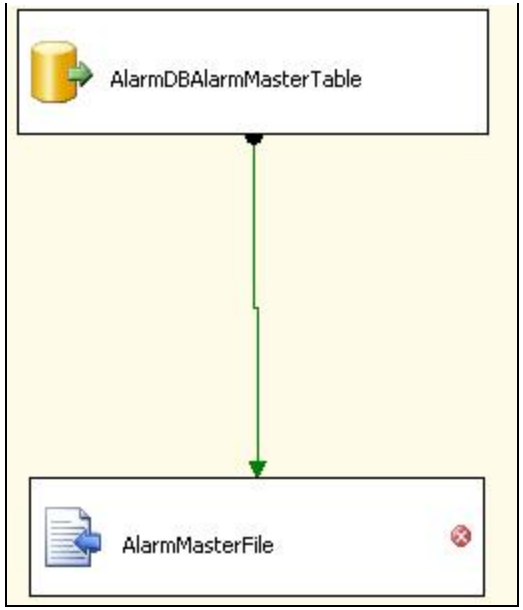

FIGURE 20: CONNECT SOURCE AND DESTINATION

22. Double-click the **AlarmMasterFile** destination to open the **Flat File Destination Editor** (Figure 21 below).

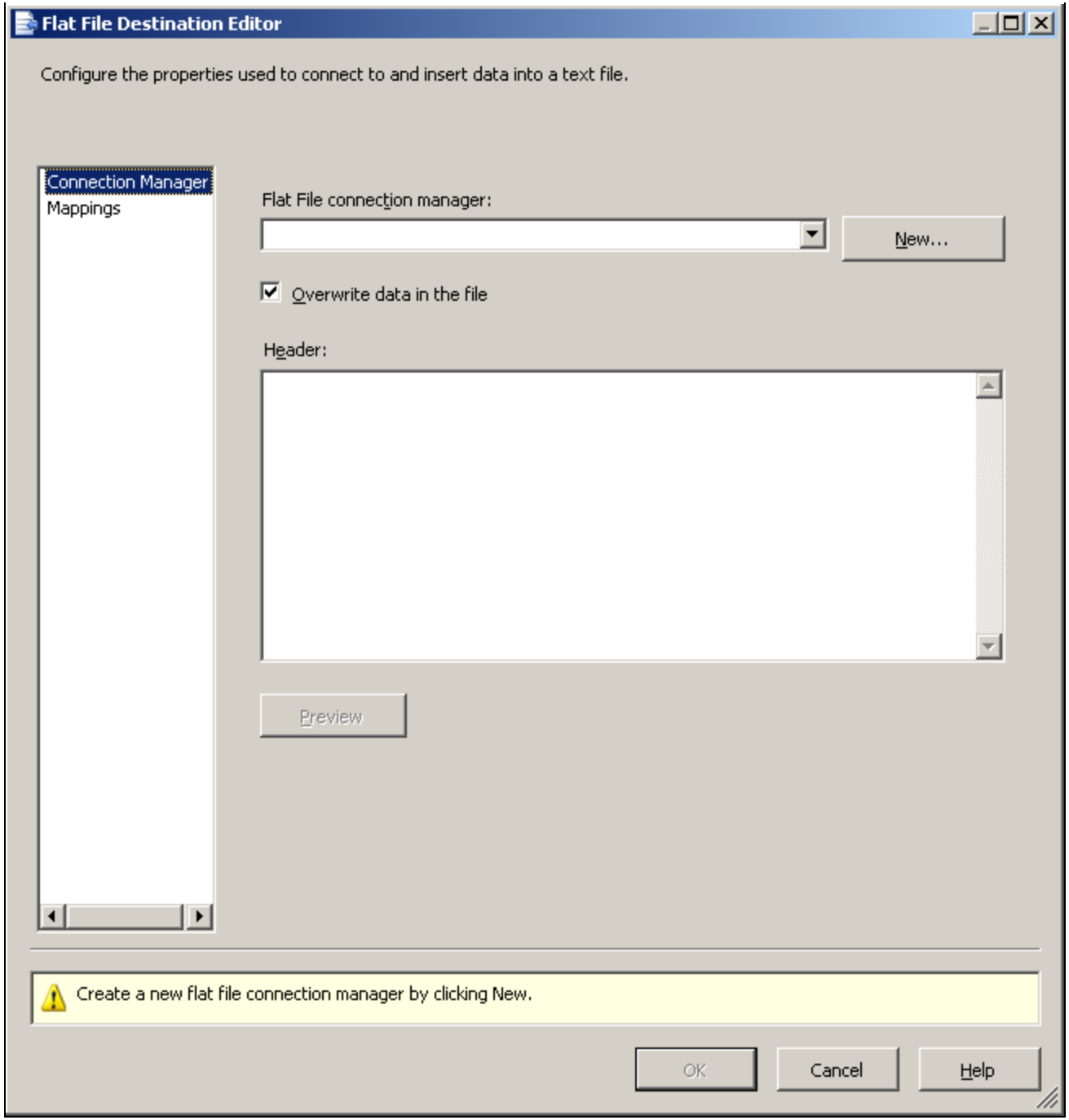

FIGURE 21: FLAT FILE DESTINATION EDITOR WINDOW

23. Click **New** to configure the **Flat File connection manager**.
24. Choose the format of the destination flat file from the **Flat File Format** window. In this example, the format is **Delimited**.

25. Click **OK** (Figure 22 below).

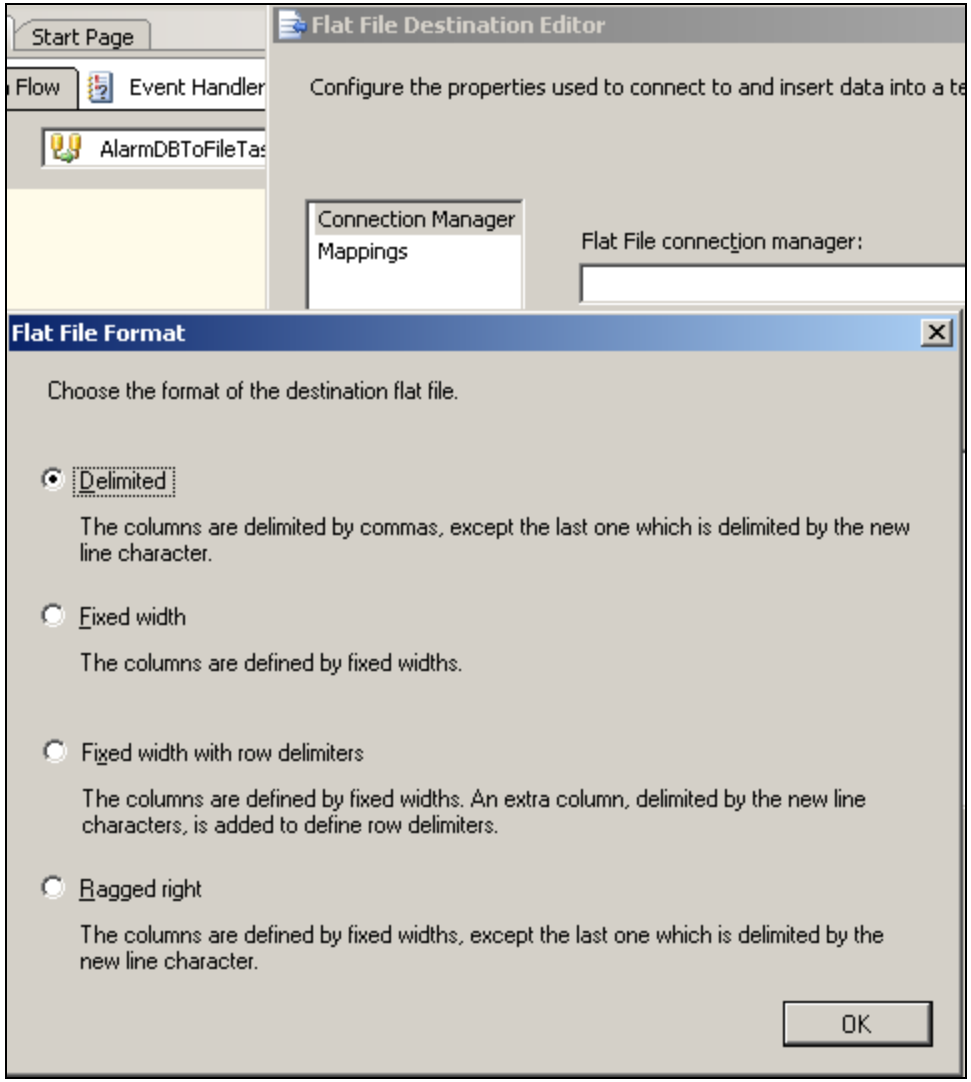

Figure 22: Flat File Format Window

26. The **Flat File Connection Manager Editor** opens. Name the connection and click the **Browse** button to select the file (Figure 23 below).

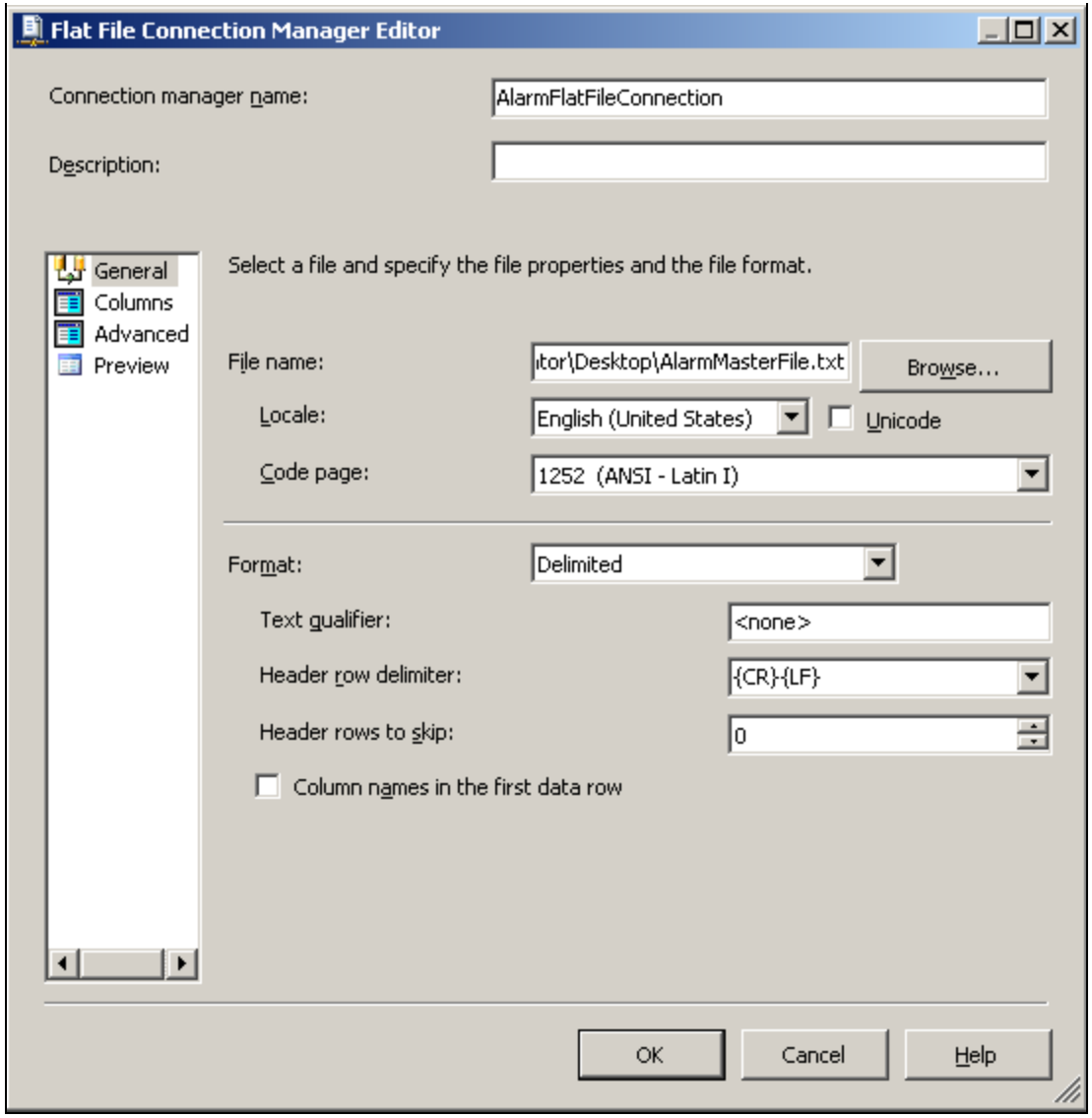

FIGURE 23: FLAT FILE CONNECTION MANAGER EDITOR

27. Click **OK**.

28. Click **Mappings** in the **Flat File Destination Editor**. All columns from the **Available Inputs** box connect to their corresponding columns in the **Available Destination** box (Figure 24 below).

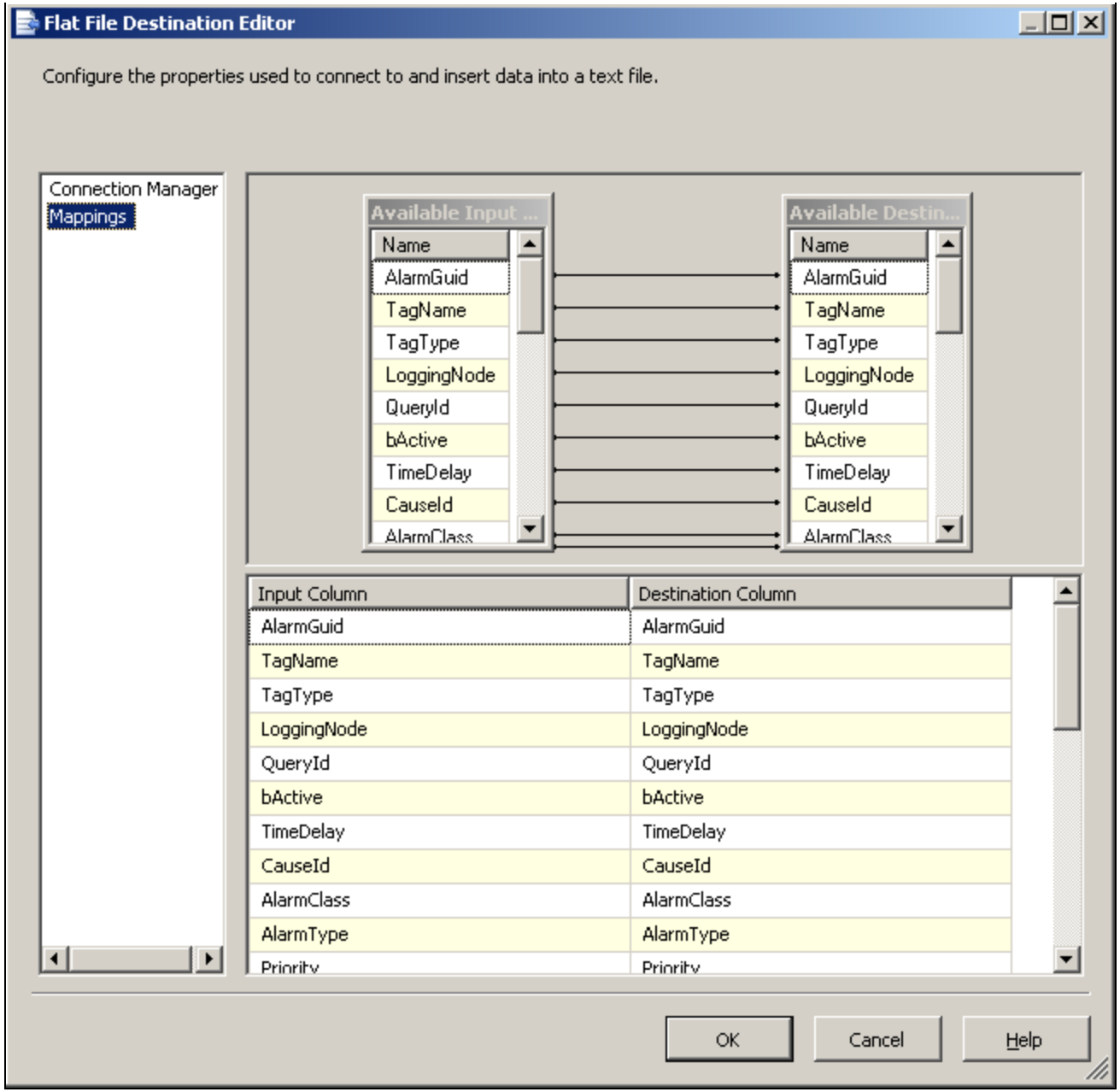

FIGURE 24: MAPPED INPUT COLUMNS TO DESTINATION COLUMNS

29. Click **OK**. The mapping is complete (Figure 25 below).

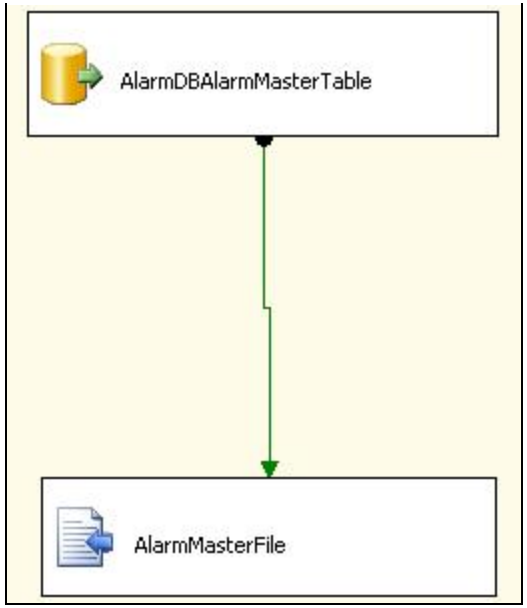

FIGURE 25: FINAL DATAFLOW

30. Run the project by clicking on the green **Start Debugging** button in the main toolbar. The Green color indicates that everything is mapped correctly (Figure 26 below).

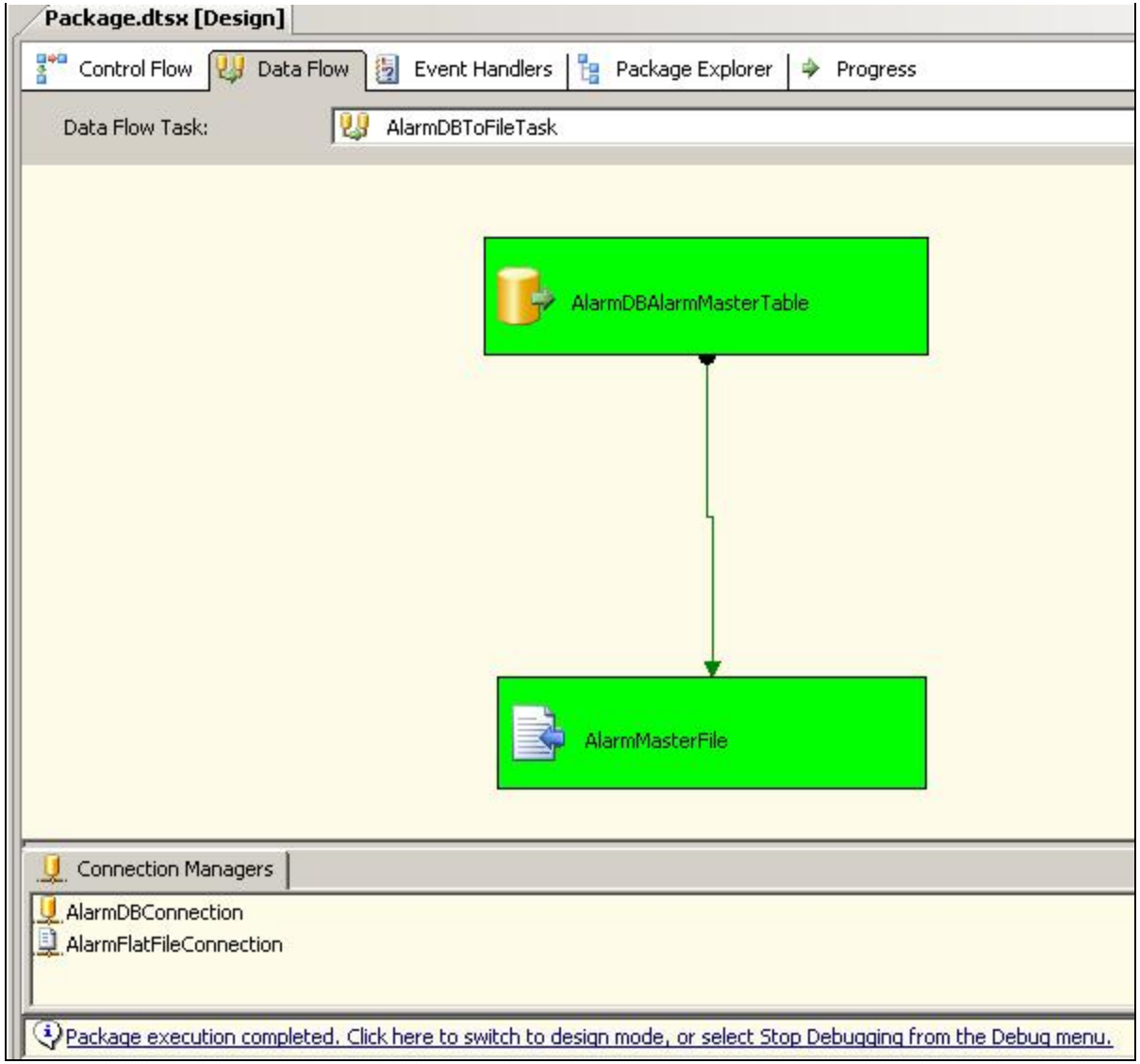

FIGURE 26: START DEBUGGING

31. Verify the file is created successfully by opening it (Figure 27 below). Use the path you configured in **Figure 23 (above)**.

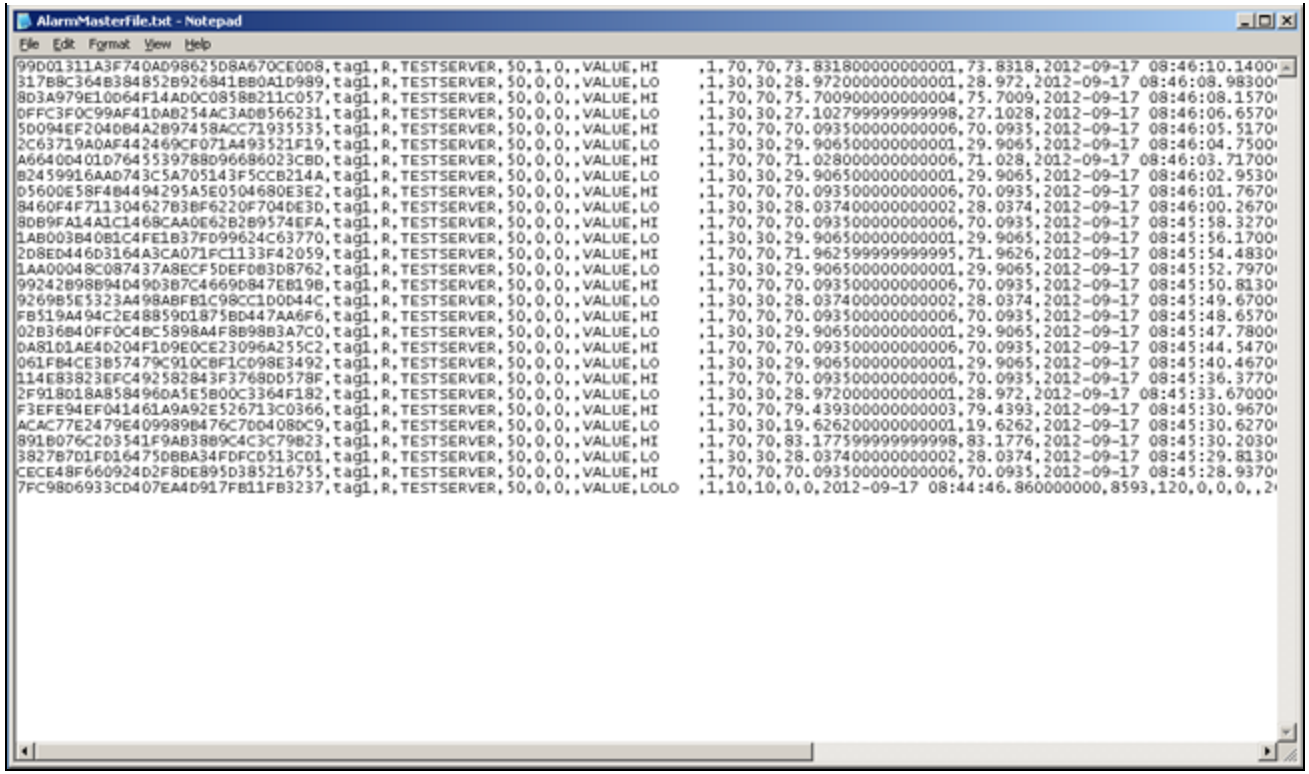

FIGURE 27: FILE CREATED SUCCESSFULLY

Example 2: Create WWAlarmDB Backup

1. Create a new Integration Service project and add **Back Up Database task** from the Maintenance Tasks Pane (Figure 28 below).

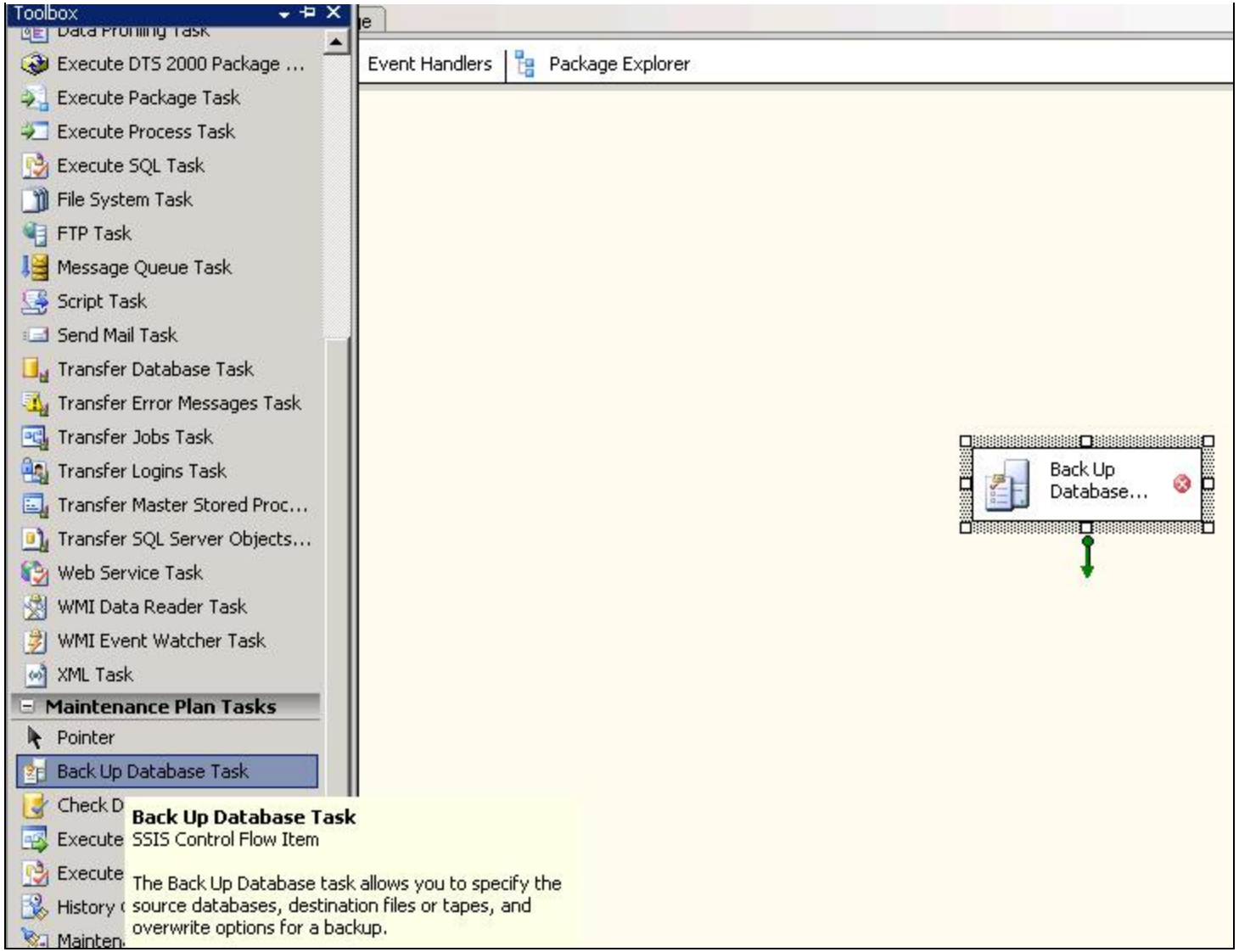

FIGURE 28: BACK UP DATABASE TASK

2. Double-click the **Back Up Database** task. The backup database task configuration window appears.
3. Click **New** to create a new connection (Figure 29 below)

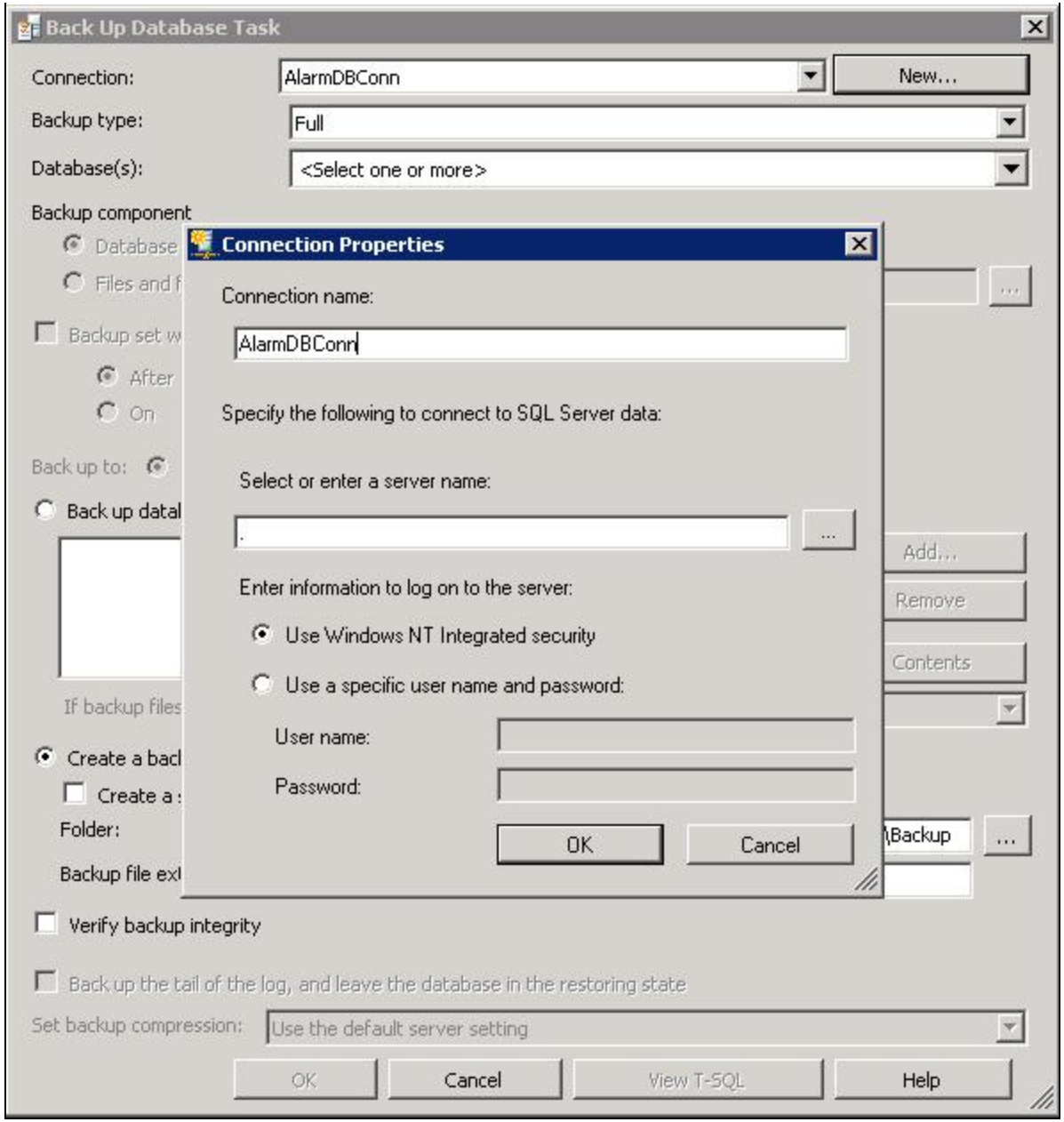

FIGURE 29: CONNECTION PROPERTIES

4. Select the database to backup. In this example, it is **WWALMDB** (Figure 30 below).

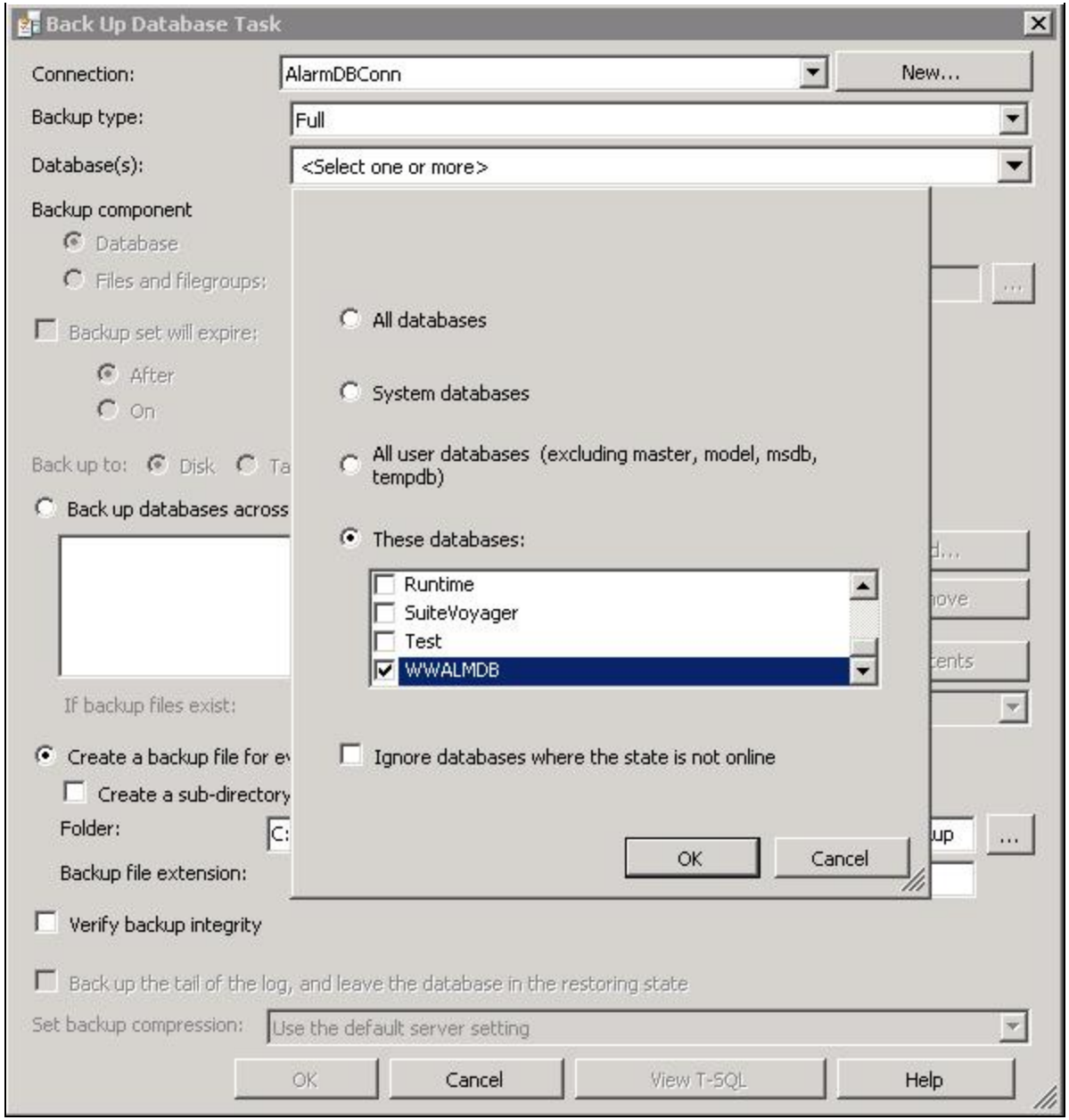

FIGURE 30: SELECT DATABASE TO BACKUP

5. Click **OK** and select the path for the backup.

> **Note:** Pay attention to a **C:** folder that some OSs don't allow for use. It is recommended to use/create something like **C:\bak**.

6. Run the project by pressing **F5** (Figure 31 below).

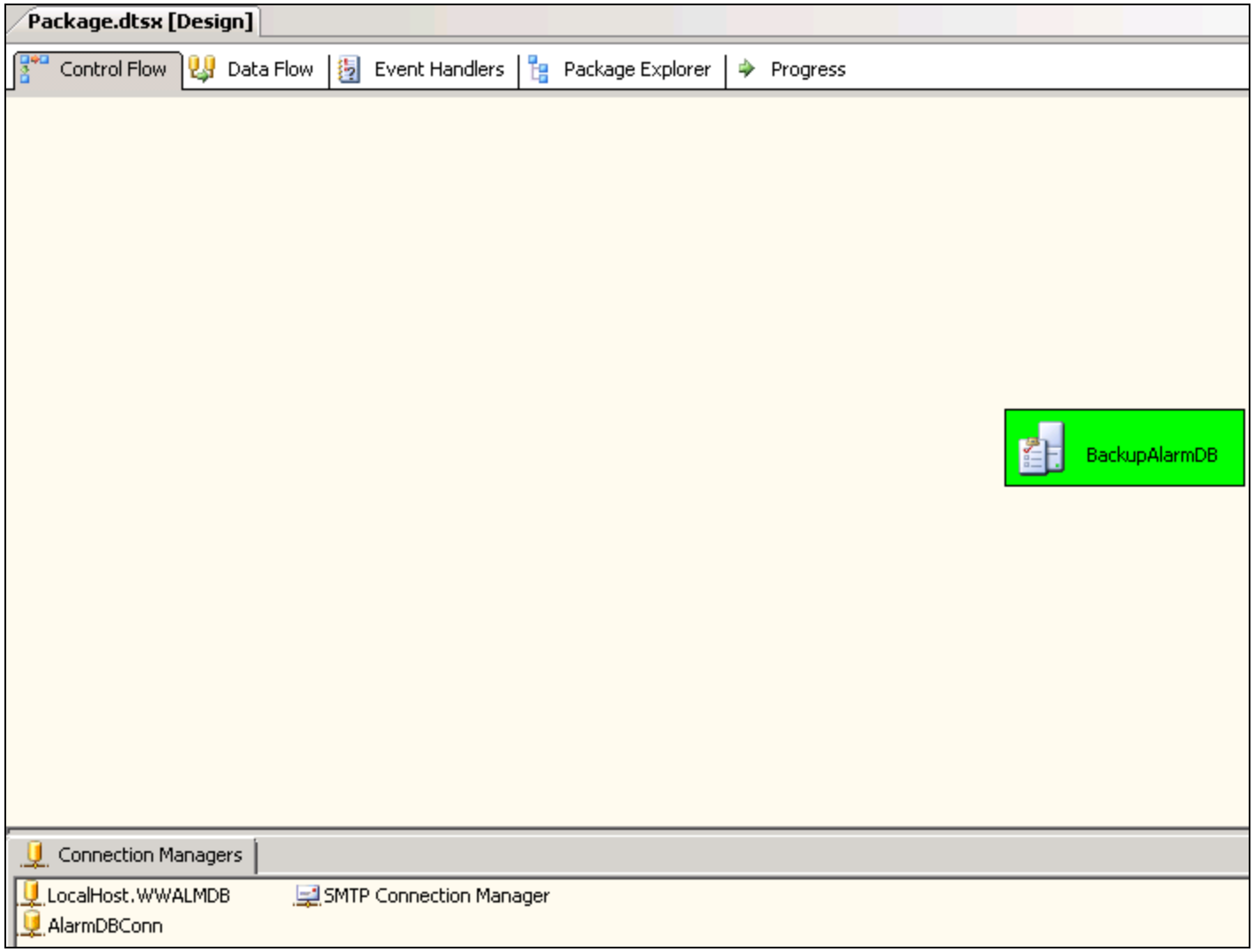

Figure 31: Block Converted to green as a result of debugging WWAlmDb backup

7. Open C drive, you will find that the backup was created (Figure 32 below)

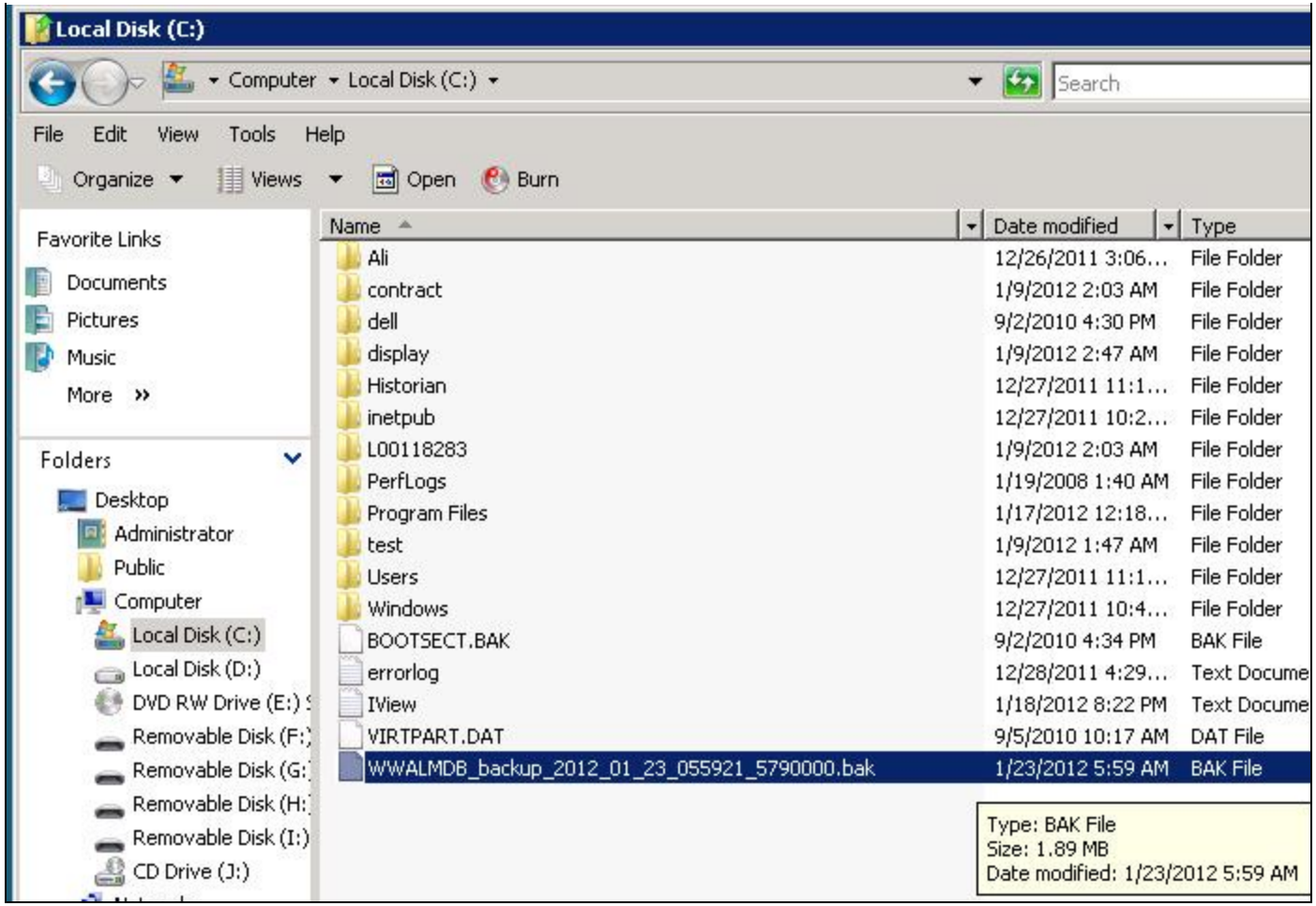

FIGURE 32: CONFIRM BACK UP FILE

Manually Executing a SQL Server SSIS Package

Using the Execute Package Utility (DTEXECUI.EXE) graphical interface, you can execute an SSIS package that is stored in a File System, SQL Server or an SSIS Package Store.

1. Run the command DTEXECUI.EXE to open up the Execute Package Utility (Figure 33 below).

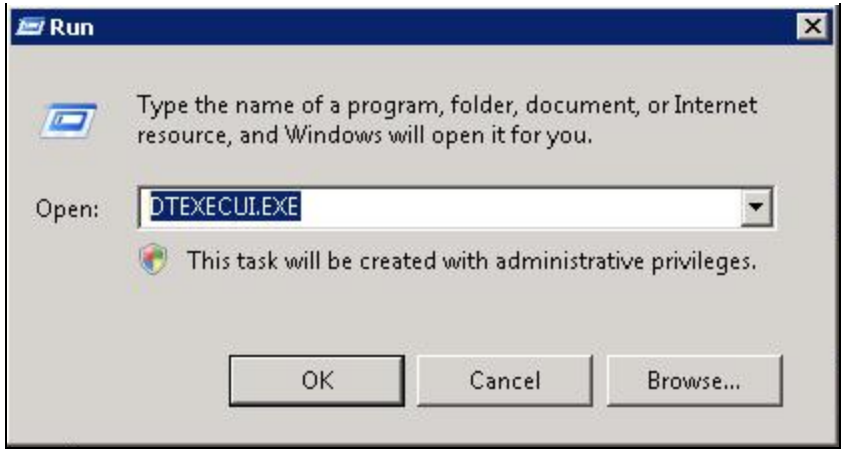

FIGURE 33: DTEXECUIEXE

The Execute Package Utility window appears.

2. From the **General** tab, choose the Package source as **File System** (Figure 34 below).

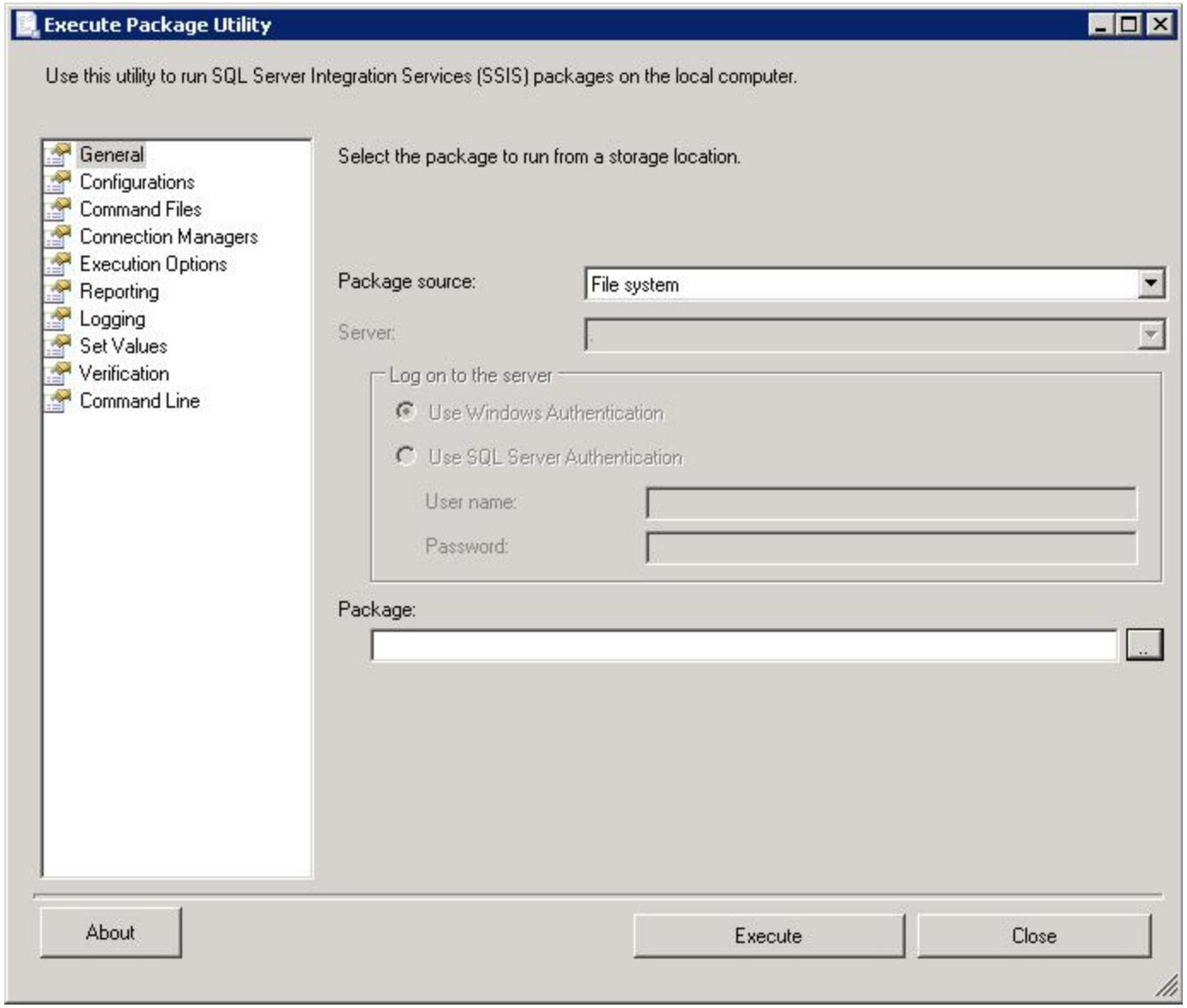

FIGURE 34: EXECUTE PACKAGE UTILITY INTERFACE

3. Use the **Package** ellipsis button to locate the SSIS **Package** file (Figure 35 below)

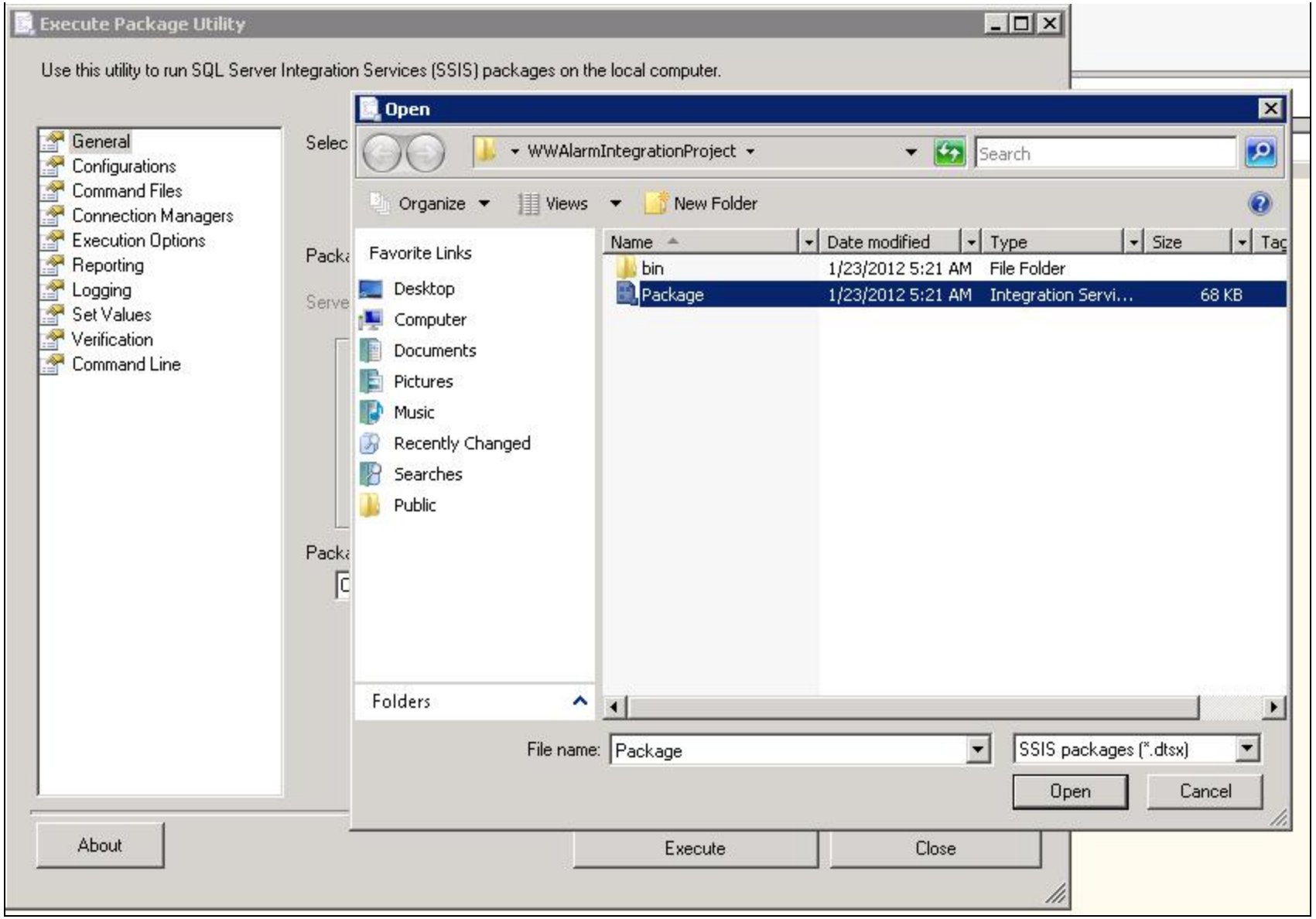

FIGURE 35: SELECT PACKAGE

4. Click **Open**, then **Execute**. The **Package Execution Progress** window appears (Figure 36 below).

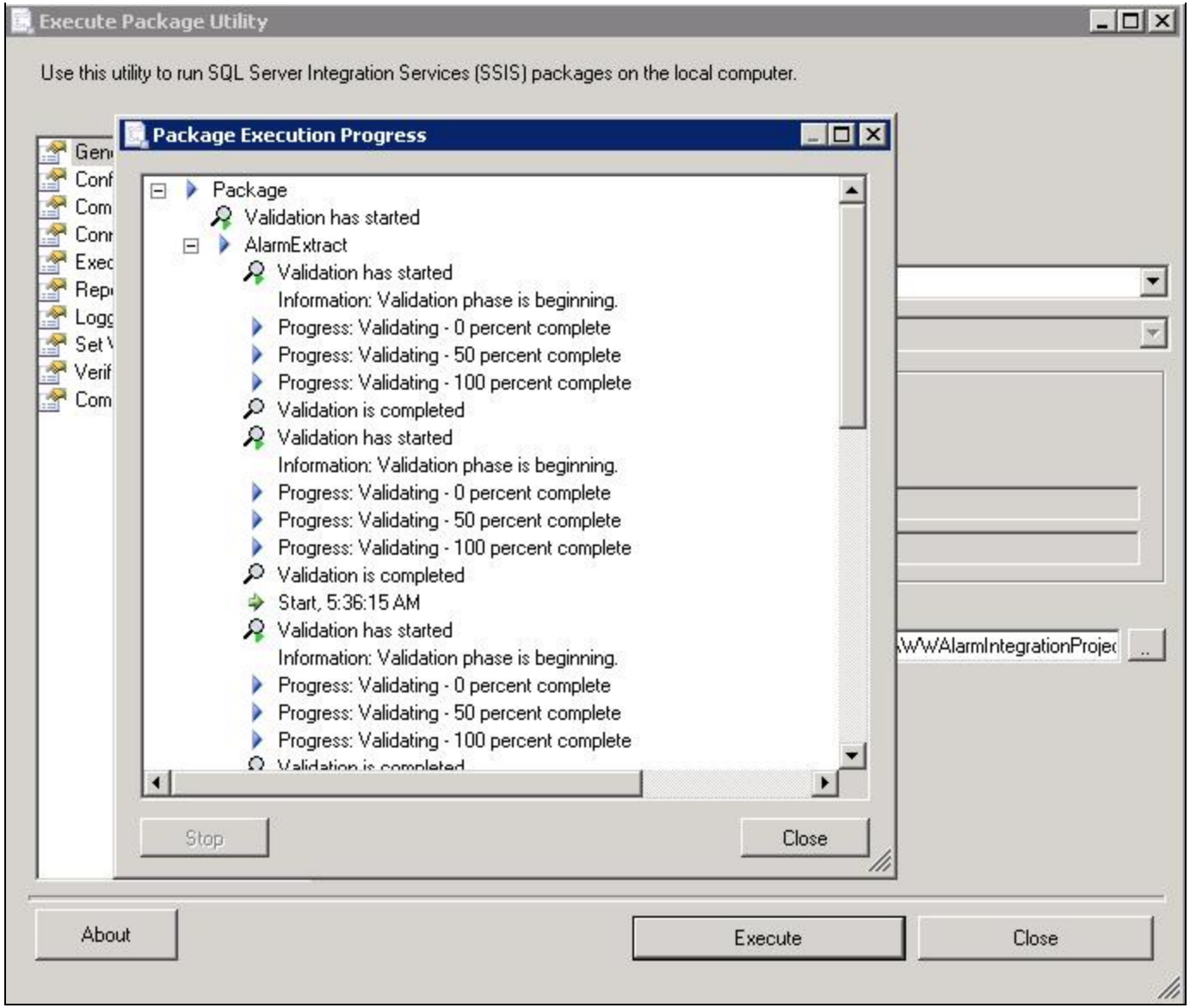

Figure 36: Package Execution Progress

Scheduling the SSIS Job

1. Open SQL Server Management Studio and click **SQL Server Agent**.
2. Right-click SQL Server Agent, then **Start** to start the SQL Agent (Figure 37 below).

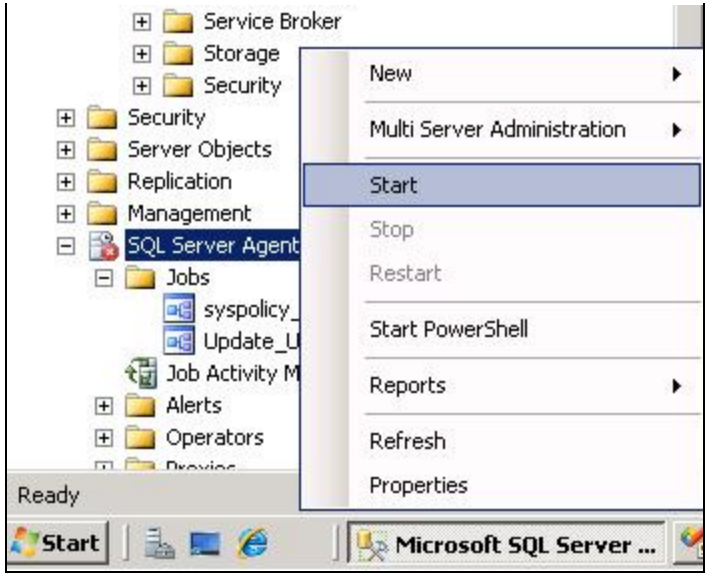

FIGURE 37: START THE SQL SERVER AGENT

FIGURE 38: CONFIRM START

3. Click **Yes** to start the SQL server Agent
4. Expand the SQL Server Agent item and click **Jobs**.
5. Right-click **Jobs** and click **New job** (Figure 39 below).

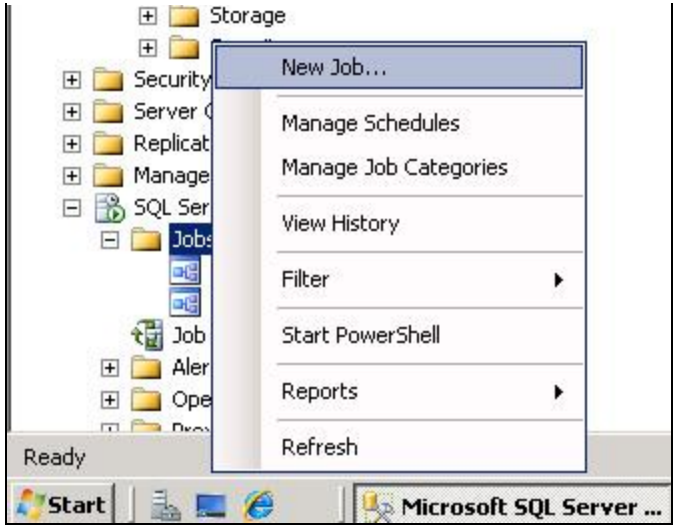

FIGURE 39: NEW JOB

6. Click the General tab and configure the following:
   - Job name,
   - The Job owner,
   - Category
   - Description (Figure 40 below)

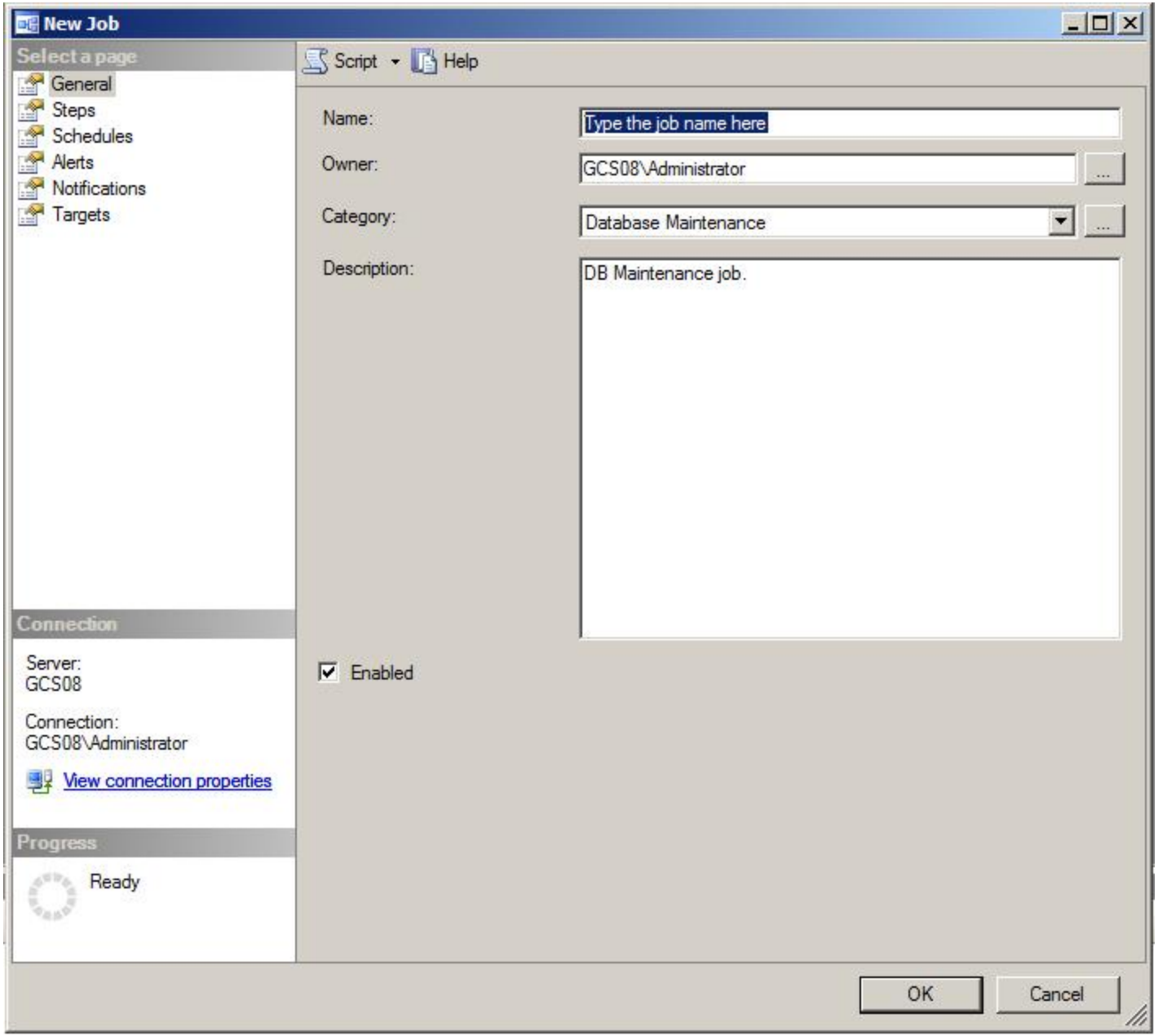

FIGURE 40: GENERAL JOB DATA

7. Click the **Steps** page and click New to create new job step (Figure 41 below).

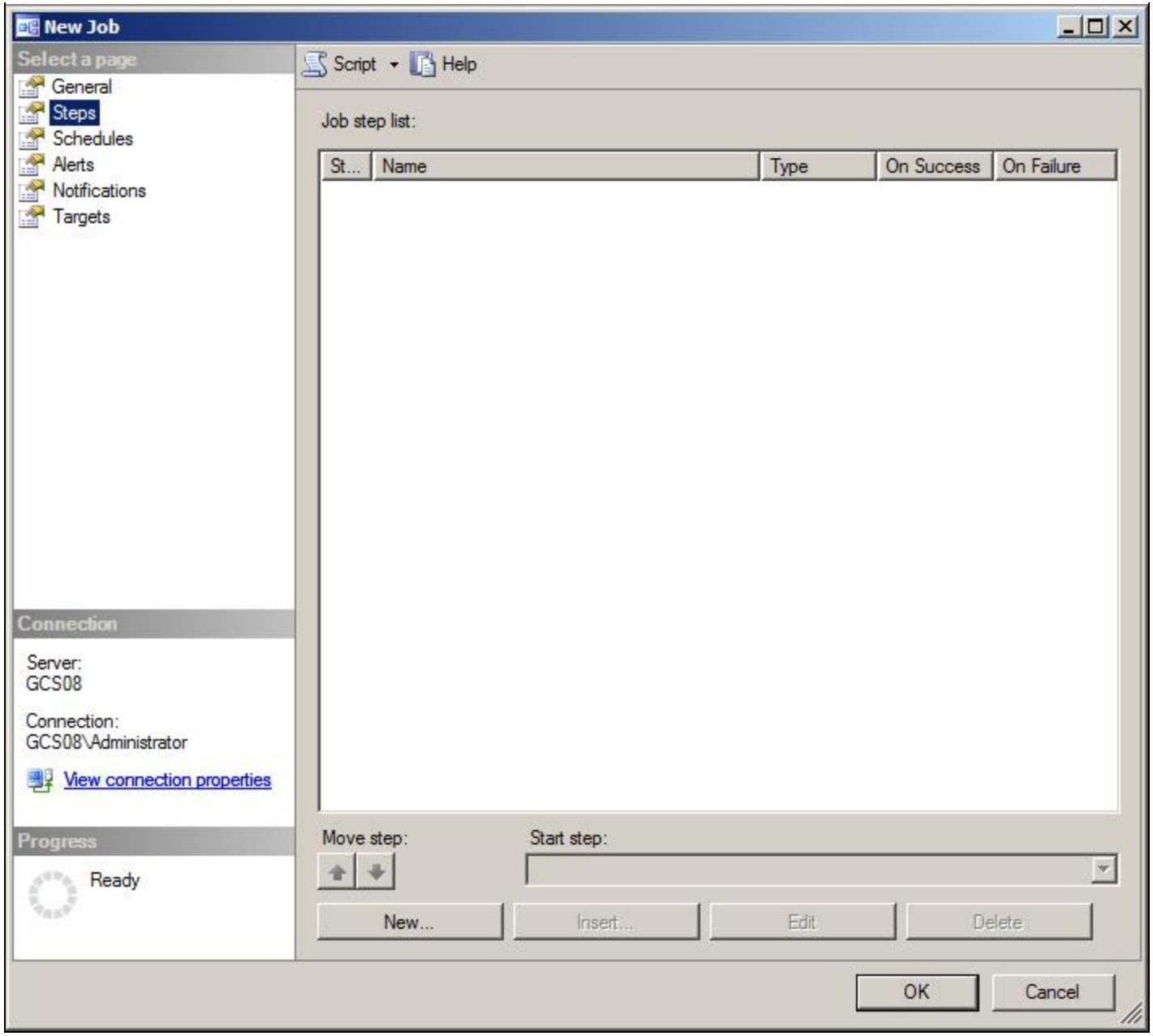

FIGURE 41: NEW JOB STEP

8. In the General page, type the **Step name**, select **SQL Server Integration Services Package** for the **Type**.

   The **SQL Server Agent Service Account** is selected by default for **Run as** (Figure 42 below).

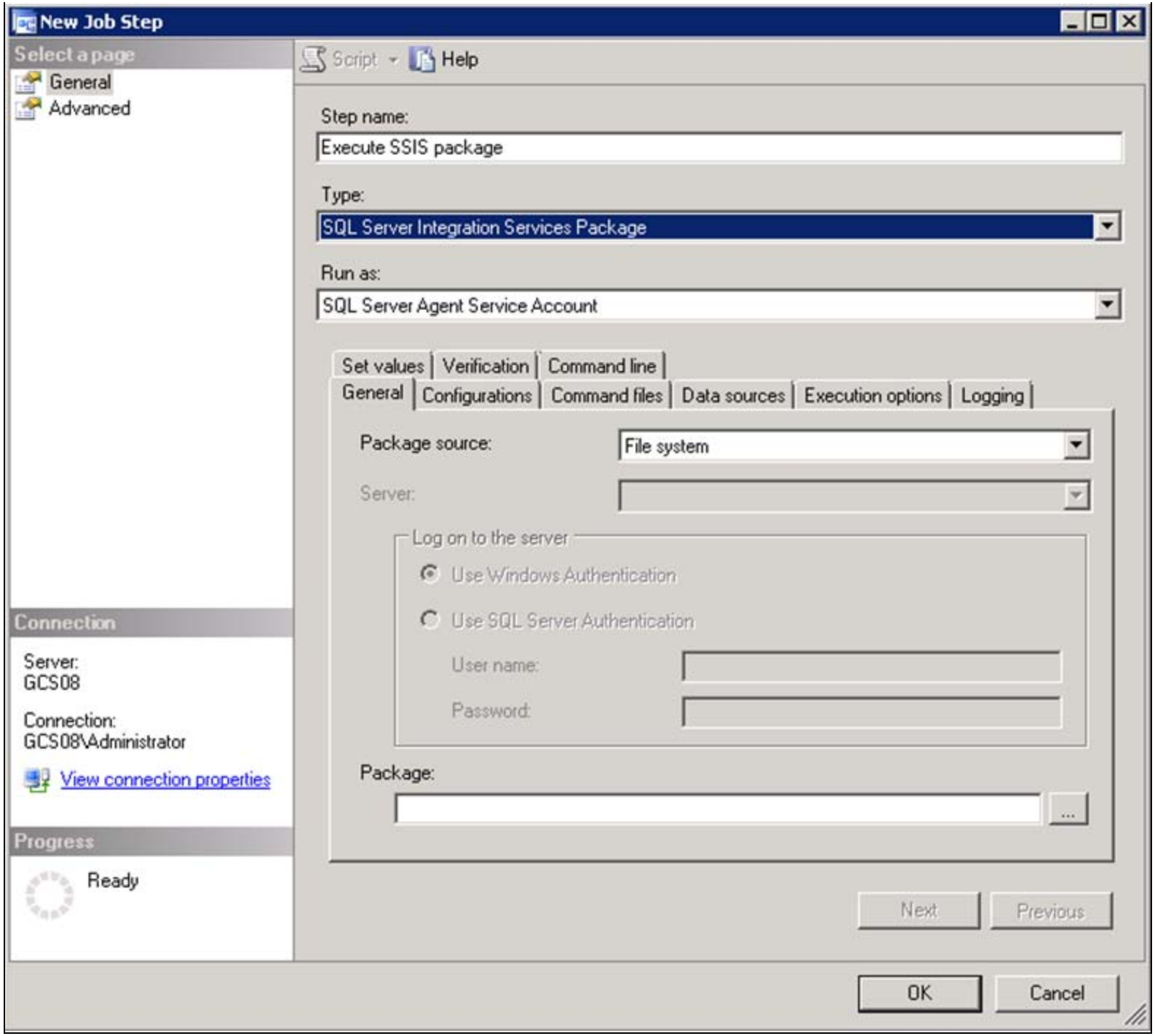

FIGURE 42: CONFIGURE A NEW JOB STEP

9. Select **File System** as the **Package Source** and provide the location of the SSIS package in the **Package** field (Figure 43 below).

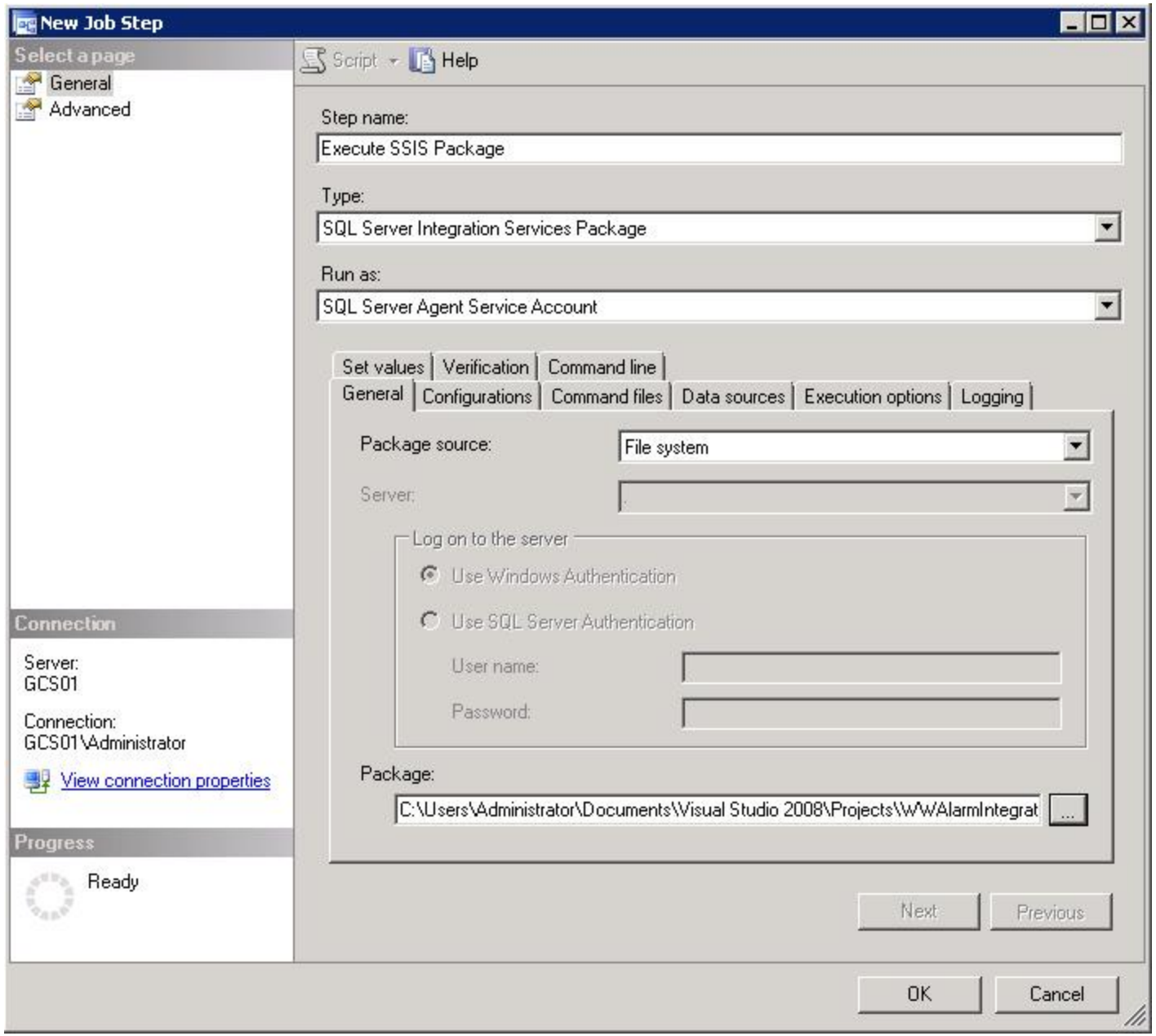

FIGURE 43: CHOOSE SQL SERVER AGENT SERVICE ACCOUNT FROM LIST

10. Click **Schedules** then right-click and click **New** to create a **Job Schedule** (Figure 44 below).

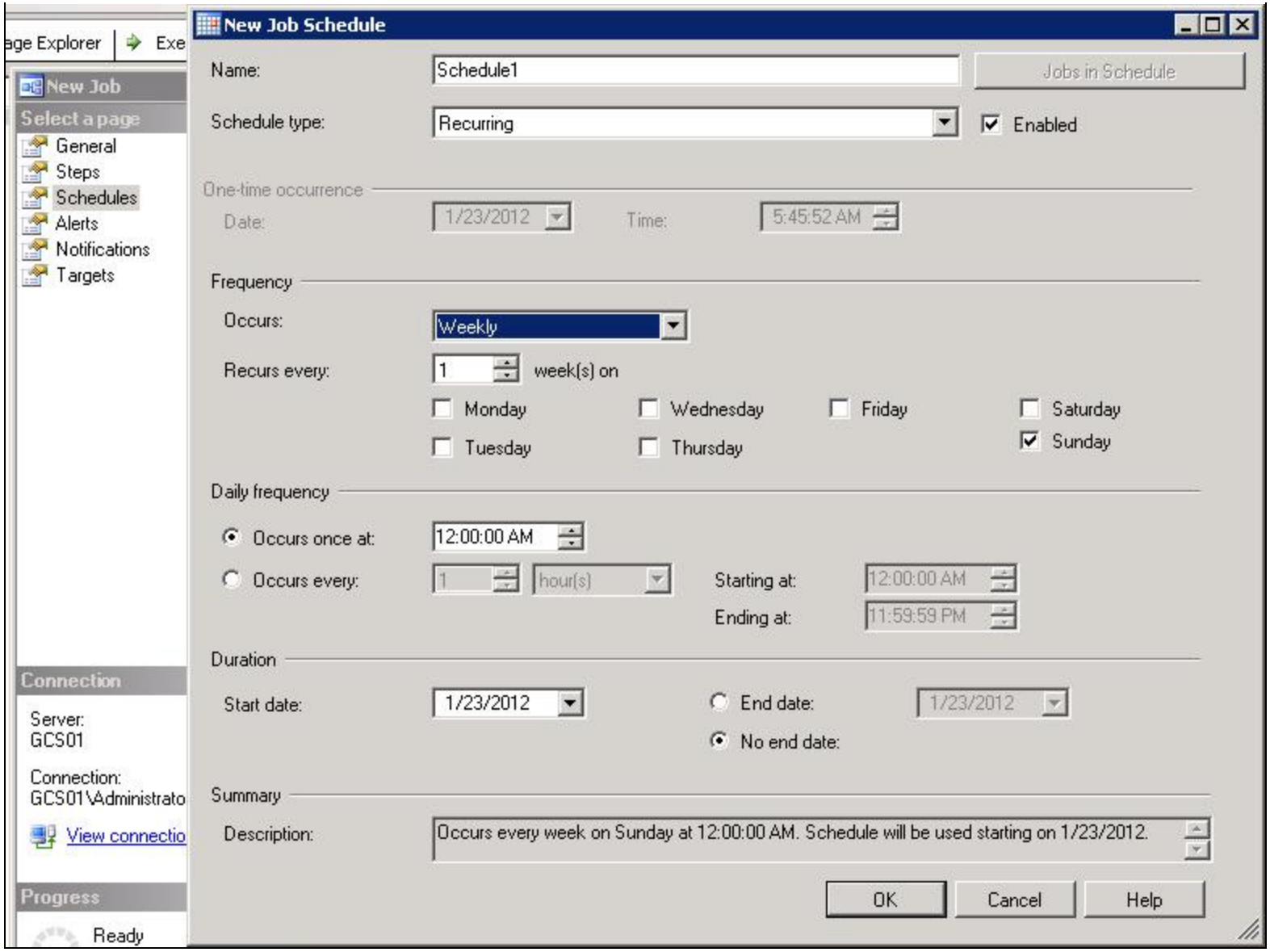

FIGURE 45: NEW JOB SCHEDULE

11. Click **OK** to save the job step settings.
12. Click **OK** again to save the SQL Server Agent Job.

R. Mahmoud

Tech Notes are published occasionally by Wonderware Technical Support. Publisher: Invensys Systems, Inc., 26561 Rancho Parkway South, Lake Forest, CA 92630. There is also technical information on our software products at **Wonderware Technical Support.**

For technical support questions, send an e-mail to **wwsupport@invensys.com**.

Back to top