by experts
for experts

Transformation Manager
Version 5.2

Tutorial 3 - .Notation

Transformation
Manager

Copyright Notice

All information contained in this document is the property of ETL Solutions Limited. The information contained in this document is subject to change without notice and does not constitute a commitment on the part of ETL Solutions Limited. No part of this document may be reproduced in any manner, including storage in a retrieval system, transmission via electronic means or other reproduction medium or method (electronic, mechanical, photocopying, recording or otherwise) without the prior written permission of ETL Solutions Limited.

© 2013 ETL Solutions Limited. All rights reserved.

All trademarks mentioned herein belong to their respective owners.

Table of Contents

Tutorial 3 1

.Notation 1

Prerequisites 1

Concepts 1

Information 1

Learning Objectives 1

Using a .Notation approach 1

Using [idx] 2

Using [1] 2

Exercise 1 - Start TM Designer and Open Your Repository 3

Option 1 - Using the File menu bar option Open Repository 3

Option 2 - Using the File menu bar option Open Recent Repositories 4

Exercise 2 - Create a New Project 4

Exercise 3 - Create a New Transform 5

Exercise 4 - Write the .Notation Transform 7

Exercise 5 - Build The Project 9

Exercise 6 - Run the Project 10

Tutorial 3

.Notation

This tutorial demonstrates how to complete a transformation task by using only one transform. We will use the .Notation method as an alternative to independent and dependent transforms.

During this tutorial we will also make use of index variables. The tutorial will demonstrate how to locate and identify related data on both the source and target sides using indexes. Used in this tutorial are two indexes, [1] and [idx] to help us in this task.

Prerequisites

Before starting this tutorial we recommend that you have completed the following tasks.

1) Transformation Manager has been installed.
2) An appropriate license has been installed.
3) The tutorial resources including data models, samples and source and target data stores have been downloaded and extracted to your Transformation Manager home directory.
4) You should have completed tutorials 1 and 2.
5) You may need to load the Derby BasicWriters data store and the BasicWriters XML data store which will be called WRITERS.

Concepts

This tutorial makes use of two ideas within Transformation Manager. The first is the use of .Notation which refers to the traversing of element relationships using the path defined to locate the required attribute or relationship. This gives rise to the term .Notation because of the separator used between elements and attributes, a .(dot). The second is the use of indexing to identify the instance of an element used in your transform.

Information

This tutorial uses two data models. The source will be the XML data model created in Tutorial 2 from the exercise called **Load an XML Data Model**. The second model will also be from Tutorial 2 from the exercise called **Load a Database Using a Derby database**. If you haven't loaded these data models then follow the exercises in Tutorial 2 to do so. The aim of the tutorial is to take two elements in the XML data model and transform those to three elements in the target data model but all within a single transform.

The tutorial uses a technique called .Notation to do the transformation task. The benefits of this approach can be efficiency and performance but it may use excessive resources for large databases. Cascades or linked transforms can provide better debugging opportunities in complex transforms due to more breaks in the process.

Learning Objectives

Using a .Notation approach

You will learn how to update several target elements and attributes from within one transform.

Using [idx]

You will learn how to use the [idx] style of index to identify particular related data elements on the target side of the transformation that are being created.

Using [1]

You will learn how to use the [<integer>] style of index, for example [1], to locate a source side related element that you wish to retrieve data from.

Exercise 1 - Start TM Designer and Open Your Repository

If you are continuing from tutorial 1 or 2 then you probably have TM Designer open so ignore this exercise. This exercise will start TM Designer and open the tutorials repository ready to create a new project.

1) Looking at your desktop, find the icon that looks like this,

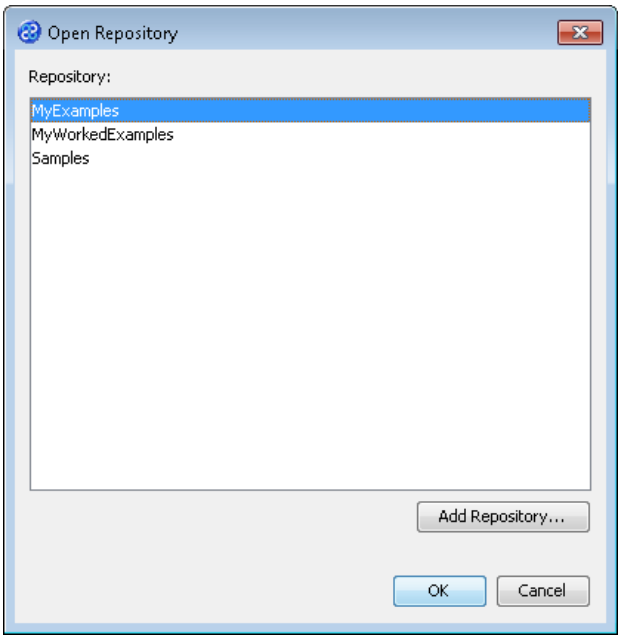
.
2) Then you have two options.
   a) Using your mouse double-click the icon using the primary mouse button.
   b) Using your mouse press the secondary mouse button, commonly the right-mouse button, while the cursor is over the icon to open the pop-up menu and select Open from the available options.
3) Once TM Designer has opened and displays its interface connect to your repository, **MyExamples**. There are two ways in which you can open a repository.

TM Designer by default displays the interface in the same state as it was left in by you when you last closed the application. So, you will not normally need to connect to the repository as it should open when you start the application.

Option 1 - Using the File menu bar option Open Repository

Transformation Manager has two menu options you can use, Open Repository... or Open Recent Repositories both of which are available from the File option on the application menu bar.

a) Click once on the File menu bar option.
b) Click once on the Open Repository... menu option. The **Open Repository** window will open listing the available repositories.



c) Click on the repository in the list that you wish to open; in our case **MyWorkedExamples**. That repository will be highlighted.

d) Click on the OK button. The **Open Repository** window will close and the repository will be opened.

Option 2 - Using the File menu bar option Open Recent Repositories

a) Click once on the File menu bar option.

b) Click once on the Open Recent Repository menu option to list the recent repositories that have been used.

c) Click once on the repository you wish to open from the list presented; in our case this will be **MyWorkedExamples**. The menu will disappear and the repository will open.

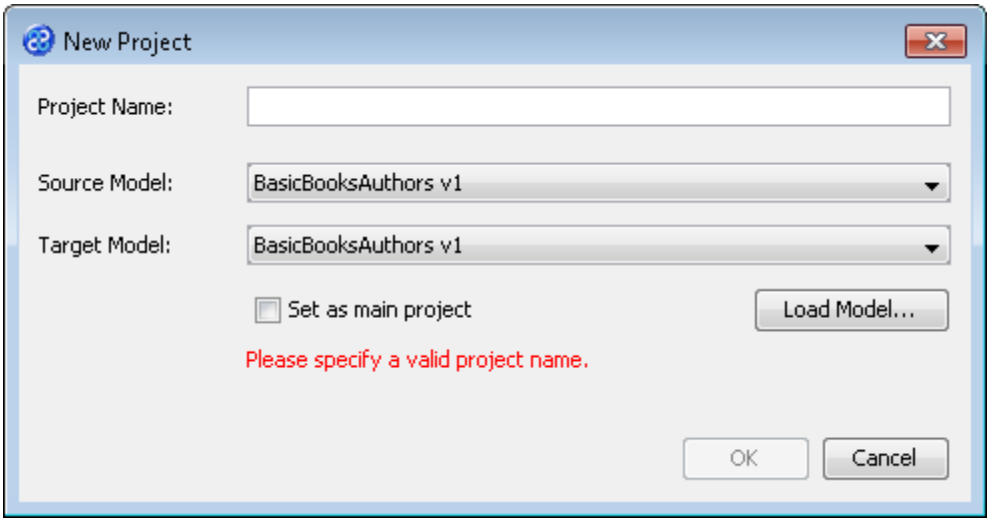
Exercise 2 - Create a New Project

Ensure you have the two required data models loaded in your repository. This exercise creates a new project where we will use the XML data model called **WRITERS v1** as the source data store and a relational database data store called **BasicWriters v1** as the target data model.

1) Using your mouse click on the File option from the menu bar of TM Designer.

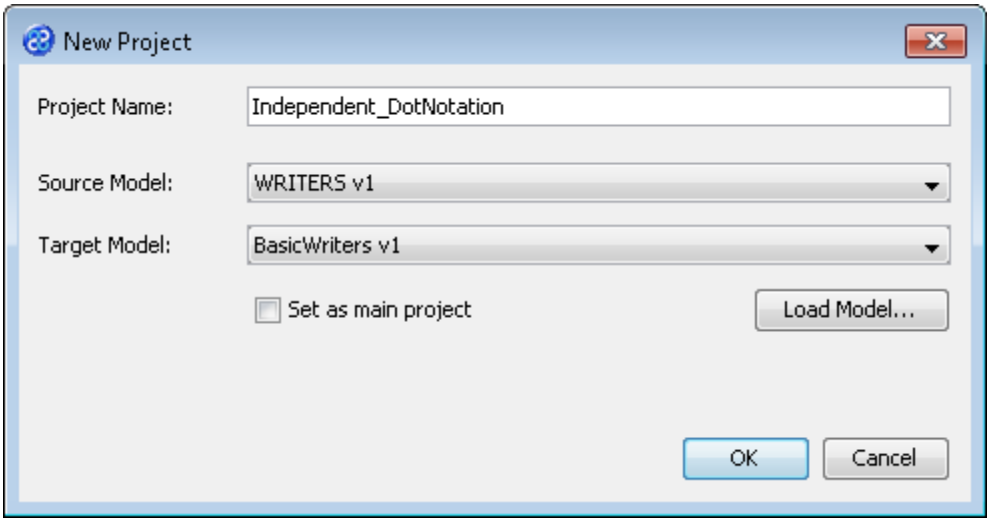
2) From the menu, click once on the New Project... option.

3) The **New Project** window will open.



4) In the **Project Name** field we will provide a name for the project. Let's call our project **Independent_DotNotation** by typing the name into the field.

5) Now we must select the target and source models for our project. In the **Source Model** field select **WRITERS v1** from the list. In the **Target Model** field select **BasicWriters v1**.



6) Now click once on the OK button. Your project will be created and be displayed in the **Projects** pane.

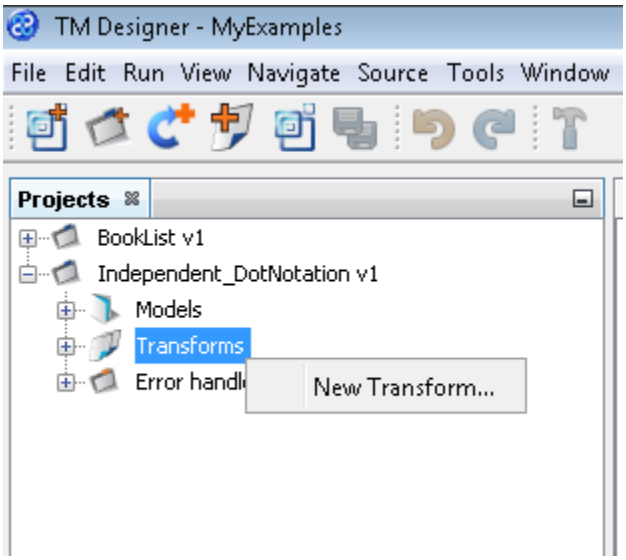
Exercise 3 - Create a New Transform

Let's create a new transform in our project.

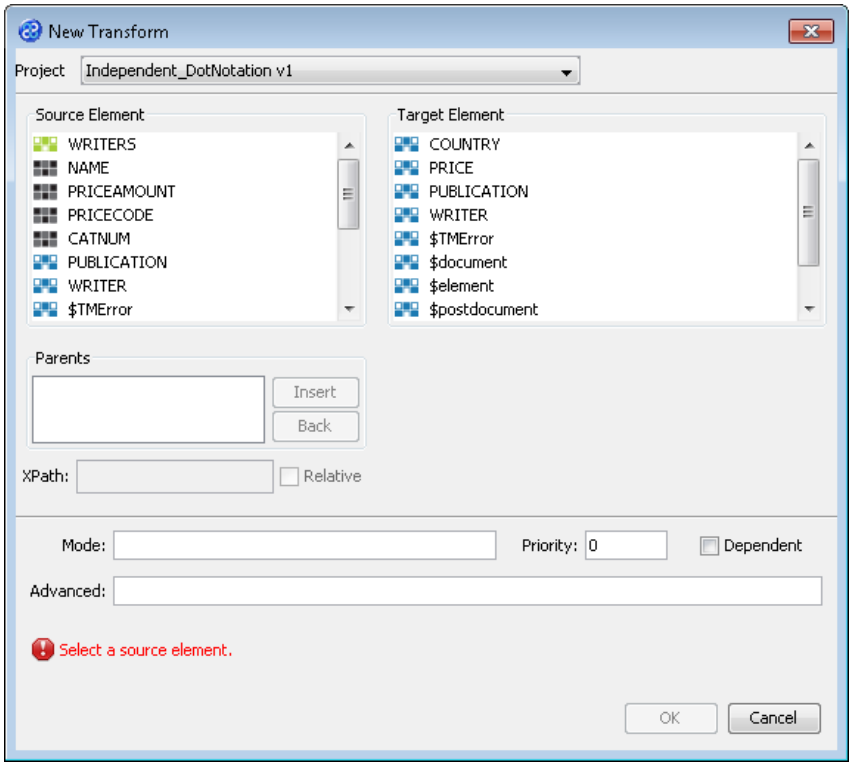
1) Go to the Project pane and find the **Independent_DotNotation v1** project.

2) Click on the icon to expand the list of sub-folders for the project if it is not already expanded.

3) Move your mouse pointer so that it is over the **Transforms** folder and use the right-mouse or secondary mouse button to display the context menu. Select New Transform... from this menu.

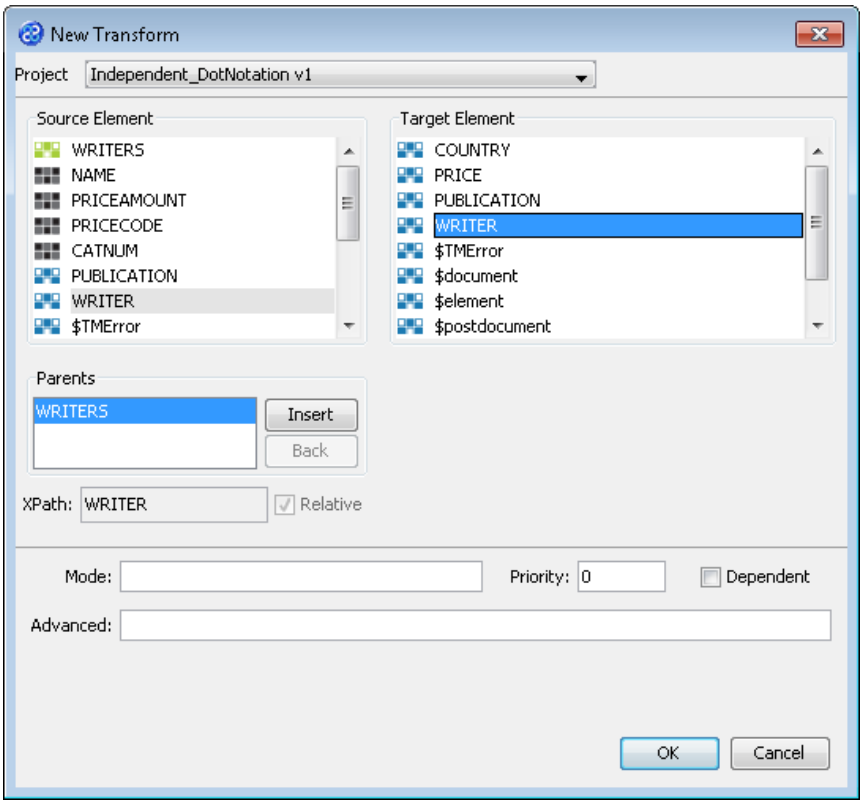


4) This will open the **New Transform** window as shown below.



5) Ensure that the correct project is selected in the **Project** drop down list box. The **Independent_DotNotation v1** project should be showing. If it is not, drop the list box down and select it from the list.

6) Now we need to select the required source and target elements in both of the data models. Ensure that you have selected the **WRITER** element in both the data models. We will not be using any of the other options so the window will look like the one below.



7) Click once on the OK button to create your transform.

8) The Editor pane will open ready for you to write your transform code once the **New Transform** is created. The Editor pane will display the name of your project as its title label. Note that the **Transform Source** and **Transform Target** panes will open populated with the relevant content.

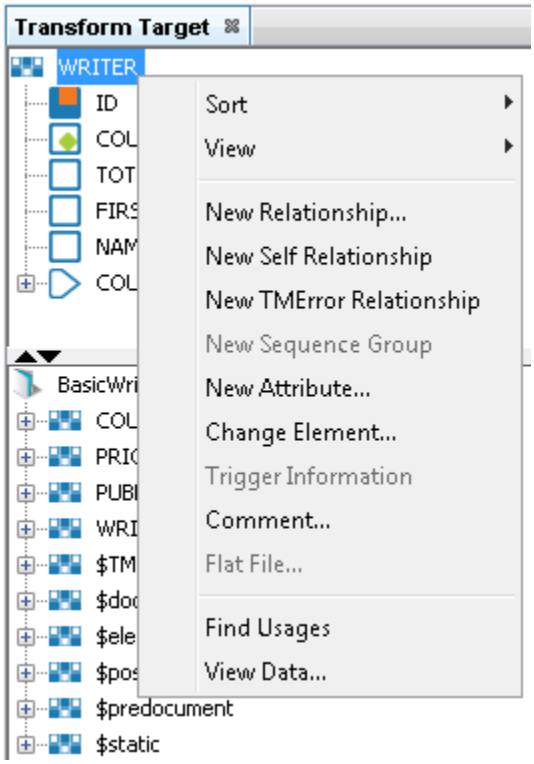
Exercise 4 - Write the .Notation Transform

Let's add our transform code. As you move through the exercise note the use of indexes in the transform. In this example the index will cause the transform to use the first valid value it finds in the source irrespective of any other candidate values it may find afterwards. We will be using drag and drop functionality to build our transform code up.

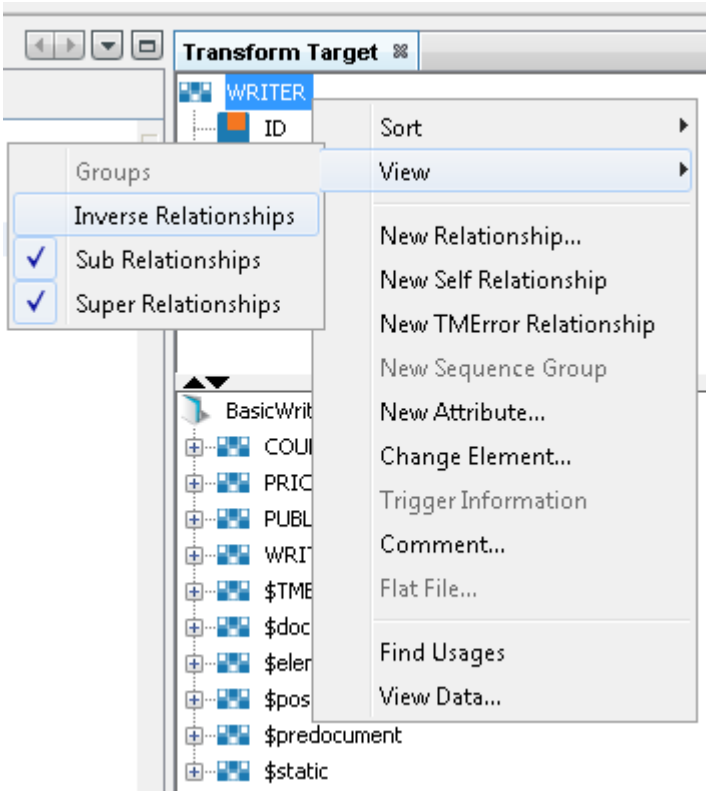
1) Let's add the first section of our transform code to the Editor pane. The code is shown below and does a direct transform between the **WRITER** elements in the two data models.

```
<ID> := <id>;
NAME := name;
```

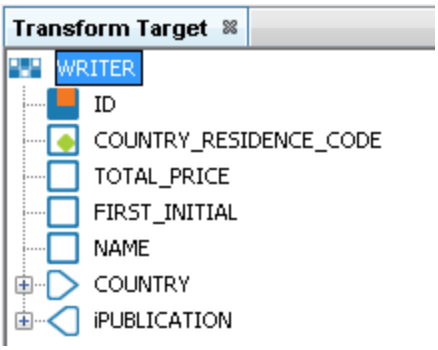
2) Now we need to display the inverse relationships for the target data model so we can continue to drag and drop attributes between the two data models. Place the mouse pointer over the target element called **WRITER**. Use the secondary or right mouse button to display the context menu as shown below.



3) Click on the View option and from the sub-menu select View Inverse Relationships.



4) You will now see the inverse relationship called **iPUBLICATION**. Note that the direction of the arrow is the reverse of the forward relationship called **COUNTRY**.



5) Click on the icon to expand the **iPUBLICATION** group and see the available attributes. Now you can assign the source attributes to the relevant target attributes.

6) Let's add our second section of code. This will add the transform code for the **PUBLICATION** element. You may need to expand the **PUBLICATION** relationship in the **Transform Source**. This will transform the relevant data in the PUBLICATION element of the source data model into the PUBLICATION element of the target data model.

```
iPUBLICATION[idx].CATNUM := PUBLICATION[1].CATNUM;
iPUBLICATION[idx].NAME := PUBLICATION[1].NAME;
iPUBLICATION[idx].PUBLICATION_DATE := PUBLICATION[1].<published-date>;
```

7) Finally, let's add the last section of code for PRICE, the last part of our transform. Expand the attributes under the iPRICE relationship. Be careful to place the index on the iPRICE relationship manually as [idy]. This final block of transform code takes the price information held in the PUBLICATION element in the source data model and transforms it to a separate PRICE element in the target data model.

```
iPUBLICATION[idx].iPRICE[idy].AMOUNT := PUBLICATION[1].PRICEAMOUNT;
iPUBLICATION[idx].iPRICE[idy].CURRENCY := PUBLICATION[1].PRICECODE;
```

8) Let's review the code we can see in our transform. The whole transform should look similar to the code below.

```
<ID> := <id>;
NAME := name;

iPUBLICATION[idx].CATNUM := PUBLICATION[1].CATNUM;
iPUBLICATION[idx].NAME := PUBLICATION[1].NAME;
iPUBLICATION[idx].PUBLICATION_DATE := PUBLICATION[1].<published-date>;

iPUBLICATION[idx].iPRICE[idy].AMOUNT := PUBLICATION[1].PRICEAMOUNT;
iPUBLICATION[idx].iPRICE[idy].CURRENCY := PUBLICATION[1].PRICECODE;
```

9) Click once on the File menu bar option.

10) Click once on the Save menu bar option to save your transform.

Exercise 5 - Build The Project

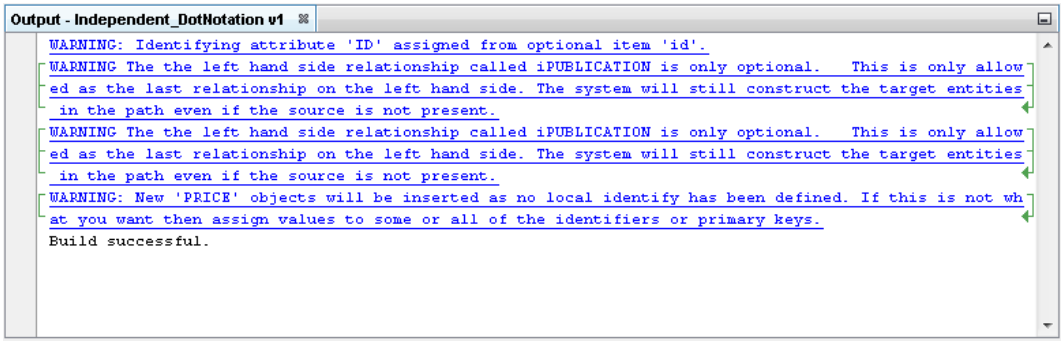
Let's build our project.

1) In the Projects pane click on the project named **Independent_DotNotation v1** so it is selected. This sets the focus for the build process.

2) Click on the Run menu option. The first two options in the list will display **Build Independent_DotNotation v1** and **Run Independent_DotNotation v1**. You must build the project before you run the project so click on the option called **Build Independent_DotNotation v1**.

You can click on this toolbar button, , to build your project as well but ensure that the project has application focus by selecting the project name in the Projects panel.

3) The **Output** pane will open and show you if there are any errors in your transform code. In this exercise you should not have any error messages and you should see a message stating **Build successful.**.

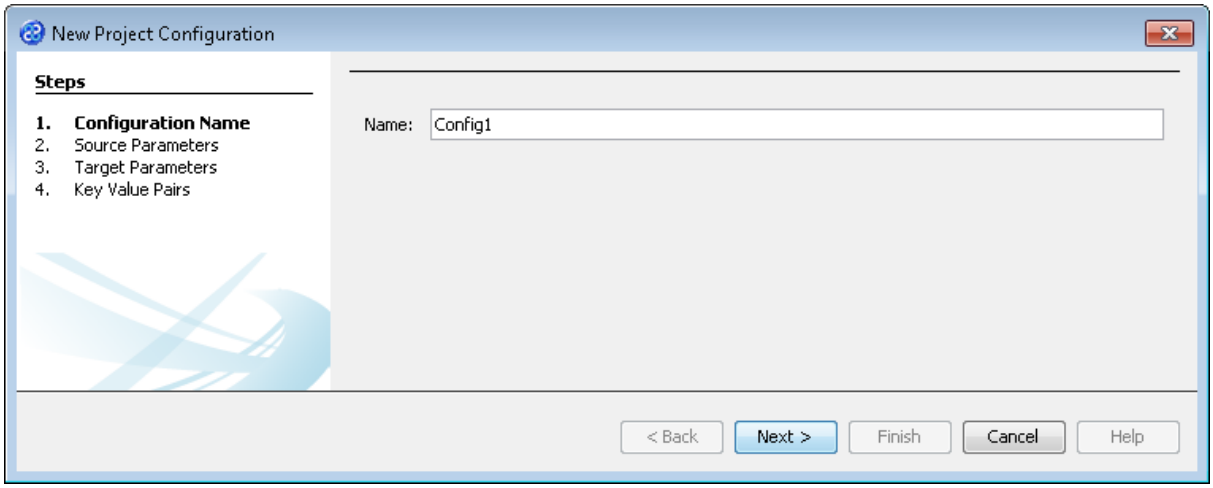


Exercise 6 - Run the Project

Now let's Run our project. This process is different to that in Tutorial 1 where we launched TM Migrator. In this exercise we will run the project within TM Designer and not in TM Migrator.

1) Click once on the Run menu bar option.

2) Click once on the Run Independent_DotNotation v1 option from the sub-menu. TM Designer will now open the New Project Configuration window where you will set the connection information for the project.



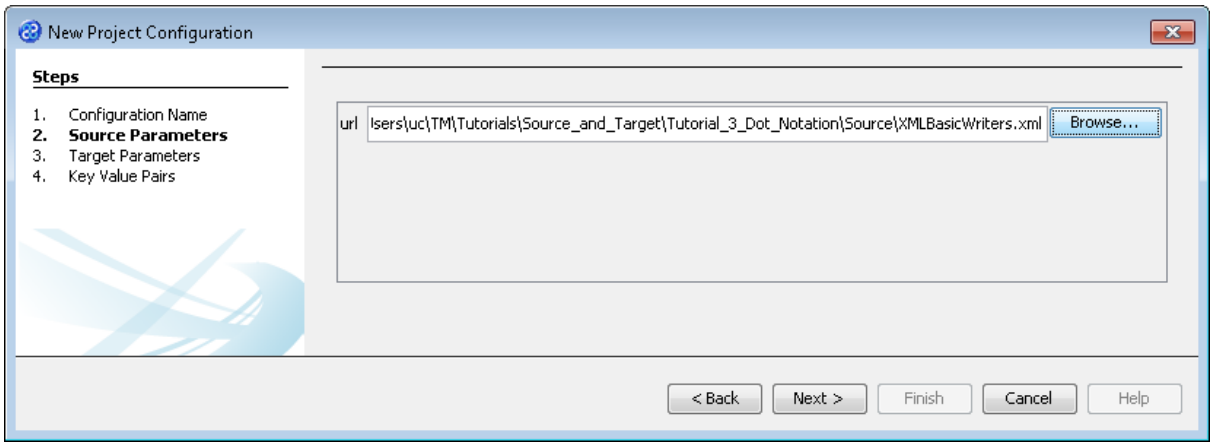
3) Lets give this configuration the name Config1. Type **Config1** into the **Name** field.

4) Click the Next > button to move to **Step 2. Source Parameters** where we will provide the details for the source data store connection.

5) Go to the **url** field of the **Source Parameters** step. In this field we will specify the location of the source data store. You can type the path and file name manually or you can use the button to select the directory and file. This will be in the following directory.

[TMHOME]\Tutorials\Source_and_Target\Tutorial_3_Dot_Notation\Source\XMLBasicWriters.xml

6) The **Source Parameters** step should look similar to the image below.



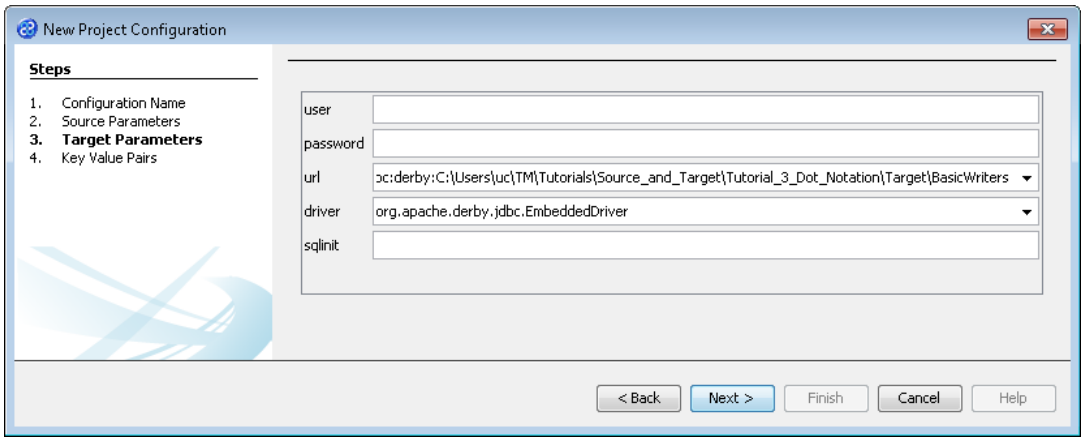
7) Click the Next > button to move to **Step 3. Target Parameters** where we will provide the details for the target data store connection.

8) Go to the **url** field. You will now need to provide the details of where to go to connect to the source data. View the list of options and select the option called **jdbc:derby:<YOUR_NAME>** from the drop down list. Now replace the <YOUR_NAME> part of the list item, including the angle brackets with the directory where the Derby database is stored. This will be in the following location.

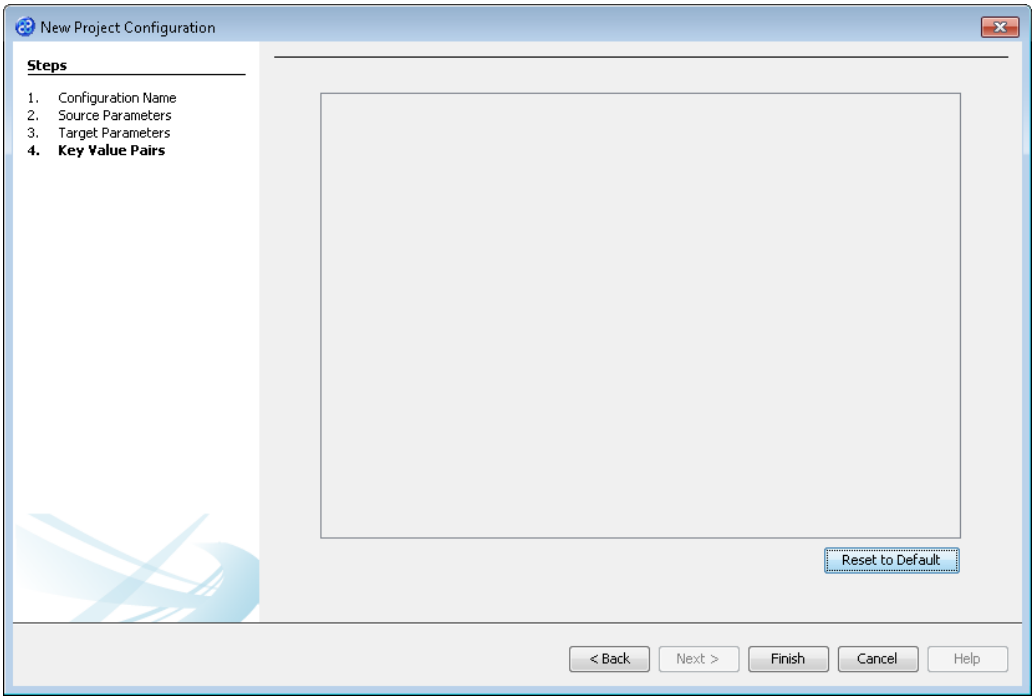
```
[TMHOME]\Tutorials\Source_and_Target\Tutorial_3_Dot_Notation\Target\BasicWr
iters
```

9) Go to the driver field and display the list of options available. Select the option called **org.apache.derby.jdbc.EmbeddedDriver**. The Adapter Configuration window will look similar to the image below. **user, password** and **sqlinit** do not require values.

10) The **Target Parameters** step should look similar to the image below.

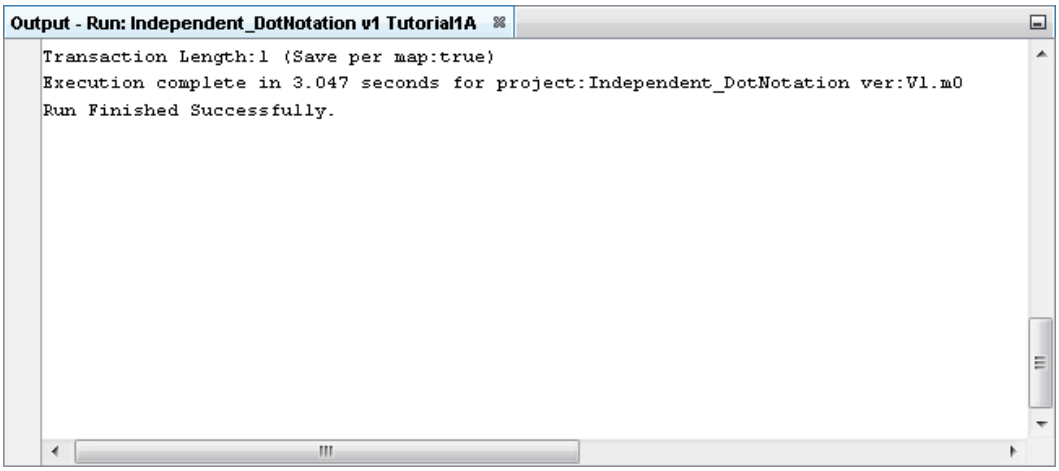


11) Click the button to move to **Step 4. Key Value Pairs**. We do not need to specify any key value pairs in this exercise.



12) Click the Finish button to save the project configuration and to run the project within TM Designer. This performs the same execution as the **Launch Migrator...** option from the project context menu.

13) TM Designer will open the Output - Run pane with the results of running the project as shown below.



14) Connect to the target data store and using the View Data... option lets see what the project has transformed for us. Below are the three tables we have updated, Writer, Publication and Price. They contain the following data.

Remember to Connect and Disconnect to the source or target data store.

Writer Table

ID	NAME	FIRST_INITIAL	COUNTRY_RESIDENCE_CODE	TOTAL_PRICE
1	Green	<null>	<null>	<null>
2	Brown	<null>	<null>	<null>
3	Blackson	<null>	<null>	<null>

Publication Table

CATNUM	NAME	PUBLICATION_DATE	WRITER_ID	TOTAL_SOLD	IS_FICTION
100111	Grass	1999-06-27	1	<null>	<null>
100222	Trees	2002-09-20	2	<null>	<null>
100444	Soot	1965-12-11	3	<null>	<null>

Price Table

PUBLICATION_ID	AMOUNT	CURRENCY
100111	20	GBP
100222	30	USD
100444	50	GBP

15) The results are as expected. You can check this by viewing the source data store.