

Simple Demonstration Application – VB .NET Version 02

Quickstart Guide

This Guide is designed to help you build your own VB.Net database applications using Westfaro Corporation's VB.Net Scripts as the starting point. If you follow the steps outlined in this guide you will be able to create the equivalent application for your databases very quickly.

Version 2 of the VB.Net demonstration scripts uses an MDI framework for the application.

A preview

The screen shot (Figure 1) shows an MDI application for maintaining the NWind database. There are 3 open forms. The one with focus shows the Categories table in field view. The form to the lower left shows the second tab, which lists the rows. A further form for Customers is in the background.

The table maintenance form supports adding, deleting and editing rows. The grid on second tab allows you to select the record to be displayed on the right. The toolbar applies to the form with focus. It allows the user to maintain records and scroll through the database table.

The Maintenance menu is used to select which table is to be maintained while the Action menu corresponds to the toolbar.

Creating the application

1. Download the SimpleVBNet02.zip file and extract the scripts (.ksc extension).
2. Extract the Northwind database Nwind.mdb. Note that it should be placed in a folder called "bin".
3. Modify the Kickstart data type mappings so they look like those shown in Figure 2. You do this by choosing **Datatype Setup** from the **Tools** menu and double clicking in each cell that needs to be edited. The reason for these changes is to support the processing of picture columns as byte arrays.
4. Connect to that Northwind database using Kickstart.
5. Select the Script IncludeAll.ksc
6. Select the same "bin" destination folder as you used for the Northwind database Nwind.mdb
7. Click the Generate button
8. Start VB.NET and open the solution "SimpleVBNet02"

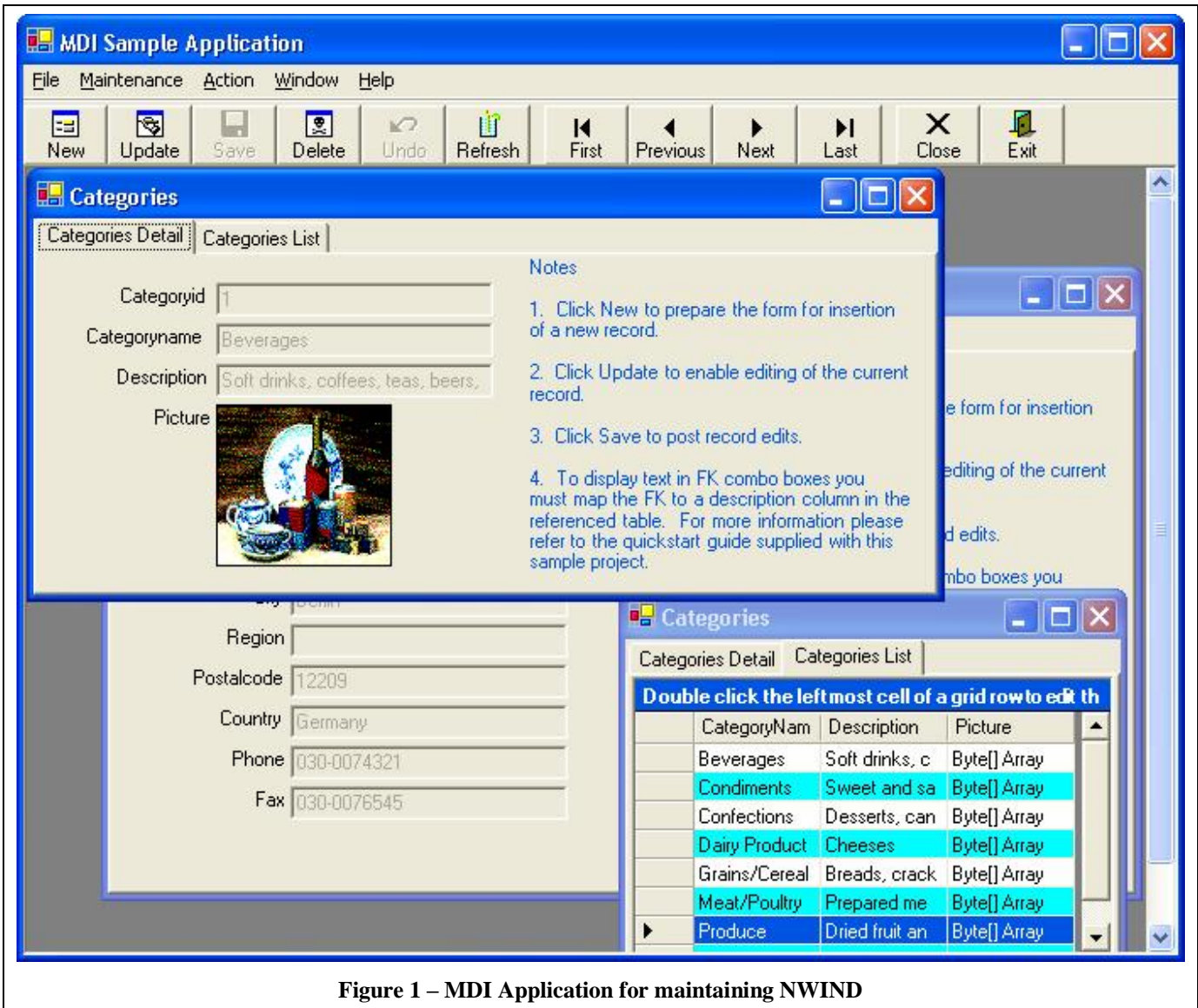


Figure 1 – MDI Application for maintaining NWIND

Next steps

The raw generated code could obviously do with some cleaning up. The biggest problem is the way it displays foreign keys (lookups). Since the Database doesn't know which columns in the **Categories** and **Suppliers** tables should be displayed to the user, Kickstart uses the referenced column by default. However, you can use Kickstart to choose which columns should be displayed in a foreign key relationship. For **Categories**, you might choose **Categoryname** or **Description** or both.

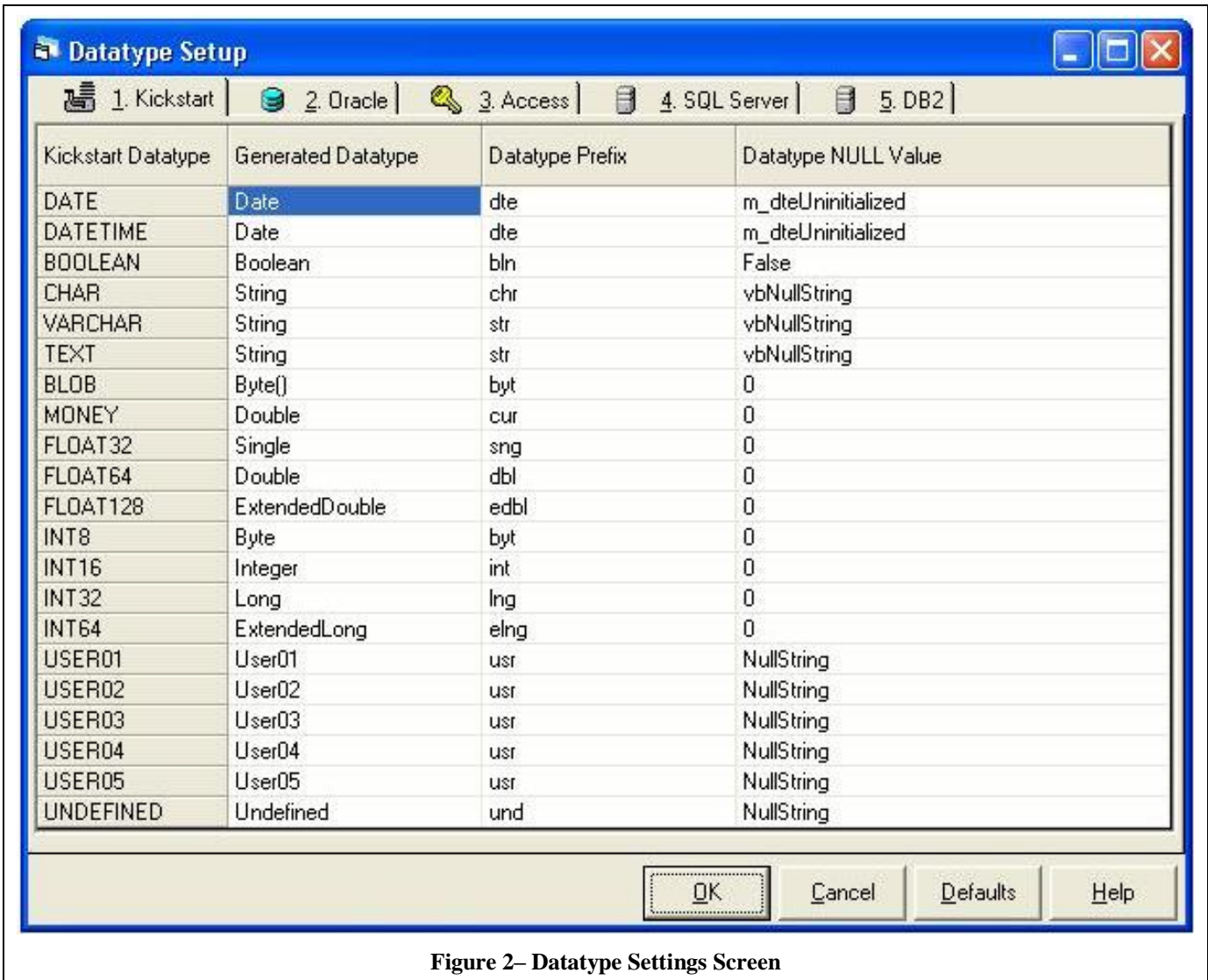


Figure 2– Datatype Settings Screen

Customizing the generated code

Some application generators attempt to generate working applications but they are often difficult to customize. Others generate source code only a computer could like.

The thinking behind Kickstart is different. Lots of database and application code is repetitive. Programmers develop a template that works for one table and then cut and paste that code for the next table. Kickstart automates that repetitive work. You write your best code for one table and then convert that code into a Kickstart script. Once you've done that you have effectively rewritten the code for every table. When you have a large database with dozens or hundreds of tables, being able to replicate your best database code at the touch of a button is awesome.