

Integracja systemów

Laboratorium 6
Prowadzący: Marek Kowal
(M.Kowal@issi.uz.zgora.pl)

Ładowanie danych do celu typu OLE DB

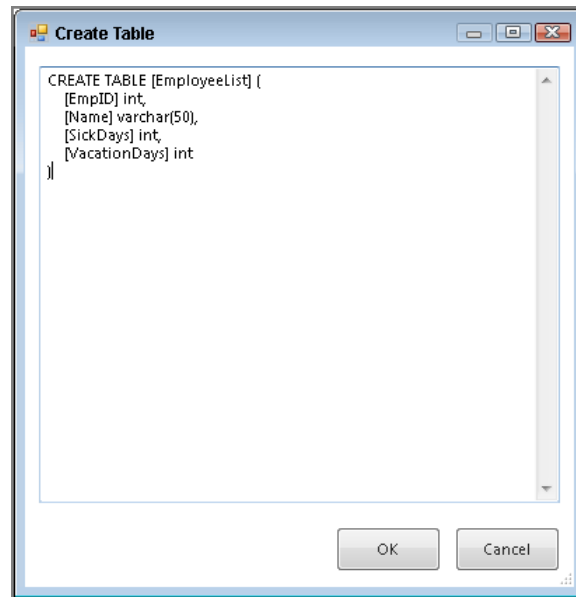
1. Cel ćwiczenia

Celem ćwiczenia jest załadowanie listy pracowników EmployeeList utworzonej podczas poprzednich ćwiczeń do nowej tabeli w bazie AdventureWorks2012.

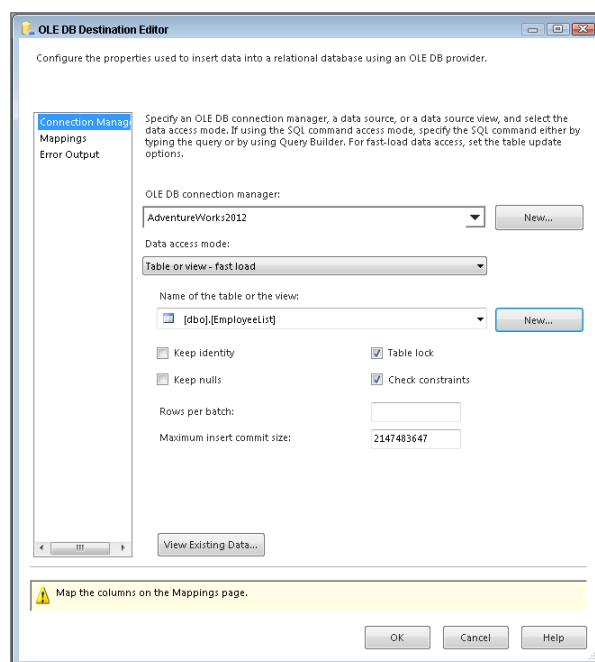
2. Przebieg ćwiczenia

- 1) Otwórz projekt i przypomnij sobie, do czego służyły pakiety utworzone podczas poprzedniego ćwiczenia
- 2) Otwórz Data Flow Task o nazwie Flat File Extract i przeciągnij na niego obiekt OLE DB Destination. Jeśli w oknie projektanta znajduje się obiekt Union All to go usuń. Zmień nazwę celu na EmployeeList
- 3) Połącz za pomocą niebieskiej strzałki Flat File Source z nowo utworzonym celem. Kliknij na nim dwukrotnie, aby uruchomić edytor.
- 4) Obiekt docelowy domyślnie zakłada wykorzystanie menadżera połączenia typu OLE DB, który został już wcześniej utworzony. Kliknij na przycisku New obok opcji Name of the table aby utworzyć tabelę, do której będą ładowane dane.
- 5) Wynikiem będzie pojawienie się okna dialogowego umożliwiającego wpisanie zapytania SQL w celu utworzenia tabeli (Rys. 1). Wpisz poniższe zapytanie i kliknij OK.

```
CREATE TABLE [EmployeeList] (  
    [EmpID] int,  
    [Name] varchar(50),  
    [SickDays] int,  
    [VacationDays] int  
)
```



- 6) Zauważ, że w oknie OLE DB Destination Editor pojawiło się ostrzeżenie, że należy jeszcze skonfigurować mapowanie (Rys. 2). Wybierz w tym celu opcję Mappings.
- 7) Na stronie Mappings pojawi się automatyczne dopasowanie na podstawie nazw kolumn pomiędzy źródłem i celem (Rys. 3).

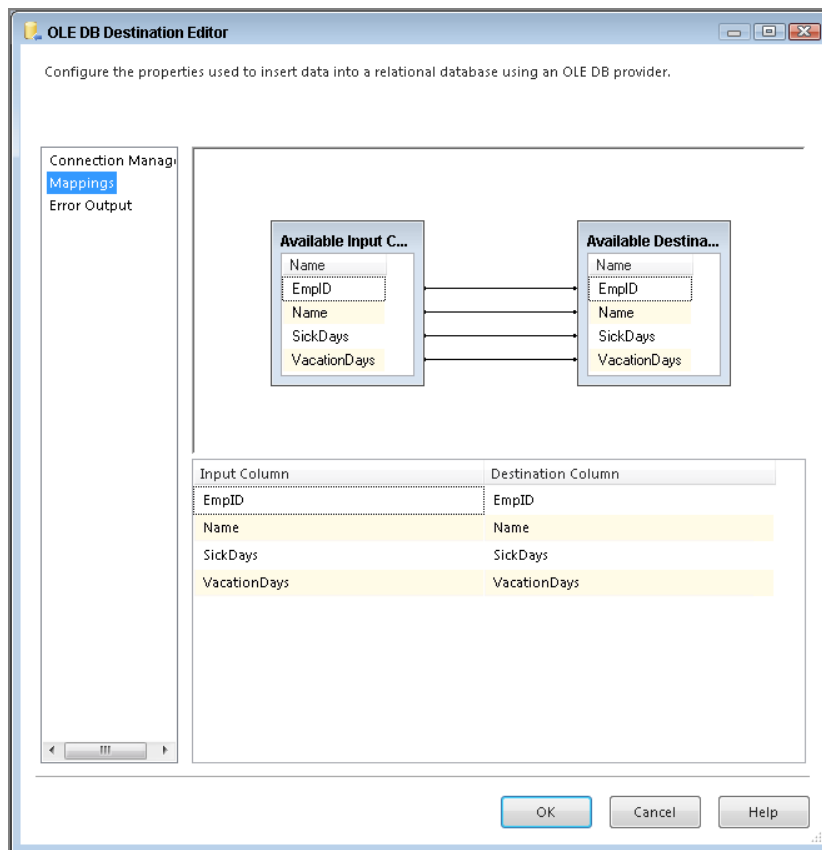


Rys. 2

- 8) Uruchom utworzony Data Flow Task sprawdź wyniki w tabeli docelowej (Rys. 4).

	EmpID	Name	SickDays	VacationDays
1	1	Devin Knight	4	10
2	2	Brian Knight	5	14
3	3	Mike Davis	0	9

Rys 4



Rys. 3

Flat File Destination

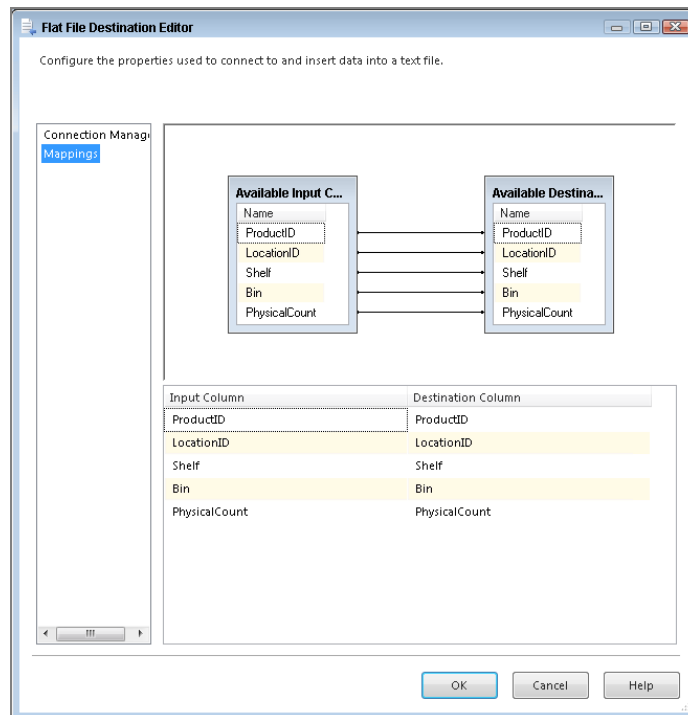
1. Cel ćwiczenia

Celem ćwiczenia jest załadowanie danych magazynowych z pliku excela do pliku płaskiego. Jako źródło danych wykorzystaj utworzony we wcześniejszym ćwiczeniu pakiet pobierający dane z pliku excel-owego.

2. Przebieg ćwiczenia

- 1) Otwórz Data Flow Task o nazwie Excel Extract i przeciągnij na niego obiekt Flat File Destination. Usuń obiekt Union All utworzony podczas poprzedniego ćwiczenia.
- 2) Połącz za pomocą niebieskiej strzałki źródło excel-owe z nowo utworzonym celem. Otwórz edytor celu.

- 3) Edytor zakłada domyślnie, że do połączenia z plikiem płaskim wykorzystany zostanie istniejący Flat File Connection Manager. Jednakże w tym przypadku, konieczne jest utworzenie kolejnego menadżera połączenia, dlatego kliknij na przycisk New
- 4) Wybierz typ pliku płaskiego, jako comma-delimited oraz wybierz miejsce na dysku do jego zapisu.
- 5) Po powrocie do Flat File Destination Editor wybierz opcję Mappings, aby zweryfikować, czy automatyczna propozycja mapowania kolumn jest prawidłowa (Rys. 1).



Rys. 1

- 6) Uruchom pakiet i sprawdź wyniki

Excel destination

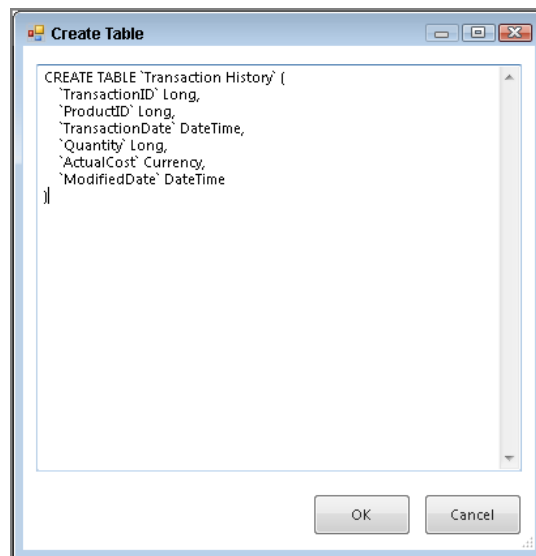
1. Cel ćwiczenia

Celem ćwiczenia jest załadowanie danych pochodzących z bazy AdventureWorks2012 do pliku excel-owego TransactionHistory.

2. Przebieg ćwiczenia

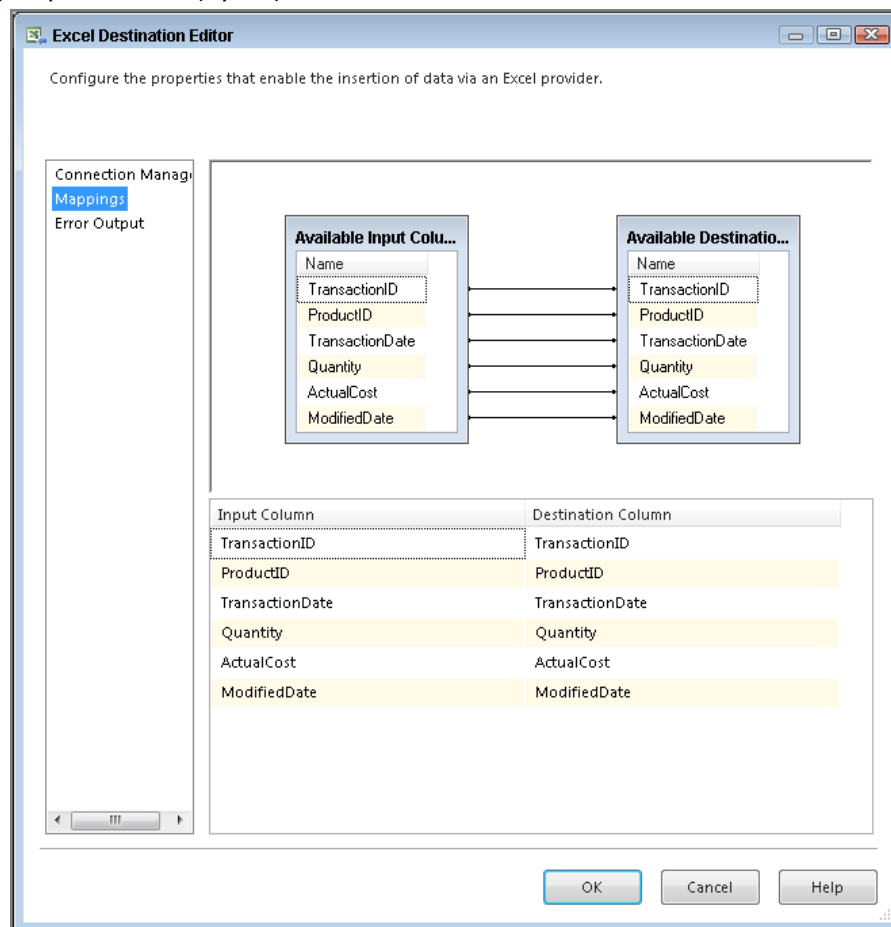
- 1) Otwórz Data Flow Task o nazwie OLE DB Extract, który utworzony został podczas poprzednich ćwiczeń. Przeciągnij obiekt Excel Destination i nazwij go TransactionHistory. Usuń obiekt Union All wstawiony podczas wcześniejszego ćwiczenia.
- 2) Połącz za pomocą niebieskiej strzałki źródło OLE DB z nowym celem i otwórz edytor celu.

- 3) Edytor domyślnie wykorzystuje utworzony wcześniej menadżer połączeń do plików Excel-a. Naciśnij przycisk **New** obok opcji **Name of the Excel sheet**, aby utworzyć skoroszyt Excel-a za pomocą automatycznie wygenerowanego zapytania (Rys. 1)



Rys. 1

- 4) Po powrocie do Excel Destination Editor upewnij się w opcji **Mappings** czy mapowanie kolumn jest prawidłowe (Rys. 2)



Rys. 2

- 5) Uruchom zadanie i sprawdź wyniki

Data Conversion Transform

1. Cel ćwiczenia

Celem ćwiczenia jest import zawartości pliku Inventory Worksheet.xls do tabeli bazy AdventureWorks2012, przy czym, przed załadowaniem danych należy dokonać konwersji typów danych dla poszczególnych kolumn.

2. Przebieg ćwiczenia

- 1) Otwórz projekt realizowany podczas poprzedniego ćwiczenia i utwórz nowy pakiet o nazwie lesson20.dtsx
- 2) Create a new Excel Connection Manager using the Inventory Worksheet.xls file and make sure the First row has column names option is checked.
- 3) Drag a Data Flow Task onto the design pane and name the new task DFT–Data Conversion.
- 4) In the Data Flow tab, drag a new Excel Source onto the Data Flow design pane and name it Excel SRC-Inventory Worksheet.
- 5) Double-click the Excel Source and change the OLE DB Connection Manager option to your only connection manager. Then change the Name of the Excel sheet option to Inventory_Worksheet and click OK.
- 6) Drag a Data Conversion Transform onto the design pane and connect it to the Excel Source.
- 7) Open the Data Conversion Transformation Editor by double-clicking the new transform and check each column from the Available Input Columns table. Change the Output Alias of all columns to ConvertedColumnName, as shown in Figure 1
- 8) For the Data Type, select string [DT_STR] for the Input Column Shelf and four-byte signed integer [DT_I4] for all other columns and click OK.
- 9) Back in the designer, drag an OLE DB Destination onto the design pane and connect it to the Data Conversion Transform.
- 10) Open the OLE DB Destination and click New next to the connection manager selection to create a new OLE DB Connection Manager, where you will select AdventureWorks2012.

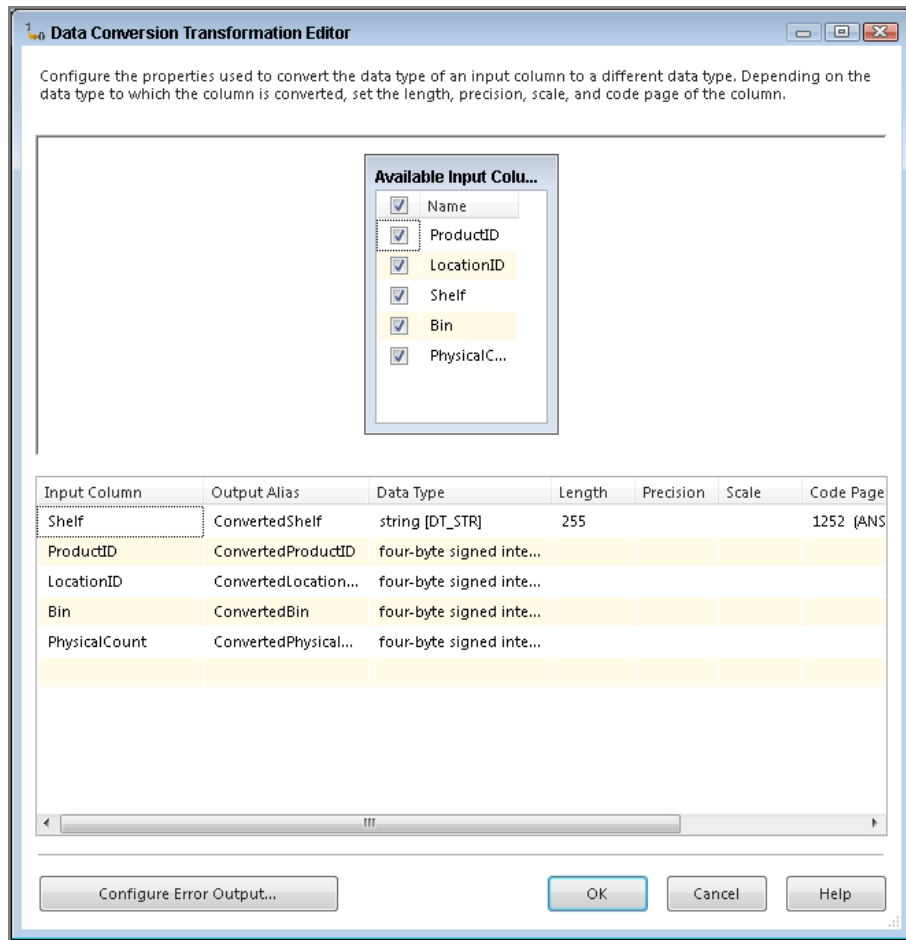
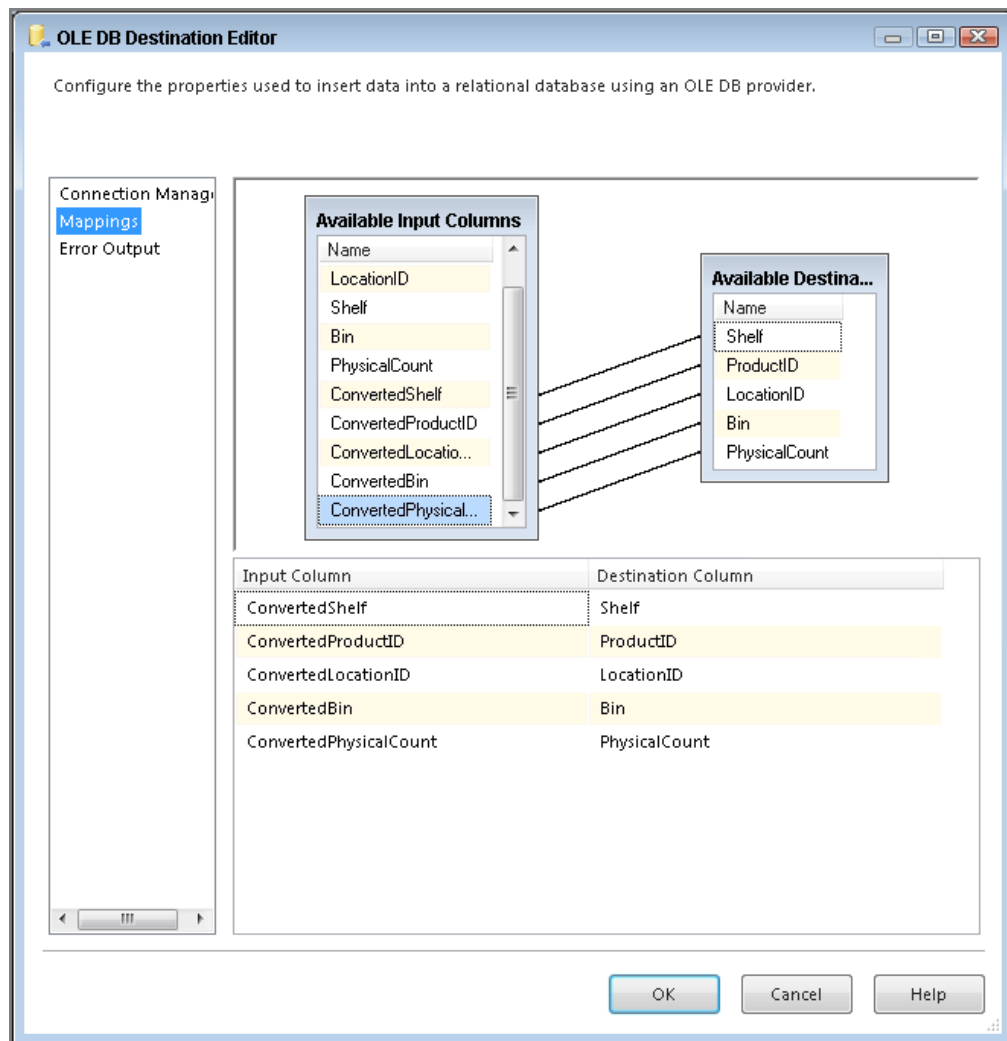


Fig. 1

- 11) Still in the OLE DB Destination Editor, click New next to the table selection to create a new table and ensure the following statement is used:

```
CREATE TABLE [ShelfInventory] (
    [Shelf] varchar(255),
    [ProductID] int,
    [LocationID] int,
    [Bin] int,
    [PhysicalCount] int
)
```

- 12) Go to the Mappings page and delete all connections between the Input Columns and Destination Columns. Now connect all Input Columns with the Converted prefix to the associated Destination Columns (Figure 2) and click OK.
- 13) Execute the package and check results



Rys. 2

Derived Column Transform

1. Cel ćwiczenia

The Derived Column Transform enables you to either create or replace a column in the data stream. You can use this component for many problems you may run into and, therefore, it is one of the most useful tools you have in the Data Flow. Your company decides that it would be best to include the date on which each row is populated in your SSIS package from Lesson 20. Your manager tells you that this date is necessary for auditing purposes. Once you have made these changes to the package, delete the content of the table before you run the package again.

2. Przebieg ćwiczenia

- 1) Open a query window in Management Studio and run this query to empty the table's data:

```
TRUNCATE TABLE ShelfInventory
```

- 2) Now run the following query to add a column to the ShelfInventory table:

```
ALTER TABLE ShelfInventory  
ADD RowStartDate datetime
```

- 3) Open the SSIS package Lesson20.dtsx that you created in the previous lesson
- 4) Click the Data Flow tab and delete the precedence constraint between the Data Conversion Transform and the OLE DB Destination.
- 5) Drag a Derived Column Transform into the Data Flow and connect it between the Data Conversion Transform and the OLE DB Destination.
- 6) Open the Derived Column Transformation Editor and add a new column by typing RowStartDate in the Derived Column Name property. Then, in the Expression property, add the system variable @[System::StartTime] by dragging it down from the variables list in the top-left section of the editor, as shown in Figure 1. Then click OK. This adds the current date and time to the column when the package is run.

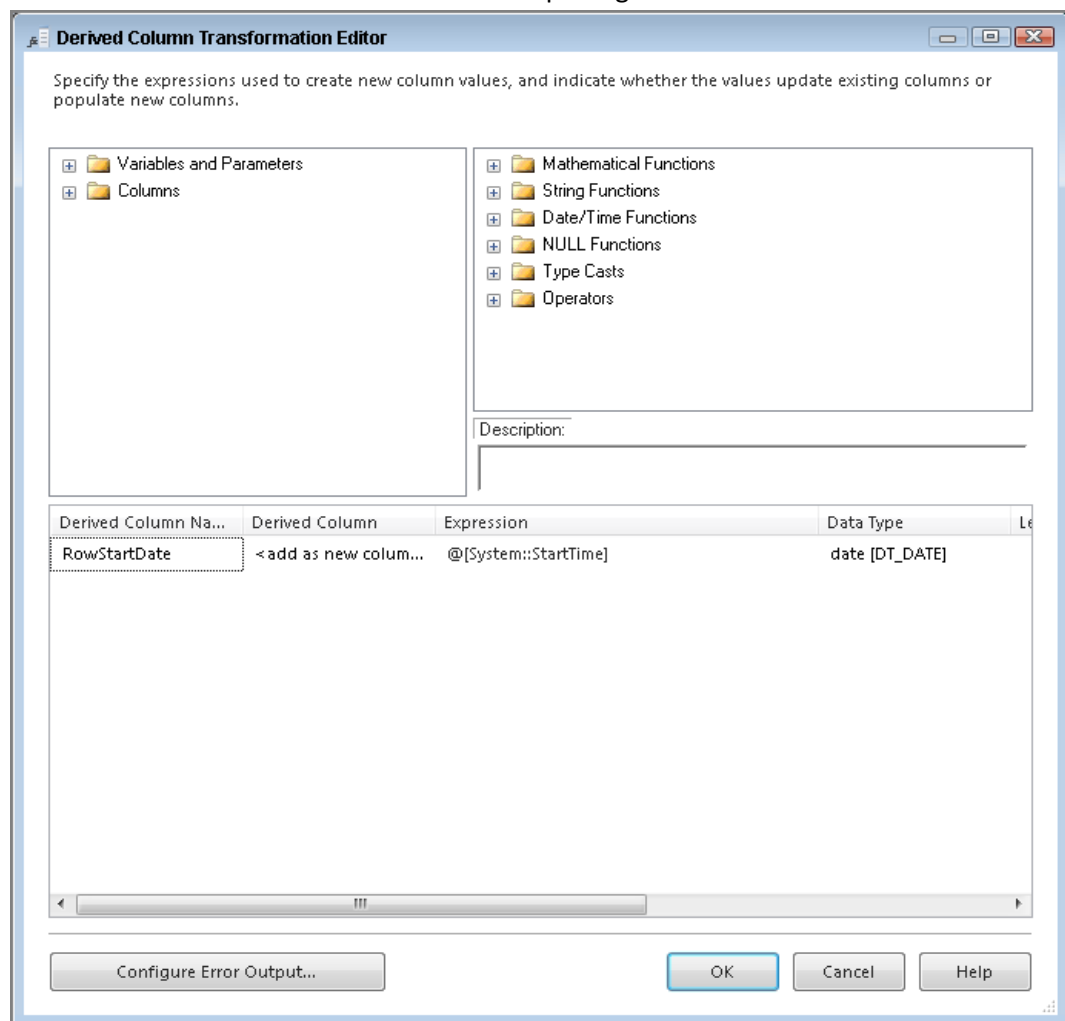


Fig. 1

- 7) Now that this column has been added, you need to make sure the destination knows to use it. Open the OLE DB Destination Editor and add the column RowStartDate to the mapping, as shown in Figure 2
- 8) Now execute the package. The ShelfInventory table has now been repopulated with the new column that holds the date and time the package was run

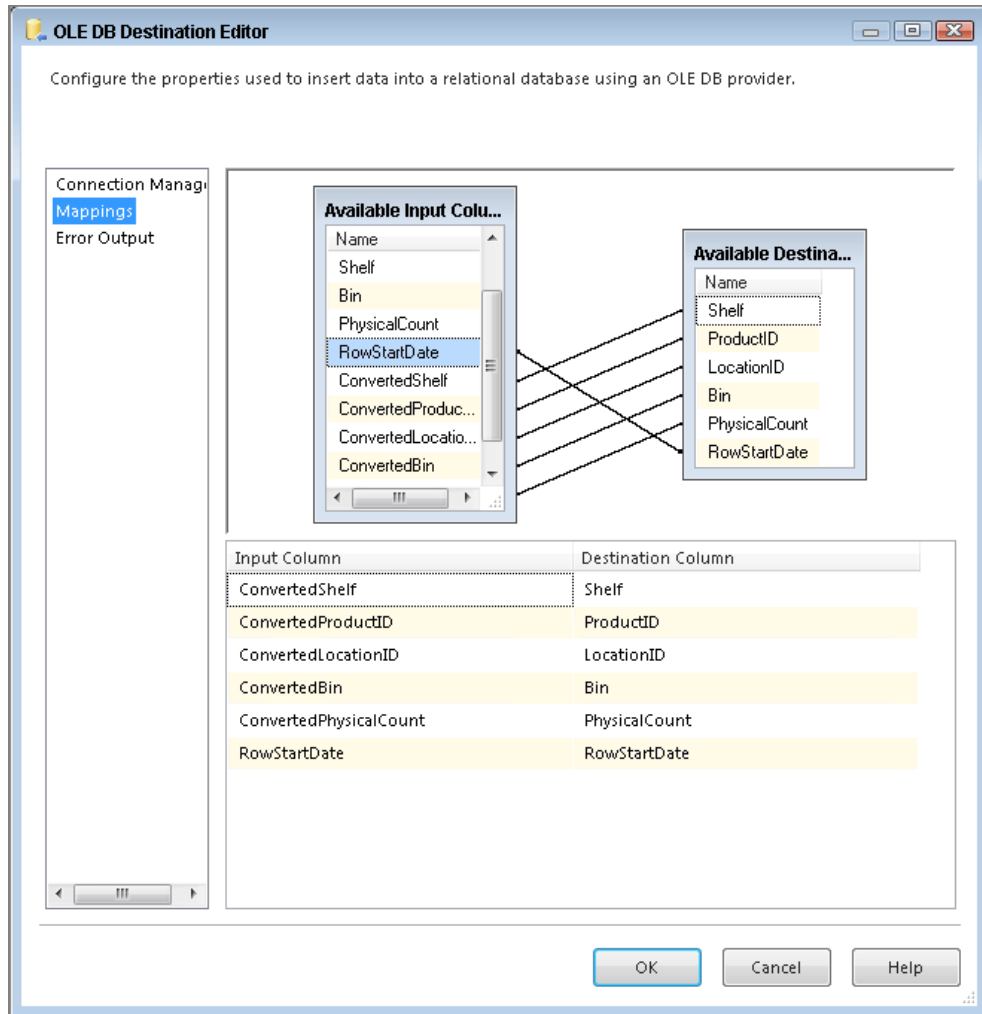
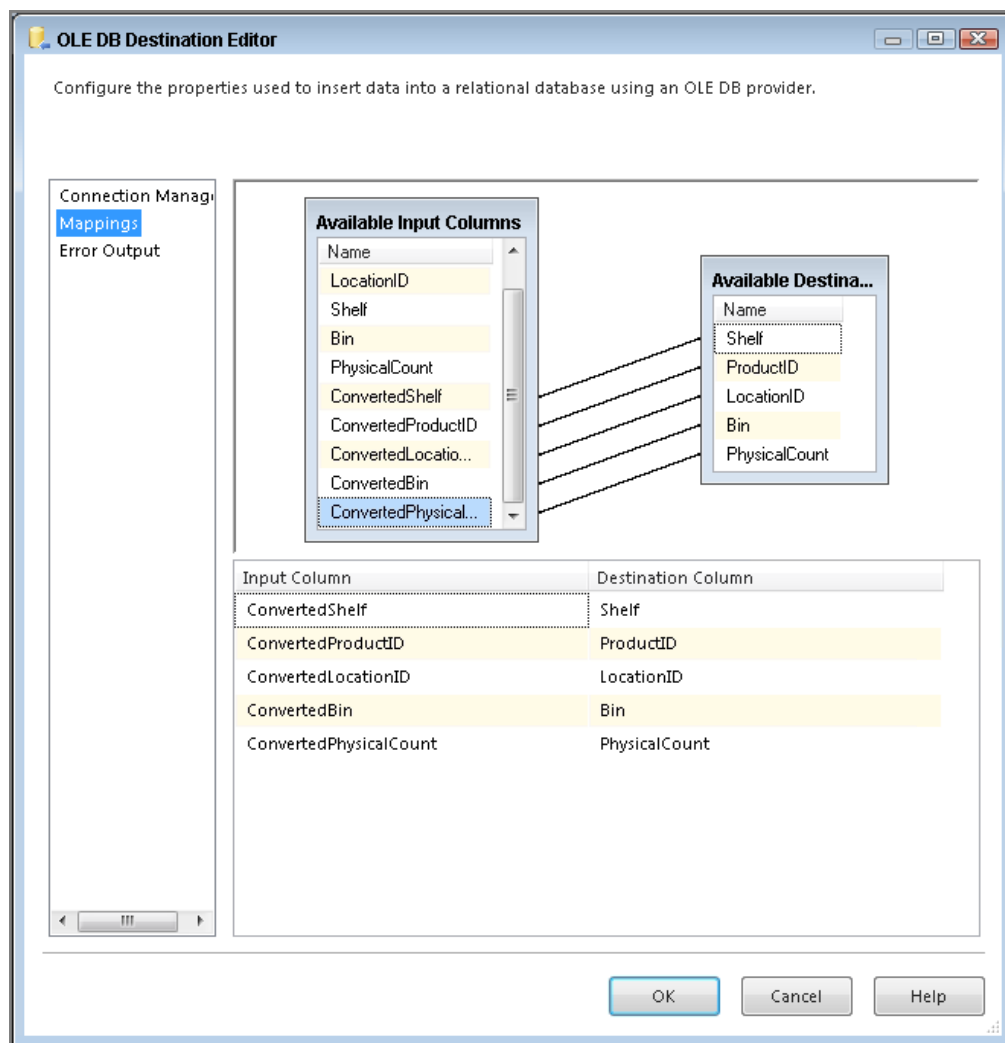


Fig. 2



Rys. 2

Bibliografia

- 1) Knight B., Knight D., Davis M., Snyder W. (2013): Knight's Microsoft® SQL Server® 2012 Integration Services 24-Hour Trainer, John Wiley & Sons.
- 2) Knight B., Veerman E., Moss J.M., Davis M., Rock C. (2012): PROFESSIONAL Microsoft® SQL Server® 2012 Integration Services, John Wiley & Sons.
- 3) <http://www.wrox.com/WileyCDA/Section/id-814197.html>
- 4) [https://msdn.microsoft.com/library/ms169917\(SQL.120\).aspx](https://msdn.microsoft.com/library/ms169917(SQL.120).aspx)
- 5) Tok W-H., Parida R. Masson M. Ding X. Sivashanmugam (2012): Microsoft SQL Server 2012 Integration Services, Promise (tłumaczenie j. polski).
- 6) Kimball R. (2004): The Data Warehouse ETL Toolkit. John Wiley & Sons