

Table of Contents

Foreword	0
Part I Introduction	2
Part II Installation	2
1 Trial Version.....	2
2 Full Version.....	3
Part III How to Distribute It	4
Part IV Reference Guide	4
1 Properties.....	4
BackColor Property	4
BarColor Property	4
CountryCode Property	4
Data Property	5
HandleTilde Property	5
Mode Property	5
Orientation Property	6
Ratio Property	6
ServiceClass Property	6
ZipCode Property	7
2 Methods.....	7
GetActualSize Method	7
Render Method	8
SaveAsImage Method	8
SaveAsMemory Method	9
SetSize Method	9
SetStructuredAppend Method	10
3 Enumerations.....	10
Mode Enumeration	10
Orientation Enumeration	11
Part V How to Use It in Reporting Services	11
1 Create a Report Project.....	11
2 Add MaxiCode Barcodes.....	14
Part VI Special Format Data	18
Part VII License	19
Index	0

1 Introduction

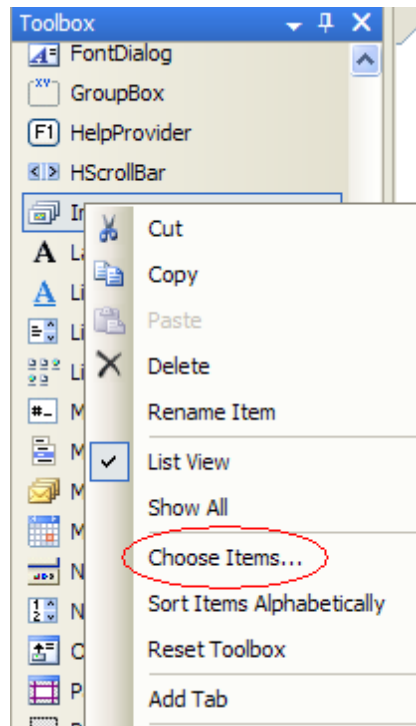
MW6 MaxiCode .NET control is a flexible and reliable .NET component and can create professional 2D MaxiCode images for your .NET application, you can save the MaxiCode as different image format files. It is easy to print the MaxiCode barcode using the PrintDocument Control provided by the .NET Framework.

MaxiCode is a fixed-sized 2D symbology created by the United Parcel Service, it can store about 93 characters of information and is primarily used for freight sortation and tracking.

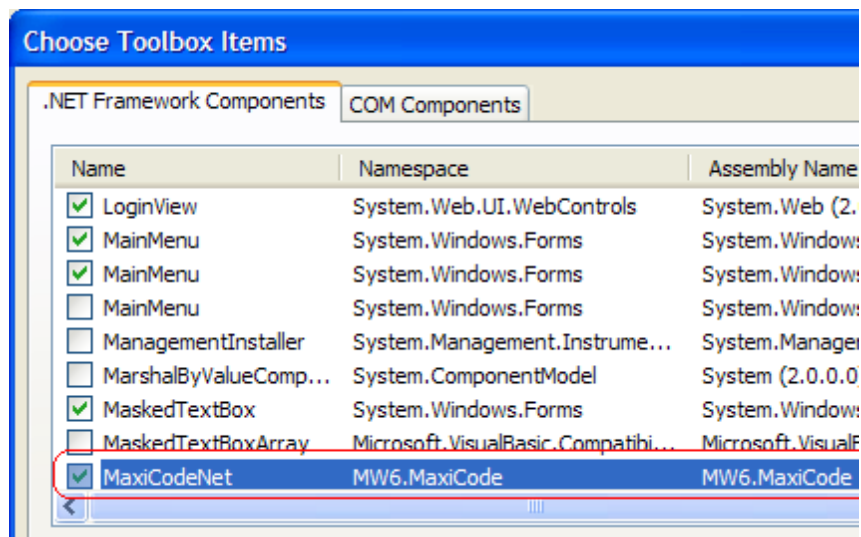
2 Installation

2.1 Trial Version

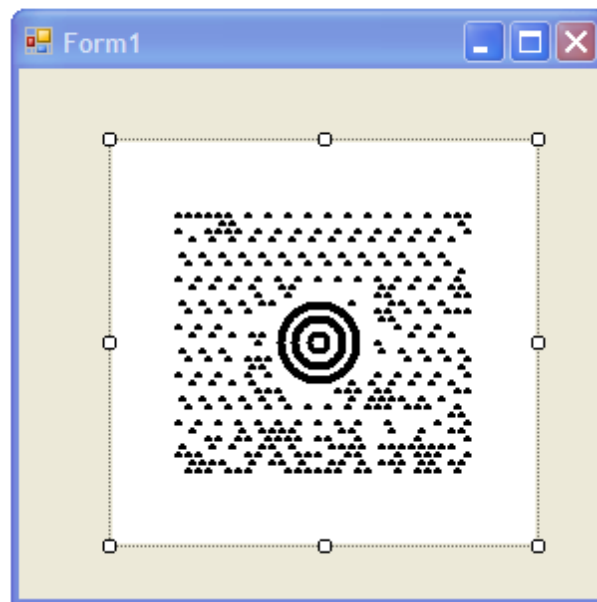
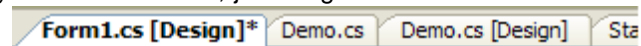
1. The trial version MaxiCode .NET control displays the "MW6 Demo" on the top of MaxiCode barcode.
2. After the installation for the trial version is finished, copy MW6.MaxiCode.dll in VB.NET or C# demo project sub folder of the destination folder (e.g. "C:\Program Files\MW6 .NET Components \MaxiCode") to your project folder.
3. Right click anywhere on the Toolbox to select "**Choose Items...**".



4. In the "**Choose Toolbox Items**" dialog, select ".NET Framework Components", click on the "**Browse**" and select MW6.MaxiCode.dll.
-



5. To use the control in your Windows form, just drag it from the Toolbox and drop it onto your form.



2.2 Full Version

1. Uninstall the trial version MaxiCode .NET control if applicable.
2. Complete the installation for the full version MaxiCode .NET control, copy MW6.MaxiCode.dll in VB.NET or C# demo project sub folder of the destination folder (e.g. "C:\Program Files\MW6 .NET Components\MaxiCode") to your project folder to replace the trial version MaxiCode .NET control .dll file.

3 How to Distribute It

If you want to redistribute the MaxiCode .NET control as part of your application, simply put MW6.MaxiCode.dll into the application local folder on the target machine.

4 Reference Guide

4.1 Properties

4.1.1 BackColor Property

Gets or sets the background color of the MaxiCode barcode.

```
[Visual Basic .NET]
```

```
Public Property BackColor As Color
```

```
[C#]
```

```
public Color BackColor {get; set;}
```

Remarks

The default value is white color.

4.1.2 BarColor Property

Gets or sets the color of the MaxiCode barcode.

```
[Visual Basic .NET]
```

```
Public Property BarColor As Color
```

```
[C#]
```

```
public Color BarColor {get; set;}
```

Remarks

The default value is black color.

4.1.3 CountryCode Property

Gets or sets the country code.

```
[Visual Basic .NET]
```

```
Public Property CountryCode As String
```

```
[C#]
```

```
public string CountryCode {get; set;}
```

Remarks

The default value is blank, if the "Data" property is properly formatted and begins with the 7 characters "[]><RS>01<GS>", this property will be automatically overridden.

See Also

Special Format Data

4.1.4 Data Property

Gets or sets the message to encode with MaxiCode .NET control.

```
[Visual Basic .NET]
```

```
Public Property Data As String
```

```
[C#]
```

```
public string Data {get; set;}
```

Remarks

The default value is "12".

4.1.5 HandleTilde Property

Gets or sets a boolean flag indicating whether to process the tilde character "~" or not.

```
[Visual Basic .NET]
```

```
Public Property HandleTilde As Boolean
```

```
[C#]
```

```
public bool HandleTilde {get; set;}
```

Remarks

If this property is set to TRUE, non-printable characters can be passed to MaxiCode .NET control by using the tilde character, "~dNNN" represents the ASCII character encoded by the 3 digits NNN, for example, "~d010" represents the character LF (line feed).

4.1.6 Mode Property

Gets or sets the encoding mode of the MaxiCode barcode.

```
[Visual Basic .NET]
```

```
Public Property Mode As enumMode
```

```
[C#]  
public enumMode Mode {get; set;}
```

4.1.7 Orientation Property

Gets or sets the orientation of the MaxiCode barcode.

```
[Visual Basic .NET]  
Public Property Orientation As enumOrientation
```

```
[C#]  
public enumOrientation Orientation {get; set;}
```

4.1.8 Ratio Property

Gets or sets the ratio value which is used to enlarge or shrink the MaxiCode barcode.

```
[Visual Basic .NET]  
Public Property Ratio As float
```

```
[C#]  
public float Ratio {get; set;}
```

Remarks

The default value is 1, a valid value must be between 0.7 and 3.

4.1.9 ServiceClass Property

Gets or sets the 3-digit service class.

```
[Visual Basic .NET]  
Public Property ServiceClass As String
```

```
[C#]  
public string ServiceClass {get; set;}
```

Remarks

The default value is blank, if the "Data" property is properly formatted and begins with the 7 characters "[]><RS>01<GS>", this property will be automatically overridden.

See Also

Special Format Data

4.1.10 ZipCode Property

Gets or sets the zip code.

[Visual Basic .NET]

```
Public Property ZipCode As String
```

[C#]

```
public string ZipCode {get; set;}
```

Remarks

The default value is blank, if the "Data" property is properly formatted and begins with the 7 characters "[]><RS>01<GS>", this property will be automatically overridden.

See Also

Special Format Data

4.2 Methods

4.2.1 GetActualSize Method

Gets the actual size of the MaxiCode barcode which is rendered onto either computer screen or other devices such as printers.

[Visual Basic .NET]

```
Public Sub GetActualSize(ByVal ScreensTarget As Boolean, _  
                        ByVal TargetG As Graphics, _  
                        ByRef ActualWidth As Integer, _  
                        ByRef ActualHeight As Integer)
```

[C#]

```
public void GetActualSize(bool ScreensTarget,  
                          Graphics TargetG,  
                          ref int ActualWidth,  
                          ref int ActualHeight);
```

Parameters

ScreensTarget

Indicates whether the MaxiCode barcode is rendered onto computer screen or not.

TargetG

Graphics object to be used for rendering, if the parameter ScreensTarget is set to TRUE, set this parameter to NULL.

ActualWidth

A pointer to the variable that receives the width of the MaxiCode barcode (in pixels).

ActualHeight

A pointer to the variable that receives the height of the MaxiCode barcode (in pixels).

4.2.2 Render Method

Renders the MaxiCode barcode onto the device such as a computer monitor or a printer.

[Visual Basic .NET]

```
Public Sub Render(ByVal renderG As Graphics, ByVal p As Point)
```

[C#]

```
public void Render(Graphics renderG, Point p);
```

Parameters

renderG

Graphics object to be used for rendering.

p

Stores the coordinates (in pixels) of the top-left corner of the MaxiCode barcode.

4.2.3 SaveAsImage Method

Exports the MaxiCode barcode image to a file.

[Visual Basic .NET]

```
Public Sub SaveAsImage(ByVal FileName As String, ByVal ImgFormat As ImageFormat)
```

[C#]

```
public void SaveAsImage(string FileName, ImageFormat ImgFormat);
```

Parameters

FileName

A string that contains the name of the file to which to save MaxiCode barcode image.

ImgFormat

Specifies the image format.

Remarks

Before you call this method, use `GetActualSize()` method to obtain the actual size of MaxiCode barcode and use `SetSize()` method to set the image size by adding the surrounding white space around the MaxiCode barcode.

See Also

[GetActualSize\(\) Method](#) | [SetSize\(\) Method](#)

4.2.4 SaveAsMemory Method

Exports the MaxiCode barcode image byte stream to the memory.

[Visual Basic .NET]

```
Public Sub SaveAsMemory(ByVal MS As MemoryStream, ByVal ImgFormat As ImageFormat)
```

[C#]

```
public void SaveAsMemory(MemoryStream MS, ImageFormat ImgFormat);
```

Parameters

MS

Specifies the memory stream that holds the byte stream of the MaxiCode barcode image.

ImgFormat

Specifies the image format.

Remarks

Before you call this method, use `GetActualSize()` method to obtain the actual size of the MaxiCode barcode and use `SetSize()` method to set the image size by adding the surrounding white space around the MaxiCode barcode.

See Also

[GetActualSize\(\) Method](#) | [SetSize\(\) Method](#)

4.2.5 SetSize Method

Sets the size of the image which contains the MaxiCode barcode.

[Visual Basic .NET]

```
Public Sub SetSize(ByVal Width As Integer, ByVal Height As Integer)
```

[C#]

```
public void SetSize(int Width, int Height);
```

Parameters

Width

The width, in pixels, of the image.

Height

The height, in pixels, of the image.

Remarks

First call `GetActualSize()` method to obtain the actual size of the MaxiCode barcode, then use this method to set image size by adding surrounding white space around the MaxiCode barcode.

See Also

`GetActualSize()` Method

4.2.6 SetStructuredAppend Method

Specifies which symbol this is in a sequence and the total number of symbols in the sequence.

[Visual Basic .NET]

```
Public Sub SetStructuredAppend(ByVal AllowSA As Boolean, _
                             ByVal SymbolID As Integer, _
                             ByVal SymbolCount As Integer)
```

[C#]

```
public void SetStructuredAppend(bool AllowSA,
                               int SymbolID,
                               int SymbolCount);
```

Parameters

AllowSA

Indicates whether the structured append is allowed in the current MaxiCode barcode, if this is FALSE, the parameters *SymbolID* and *SymbolCount* are irrelevant.

SymbolID

Specifies which symbol this is in a sequence, the parameter must be between 1 and *SymbolCount*.

SymbolCount

Specifies the total number of symbols in the sequence, the maximum value is 8, which means that up to 8 symbols can be linked together using the structured append protocol.

Remarks

Don't call this method if you don't need the structured append feature.

4.3 Enumerations**4.3.1 Mode Enumeration**

An enumeration type for all possible encoding mode values.

Members

Name	Comment
------	---------

mdMode2	Mode 2
mdMode3	Mode 3
mdMode4	Mode 4
mdMode5	Mode 5

4.3.2 Orientation Enumeration

An enumeration type for all possible orientation values.

Members

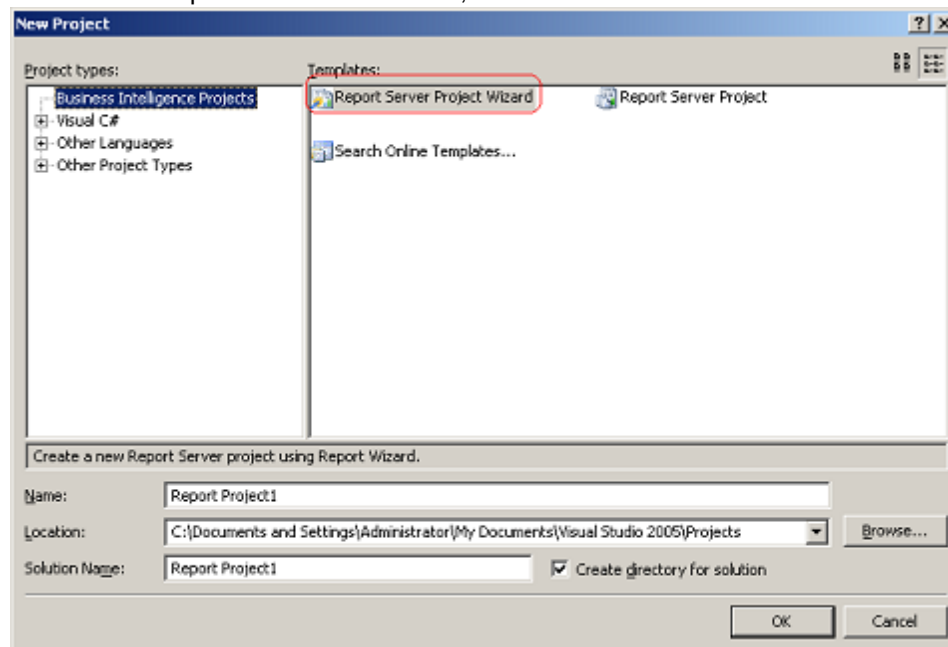
Name	Comment
or0	0 Degree
or90	90 Degrees
or180	180 Degrees
or270	270 Degrees

5 How to Use It in Reporting Services

5.1 Create a Report Project

Follow the instructions to create a report project in the Reporting Services environment:

1. Select **File | New | Project** from the Visual Studio .NET IDE menu, in the **New Project** dialog, highlight the "**Business Intelligence Projects**", then choose the "**Report Server Project Wizard**", enter the name of the report in the "**Name**" box, click the "**OK**" button.

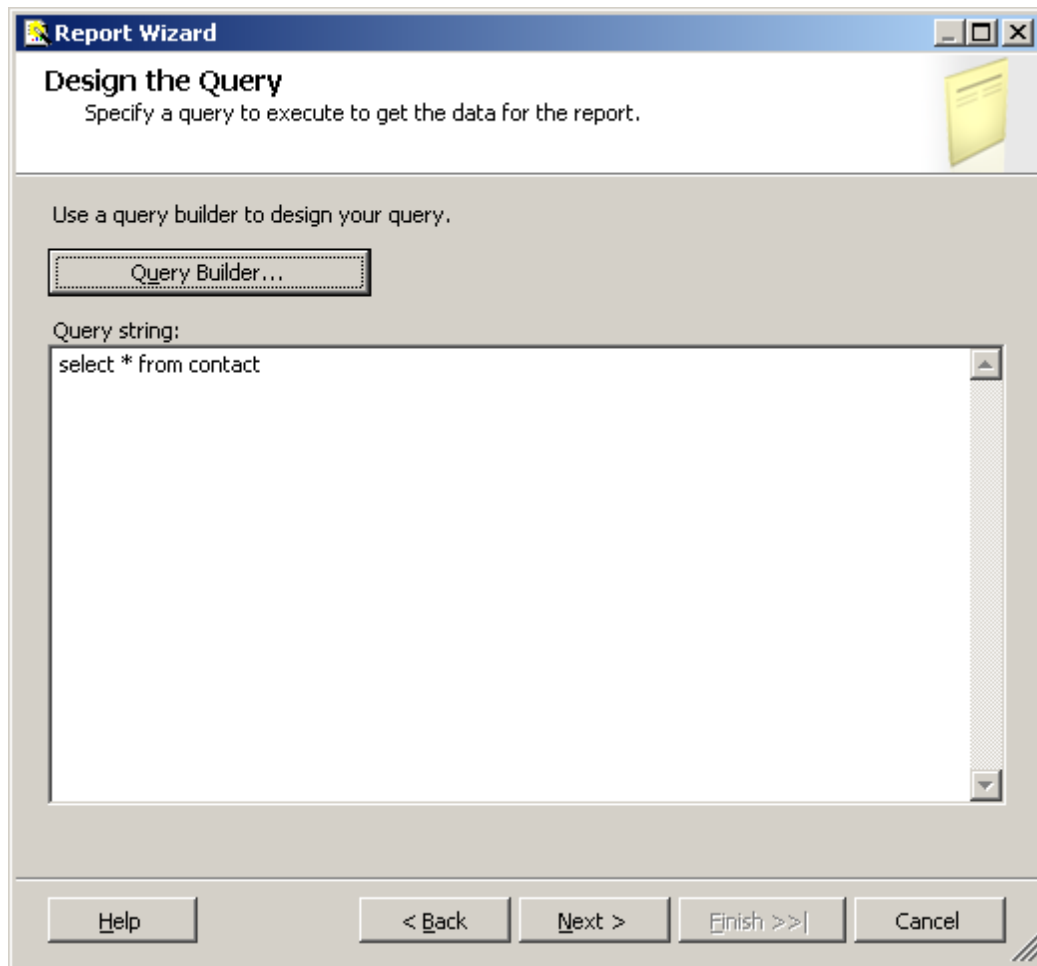


2. In the "Select the Data Source" dialog, click the "Edit" button to open the "Connection Properties" dialog, enter the SQL server instance name in the "Server name" box, choose a database from the drop-down list, click the "OK" button.

The screenshot shows the "Connection Properties" dialog box. The "Server name" dropdown is set to "PC-V40G4L4T6\SQLEXPRESS_NEW" and is highlighted with a red box. The "Connect to a database" section has the "Select or enter a database name:" radio button selected, and the dropdown menu is set to "Customers", also highlighted with a red box. The "Data source" is set to "Microsoft SQL Server (SqlClient)". The "Log on to the server" section has "Use Windows Authentication" selected. The "Advanced..." button is visible at the bottom right of the dialog.

The screenshot shows the 'Report Wizard' dialog box with the title 'Select the Data Source'. The instructions read: 'Select a data source from which to obtain data for this report or create a new data source.' There are two radio buttons: 'Shared data source' (unselected) and 'New data source' (selected). Under 'New data source', there is a 'Name:' field containing 'DataSource1', a 'Type:' dropdown menu set to 'Microsoft SQL Server', and a 'Connection string:' text area containing 'Data Source=PC-V4OG4L4T6\SQLEXPRESS_NEW;Initial Catalog=Customers'. To the right of the connection string are 'Edit...' and 'Credentials...' buttons. At the bottom, there is a 'Make this a shared data source' checkbox (unchecked) and a row of navigation buttons: 'Help', '< Back', 'Next >', 'Finish >>', and 'Cancel'.

3. Click the "**Next**" button to open the "**Design the Query**" dialog, enter the query string to extract the data, click the "**Next**" button

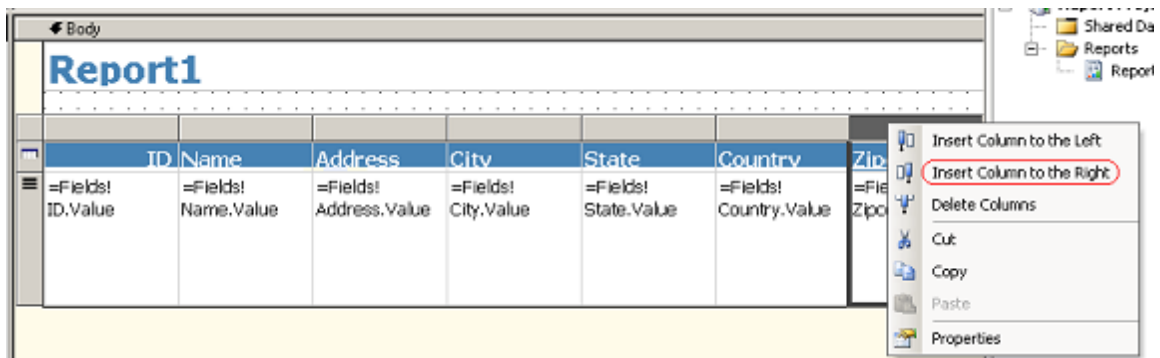


4. In the following dialogs, choose the appropriate options for the type of the report, the way of how to group the data in the table and the table style, then click the "**Finish**" button.

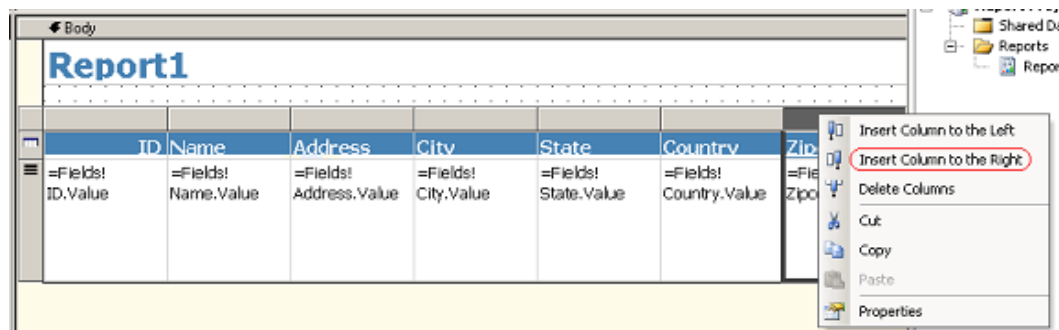
5.2 Add MaxiCode Barcodes


Follow the instructions to add the MaxiCode barcodes to the report.

1. In order to use the MaxiCode .NET control in the Reporting Services, please copy MW6.MaxiCode.dll to "*C:\Program Files\Microsoft Visual Studio X\Common7\IDE\PrivateAssemblies*" for 32 bit OS or "*C:\Program Files (x86)\Microsoft Visual Studio X\Common7\IDE\PrivateAssemblies*" for 64 bit OS, the X value is associated with Visual Studio .NET version, it might be 8 for .NET 2005, 9.0 for .NET 2008, 10.0 for .NET 2010, 11.0 for .NET 2012.
2. Right click the last column in the table, Select the "**Insert Column to the Right**".



3. Change the column title to the "**MaxiCode Barcode**".



4. On the "**Report**" menu, click the "**Report Properties**", click the "**References**" tab, click the two-dot button  to open the "**Add Reference**" dialog.

- A. Click the "**Browse**" tab, navigate to the location of the assembly MW6.MaxiCode.dll, select the file and click the "**Add**" button.
- B. Click the "**.NET**" tab, highlight the assemblies "**System.Drawing**" and "**System.Windows.Form**", click the "**Add**" button.
- C. Enter "MW6.MaxiCode.MaxiCodeNet" in the "**Class name**" box, enter "objMaxiCode" in the "**Instance name**" box to create an assembly object to use in the code to retrieve the MaxiCode barcode image byte stream.

5. On the same "**Report Properties**" dialog, click the "**Code**" tab, copy and paste the following code into this tab, this function is used to retrieve the MaxiCode barcode image byte stream, modify the code a bit to meet your application requirements.

```
Public Function GetImgStream(ByVal DataStr As String) As Byte()

    Dim ActualWidth As Integer, ActualHeight As Integer
    Dim ExtraWidth As Integer, ExtraHeight As Integer

    ' Message
    objMaxiCode.Data = DataStr

    ' Ratio
    objMaxiCode.Ratio = 1.0

    ' Mode
    objMaxiCode.Mode = 0

    ' Orientation
    objMaxiCode.Orientation = 0

    ' Handle Tilde Character?
    objMaxiCode.HandleTilde = True

    ExtraWidth = 60
    ExtraHeight = 80

    objMaxiCode.GetActualSize(True, Nothing, ActualWidth, ActualHeight)

    objMaxiCode.SetSize(ActualWidth + ExtraWidth, ActualHeight + ExtraHeight)

    Dim MS As System.IO.MemoryStream = New System.IO.MemoryStream
    Dim ImgStream As Byte()

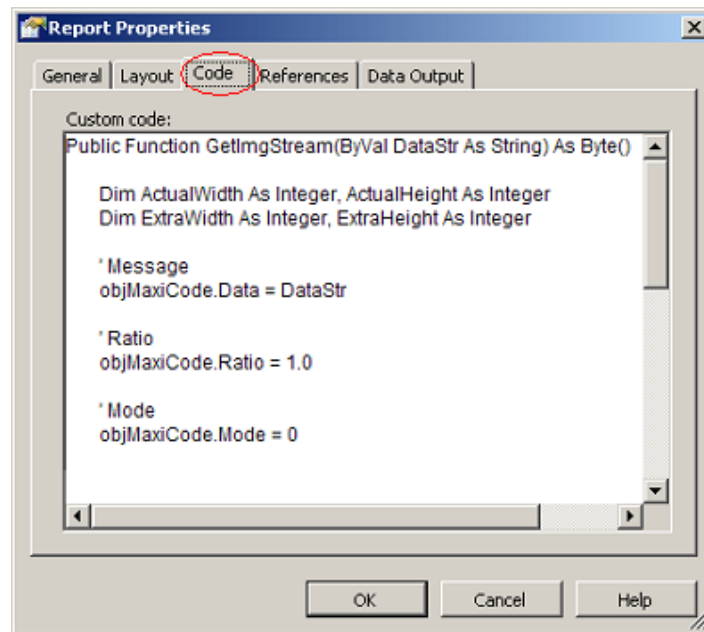
    objMaxiCode.SaveAsMemory(MS, System.Drawing.Imaging.ImageFormat.Jpeg)

    ImgStream = MS.ToArray

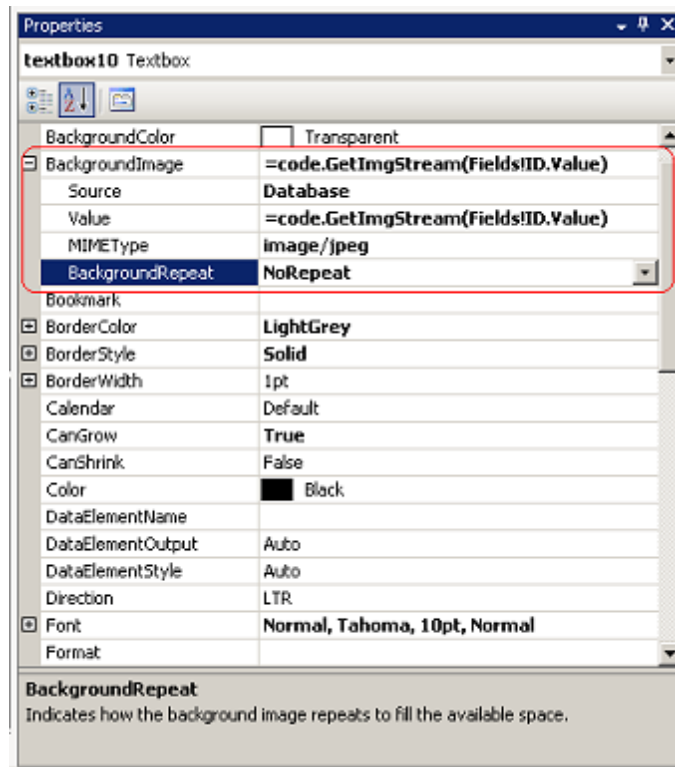
    MS.Close()

    Return ImgStream

End Function
```

6. Change the "**BackgroundImage**" property of the text box in the "**MaxiCode Barcode**" column to display the MaxiCode barcode image encoding ID value.
- A. **Source:** Select the Database from the drop-down list, since the data will be pulled from a database field.
 - B. **Value:** Enter the string "=code.GetImgStream(Fields!ID.Value)", it will ask the report to use the GetImgStream() function to retrieve the MaxiCode barcode image byte stream for the ID field value of the database.
 - C. **MIMEType:** Select the image/jpeg from the drop-down list, since the MaxiCode barcode image is in jpeg format.
 - D. **BackgroundRepeat:** Select the NoRepeat from the drop-down list, so only one image will be placed in the text box.



7. Click the "**Preview**" tab to check the MaxiCode barcode result.
8. For the reporting services deployment, check out Microsoft Knowledge Base article 842419 with the title "**How to grant permissions to a custom assembly that is referenced in a report in Reporting Services**", it is required to update the security settings of the .NET Framework to allow the MW6.MaxiCode.dll assembly to run properly.

6 Special Format Data

If the "Data" property is properly formatted and begins with the 7 characters "[]><RS>01<GS>", ZipCode, CountryCode and ServiceClass properties will be automatically overridden.

For example, let's set the "Data" property to the following value:

```
[ ]><RS>01<GS>9615238<GS>840<GS>001<GS>AIM, Inc<GS>634 Alpha
Drive<GS>Pittsburgh<GS>PA<RS><EOT>
```

In this format, the identifier "[]><RS>01<GS>" is followed by a date (YY), in this example, it is "96".

The above data is encoded in a particular manner as follows:

- 1) The first 9 data characters []><RS>01<GS>YY are extracted
- 2) The next 3 data elements separated by <GS>, representing respectively the zip code, country code and service class, are extracted and encoded in the primary message. In this example, they are

15238, 840 and 001, so Zipcode, Country and ServiceClass properties are automatically overridden.

- 3) The remaining string preceded with [)]><RS>01<GS>YY is encoded in the secondary message. In this example, it is

```
[)]><RS>01<GS>YYAIM, Inc<GS>634 Alpha Drive<GS>Pittsburgh<GS>PA<RS><EOT>
```

Remarks:

<RS>, <GS> and <EOT> indicate 3 characters with ASCII values 30, 29 and 4 respectively.

7 License

License agreement

This License Agreement ("LA") is the legal agreement between you and MW6 Technologies, Inc. ("MW6") for the font, and any electronic documentation ("Package"). By using, copying or installing the Package, you agree to be bound by the terms of this LA. If you don't agree to the terms in this LA, immediately remove unused Package.

1. License

* The Single User License allows the use of the software on **ONE** computer by **ONE** person in your organization.

* The Site License allows the use of the software at exactly 1 physical site by up to 10,000 users in your organization.

* The Single Developer License allows 1 developer in your organization the royalty-free distribution (up to 10,000 users) of the software to the third parties, **each individual developer requires a separate Single Developer License as long as he or she needs access to MW6's product(s) and document(s).**

* The 2 Developer License allows 2 developers in your organization the royalty-free distribution (up to 10,000 users) of the software to the third parties.

* The 3 Developer License allows 3 developers in your organization the royalty-free distribution (up to 10,000 users) of the software to the third parties.

* The 4 Developer License allows 4 developers in your organization the royalty-free distribution (up to 10,000 users) of the software to the third parties.

* The 5 Developer License allows 5 developers in your organization the royalty-free distribution (up to 10,000 users) of the software to the third parties.

* The Unlimited Developer License allows unlimited number of developers in your organization the royalty-free distribution (unlimited number of users) of the software to the third parties.

2. User Disclaimer

The software is provided "as is" without warrant of any kind, either expressed or implied, including, but not limited to, the implied warranties of merchantability, fitness for a particular purpose, or noninfringement. MW6 assumes no liability for damages, direct or consequential, which may result from the use of the software. Further, MW6 assumes no liability for losses caused by misuse or abuse of the software. This responsibility rests solely with the end user.

3. Copyright

The software and any electronic documentation are the proprietary products of MW6 and are protected by copyright and other intellectual property laws.
