



## **DME API. Address Validation (Win Forms)**

## Contents

Introduction .....	3
Adjusting of settings .....	3
Changing connection string to the database. ....	3
Data Source creation.....	5
DME projects creation .....	6
Application workflow .....	8

## Introduction

The 'AddressValidation.Backend.WinForms' example shows how to work with address verification, transformation, and matching. This example works with the specified data source and their column names. which are hardcoded in project's source code. It should be adjusted for the appropriate working state: editing settings, load data in the database, create DME projects with this data, add the names of these projects into settings.

MS SQL Server or SQL Server Express should be installed for working with this example. It can be installed on the same PC or remote server.

## Adjusting of settings

Changing connection string to the database.

Please correct the connection string in the settings of projects 'AddressValidation.Backend.WinForms' and 'AddressValidation.DataLayer' projects have connection string, and it should be corrected. App.config files are described on Fig.1. Fig. 1 has settings for the first project and Fig.3 for the second project. After changing the connection string, please check that files 'app.config' selected on Fig.1 contain a new value.

Also, please change the path to the file with DME settings shown in Fig.2. This parameter has the name 'ConfigFile'. Usually, this file placed via **Documents->DataMatch Enterprise->Config path**. It has the *dmeconfig.xml* name and contains Data Match Enterprise settings. The correct path allows the application uses the projects created in DME.

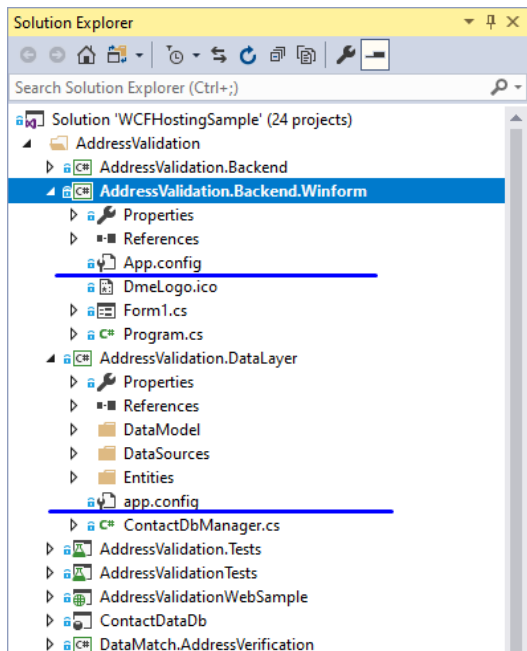


Fig.1

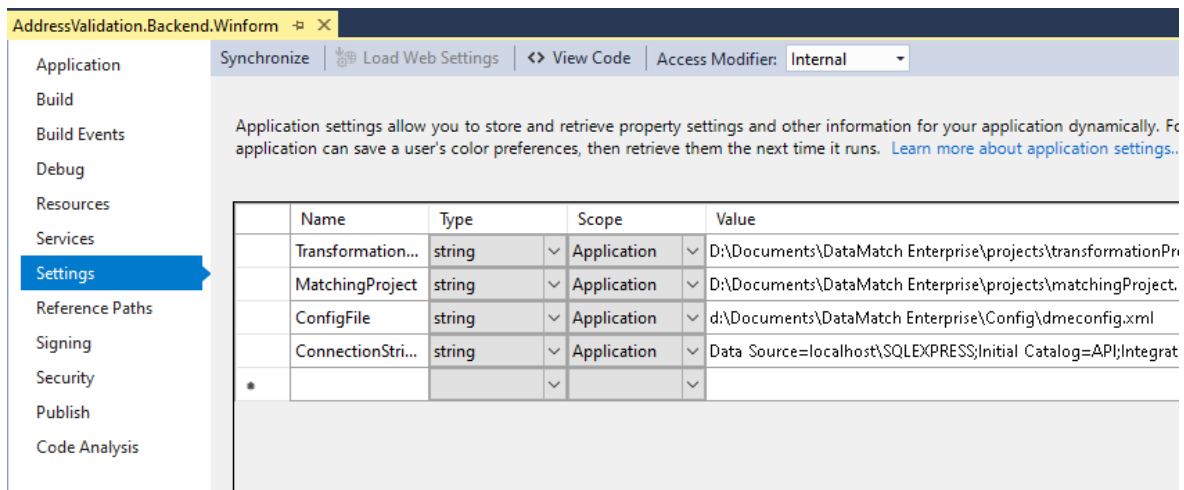


Fig.2

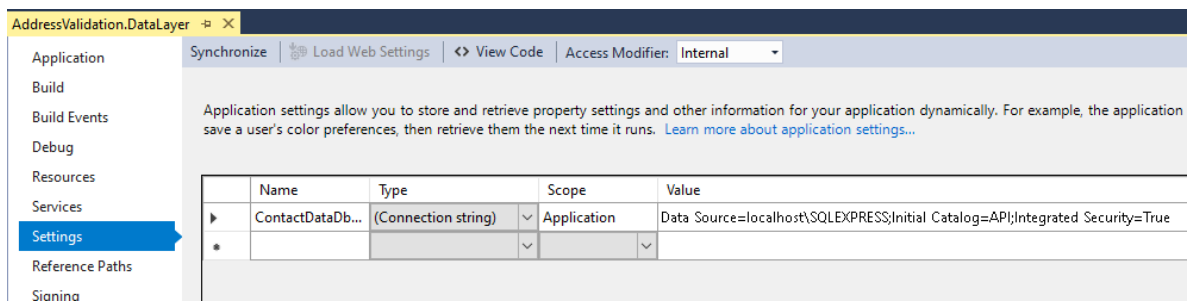


Fig.3

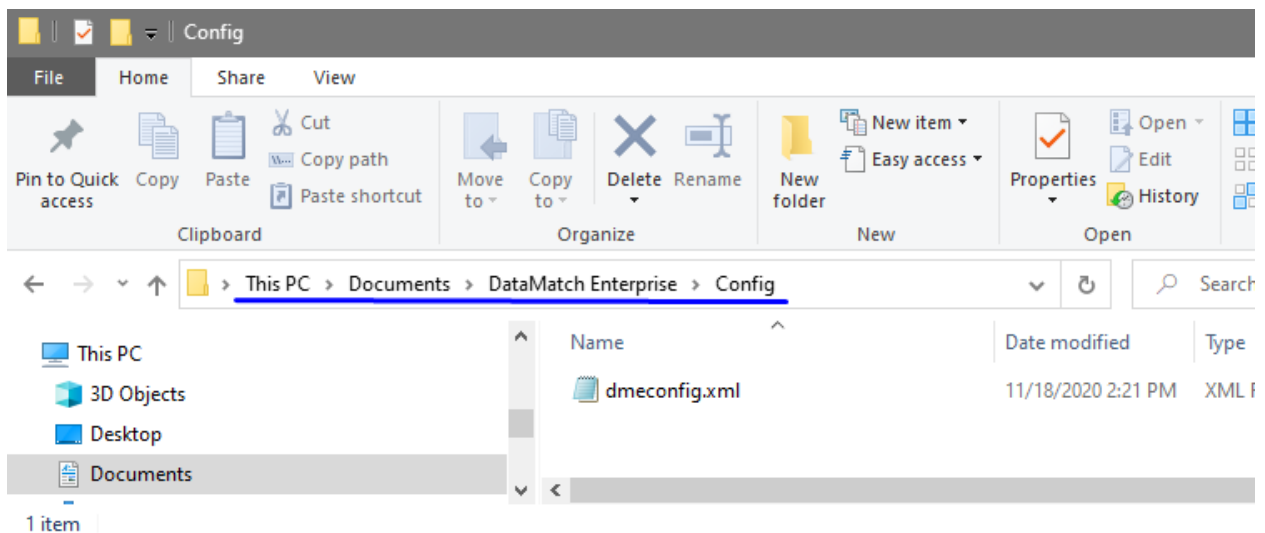


Fig.4

## Data Source creation

After correcting all connection settings, data source should be created in database. Please select the 'AddressValidation.Backend.WinForms' project in Visual Studio as a startup project. Build it and start.

Fig.5 shows application view after start. Please press the button selected on this picture. This button recreates the data source used in the application. Also, the button clears all changes that were done during the work with application. After pressing the button, the application creates *Contact table* in the database and copies data from embedded resources into it.

After table creation, you can check this table in SQL Server Management Studio. It's shown in Fig.6

The screenshot shows the 'AddressValidation' application window. At the top, there is a title bar with the application name and standard window controls. Below the title bar, there is a button labeled 'Init/Reset tables' which is highlighted with a blue rectangle. To the right of this button is a 'History' section with a large greyed-out area. Below the 'Init/Reset tables' button is a form with several input fields: 'FirstName', 'LastName', 'PrimaryAddress', 'SecondaryAddress', 'City', 'Zip', 'Phone', 'Email', 'Company Name', and 'Department'. Below these fields is an 'Insert Record' button. To the right of the form is a 'Transformed' section with a large greyed-out area. Below the 'Transformed' section is a 'Matches' section with a large greyed-out area. Below the 'Matches' section is a 'Supressed' section with a large greyed-out area. Below the 'Supressed' section is a 'Master' section with a large greyed-out area. At the bottom of the window, there is a status bar with the following text: 'Transformation project: D:\Documents\DataMatch Enterprise\projects\transformationProject.dmeproj' and 'Matching project: D:\Documents\DataMatch Enterprise\projects\matchingProject.dmeproj'.

Fig.5

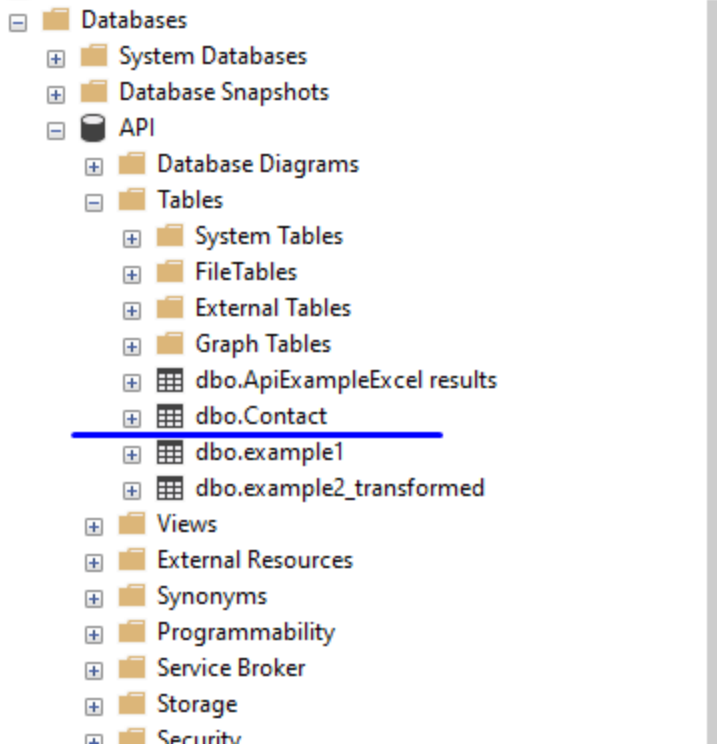


Fig.6

### DME projects creation


Please create in DataMatch Enterprise two projects described below


The first project defines standardization rules. Please import created early *Contact* data source from the database into the project and select rules on how it's shown in Fig.7. Save it and close.

Contact	
Field Name	Type
Id	
FirstName	First Name
LastName	
Address1	VPrimaryAddress
Address2	VSecondaryAddress
City	VCityName
State	VStateName
Zip	VZipCode
Country	VCountry
Phone	
Email	

Fig.7

The second project defines matching rules. Please import the table Contact. Go to the matching tab and define a definition and criteria how it's shown in Fig.8


Split Horizontal


Match

Contact	Include
Id	<input checked="" type="checkbox"/>
FirstNa...	<input checked="" type="checkbox"/>
LastNa...	<input checked="" type="checkbox"/>
Address1	<input checked="" type="checkbox"/>
Address2	<input checked="" type="checkbox"/>

Definition 1

Contact	Exact	Phonetic	Fuzzy	Numeric	Fast Level	Level	Weight
▶ FirstName	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	80	90	100
LastName	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	80	90	100

Fig.8

Please change the settings of 'AddressValidation.Backend.Winform' project. Define parameters **TransformationProject** and **MatchingProject** shown in Fig.2. Enter path in these parameters to just created DME projects.

## Application workflow

Please start the application.

The application is shown in Fig. 9. Several text fields are located on the left side of the window. Their purpose is to prepare a new entry before inserting it into the *Contact* table. Users should click the *Insert Record* button after completing these fields.

Five tables are shown on the right side of the window.

The table *History* contains all entries that users enter and try to insert in the database during the current session. This table will be cleared after the restart of the application or after clicking the Init/Reset button.

The *Transformed* table contains cleared entries according to DME **TransformationProject**. You can see results in Fig.9. User has entered Jack Taylor as first/last name and 'Pensilvanya' address with the error. After cleansing, the app found a common name for the first name and fix errors in the street name. The *Matches* table contains duplicates that were found in the *Contact* table according to **MatchingProject**. In case if duplicates are not found the cleansed record will be inserted in *Contact* table with value 0 in column *IsDuplicate* and will be showed at the end of table *Master* of the app. In case if duplicates are found *IsDuplicate* will be 1 and the record will be shown in the table *Suppressed*

Contact's table content is shown in Fig.10. This is a screenshot from SQL Server Management Studio. There were a few insertions. Insertions have Id values of more than 2131. Duplicate records are selected on this picture. The last column shows, if address was verified or not. We can see that for records with Id in (2132, 2133), the addresses were not verified, because errors were severe in the street names.



AddressValidation

Init/Reset tables

First Name  
Jack

Last Name  
Taylor

Primary Address  
1600 Pensilvania ave

Secondary Address

City

Zip  
20500

Phone

Email

Company Name  
White House

Department  
Of No 1

Insert Record

Record is inserted in Holding table  
Record is cleansed  
Record is unique  
Record is inserted in Master table

History

First Name	Last Name	Address1	Zip	City
Johny	Smithson	1600 Pensilvania...	20500	
Jack	Taylor	1600 Pensilvania...	20500	

Transformed

First Name	Common Name	Last Name	V Primary Address	Address1
Jack	John	Taylor	1600 PENNSYL...	1600 Pensilvania ave

Matches

Suppressed

First Name	Last Name	Address1	Address2	City
John	Smithson	1600 Pensilvania aw		WASHII

Master

First Name	Last Name	Address1	Address2	City
Christopher	Zehnal	220 N Main St # 2	Hudson	Hudson
John	Smith	1600 Pensilvania		WASHIN
John	Taylor	1600 PENNSYLVANIA AVE NW		WASHIN

Transformation project: D:\2\projects\transformationProject.dmeprj  
Matching project: D:\2\projects\matching\_project.dmeprj

Fig.9

	Id	First Name	Last Name	Address1	Address2	City	State	Zip	Country	Phone	Email	C...	C...	IsDuplicate	IsVerified
2128	2128	Candela...	Mirilles	4434 W THOMAS ...	Phoenix, AZ	Phoenix	AZ	85031	USA	+1 (5...	cminl...			0	1
2129	2129	Christop...	Jones	144 W BRIGHAM ...	Saint George, UT	Saint George	UT	84790	USA	+1 (1...	cjone...			0	1
2130	2130	Christop...	Liti	207 N 3RD ST	Apollo, PA	Apollo	PA	15613	USA	+1 (8...	cliti@...			0	1
2131	2131	Christop...	Zehnal	220 N Main St # 2	Hudson	Hudson	NULL	ZIP44	USA	+1 (6...	czehn...			0	0
2132	2132	John	Smith	1600 Pensilvania		WASHING...	DC	20500				W.	O.	0	0
2133	2133	John	Smithson	1600 Pensilvania aw		WASHING...	DC	20500				W.	O.	1	0
2134	2134	John	Smithson	1600 PENNSYLV...		WASHING...	DC	20500				W.	O.	1	1
2135	2135	John	Taylor	1600 PENNSYLV...		WASHING...	DC	20500				W.	O.	0	1
2136	2136	John	Taylor	1600 PENNSYLV...		WASHING...	DC	20500				W.	O.	1	1

Fig.10