TBarCode OCX
Barcode ActiveX® Control

Version 8.0

Developer Manual

3 December 2007

TEC-IT Datenverarbeitung GmbH
Wagnerstrasse 6
A-4400 Steyr, Austria

t ++43 (0)7252 72720
f ++43 (0)7252 72720 77
office@tec-it.com
www.tec-it.com

1 Content

2 Disclaimer

The actual version of this product (document) is available as is. TEC-IT declines all warranties which goes beyond applicable rights. The licensee (or reader) bears all risks that might take place during the use of the system (the documentation). TEC-IT and its contractual partner cannot be penalized for direct and indirect damages or losses (this includes non-restrictive, damages through loss of revenues, constriction in the exercise of business, loss of business information or any kind of commercial loss), which is caused by use or inability to use the product (documentation), although the possibility of such damage was pointed out by TEC-IT.

We reserve all rights to this document and the information contained therein. Reproduction, use or disclosure to third parties without express authority is strictly forbidden.

Für dieses Dokument und den darin dargestellten Gegenstand behalten wir uns alle Rechte vor. Vervielfältigung, Bekanntgabe an Dritte oder Verwendung außerhalb des vereinbarten Zweckes sind nicht gestattet.

3   Introduction

3.1   What is TBarCode?

TBarCode is a set of professional tools for the generation of barcodes. More than 70 different symbologies (linear as well as 2D or stacked barcode variants) can be printed or exported as graphics file. All industry formats, in the highest possible resolution and quality, are generated.

TBarCode is available in several versions for different operating systems, applications and programming environments. The following versions are included in this setup:

TBarCode OCX	A Microsoft® ActiveX® compliant barcode control. It can be used with Microsoft® Office applications as well as by software developers.
TBarCode .NET	A .NET barcode library for software developers. It includes barcode controls for Windows Forms and ASP.NET 2.0.
TBarCode Library	A DLL for Windows platforms for software developers. (A DLL for Windows CE/Windows Mobile is available on request. Please contact us with your platform details.)

3.2   Important Security Notice

> ► The ActiveX control **TBarCode** OCX is now a site-locked ActiveX control. This security enhancement was implemented in response to a reported possible security leak in conjunction with Internet Explorer and an installed **TBarCode** OCX. It is strongly recommended to update all **TBarCode** installations as soon as possible.
> As part of the fix the **TBarCode** OCX methods SaveImage() and SaveImageEx() now are restricted to specific internet zones. This happens only if **TBarCode** OCX runs in Microsoft Internet Explorer as part of a web-page. The allowed zones can be specified in the registry. In addition these functions append the correct image filename extension if the file extension specified by the user does not match the supported image formats. Please refer to the **TBarCode** OCX Developer Reference for details.

3.3   What is TBarCode OCX?

TBarCode OCX is a Microsoft® ActiveX® compliant barcode control. Such an ActiveX control is sometimes called OLE control or OCX control. In German it is named "Steuerelement".

You can use **TBarCode OCX** in:

- Microsoft® Office: barcodes are embedded into documents, forms or databases easily.
- Development environments: integrate barcodes into your own application(s).
- Report-Generators and Reporting Engines: Enhance the output with barcode functionality.
- Web Applications: Use barcode technique as part of browser-based applications.
- Most programming and script languages.

3.4   Scope of this Document

This document shows developers how to use **TBarCode OCX** with various IDEs (integrated development environments) like Visual Studio.

3.5   Restrictions of the Demo Version

In the demo version the barcodes will be drawn with a demo-hint. That means that the word "Demo" or the phrase "www.tec-it.com" is drawn partially over the barcode. The demo-hint does not influence the readability of the barcode in a negative way.

> ► In special cases (e.g. very small or high-resolution barcodes) you may want to test the product without restrictions. To obtain a temporary license key contact sales@tec-it.com.
>
> ► For enabling the full-featured version (without the demo hints) you can obtain a license key from TEC-IT (http://www.tec-it.com/order/).
>
> ► For more information on licensing **TBarCode,** please refer to section A.2.

4 Installation

4.1 Requirements

The following operating systems are supported:

- Microsoft Windows 98
- Microsoft Windows 98 Second Edition
- Microsoft Windows ME
- Microsoft Windows NT 4
- Microsoft Windows 2000
- Microsoft Windows XP
- Microsoft Windows Server 2003
- Microsoft Windows Vista

4.2 Download and Setup

Download **TBarCode** from http://www.tec-it.com/Download ► *Software* ► *Barcode Tools for Microsoft Windows* and execute the setup application. Depending on your selection one or more of the following components are installed – along with documentations and sample applications:

- **TBarCode OCX**
  A barcode ActiveX control for Microsoft Office users and for software developers. User manual and API reference are included in the setup.
- **TBarCode Library**
  A DLL for software developers. Developer manual and API reference are included in the setup.
- **TBarCode .NET**
  For software developers. Developer manual and API reference are included in the setup.

►	Administrative rights are required to install.
►	**TBarCode OCX** is an ActiveX control. ActiveX controls must be registered within the Microsoft Windows operating system before they can be used. This registration is executed automatically by the setup application. On demand you can register the OCX file manually with the command line tool *"regsvr32.exe"*.

5 Microsoft Development Environments

5.1 Microsoft Visual Basic 6

In order to insert **TBarCode OCX** in Visual Basic 6, please execute the following steps:

- Open or create your VB Project.
- Right-click the Toolbox and select *Components* from the menu.
- Select *TBarCode OCX 8.0 Type Library* from the list (Insertable Objects) and confirm with *OK.*
- Now you can use **TBarCode** as VB Control (it is shown in the Toolbox).
- Set the characteristics of the barcode in the VB properties window *("Properties TBarCode81").*
- In case you prefer the **TBarCode** *"Property Pages"* (instead of the VB properties window), right-click the object and select *Properties* from the pop-up menu.

5.2 Microsoft Visual C++ 6.0

5.2.1 MFC Applications

In order to use **TBarCode OCX** in MFC, please follow these steps:

- Open or create your MFC based application.
- If you create a new project, make sure, that *ActiveX Controls* ❶ is checked (default) on page 3 of the *"MFC AppWizard".*
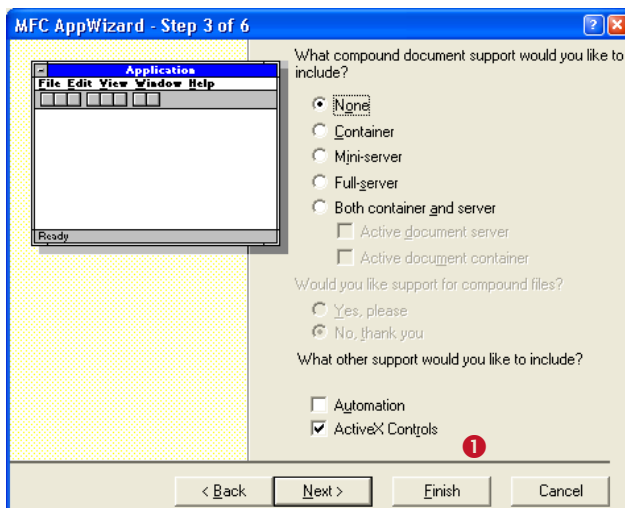

Figure 1: Use TBarCode in MSVC++ 6.0 (MFC Application) (1)

- Select *Project ► Add to Project ► Components and Controls…* from the menu.
- Change to the folder *Registered ActiveX Controls.*
- Select *TBarCode* from the list and click *Insert.*
- A wrapper class for **TBarCode OCX** is generated automatically (❷). Click *OK* to confirm.
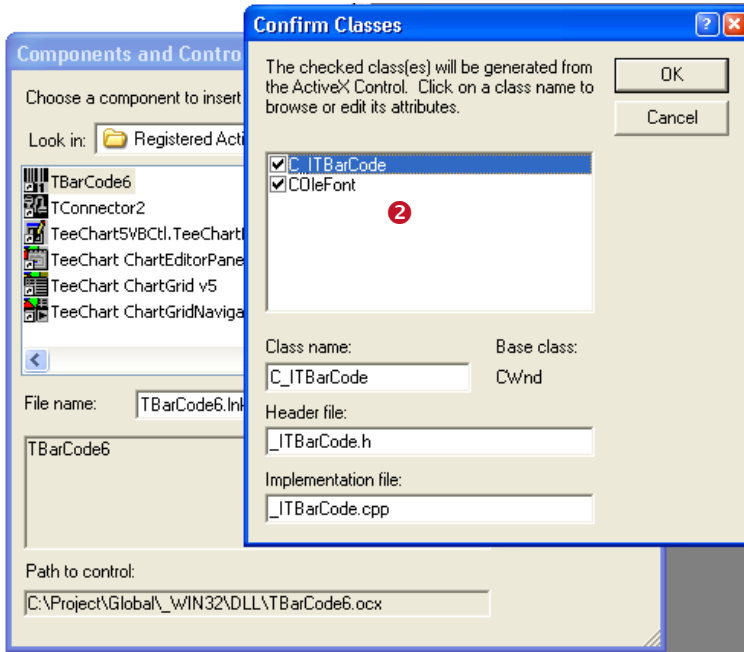
Figure 2: Use TBarCode in MSVC++ 6.0 (MFC Application) (2)

- The control is thereafter selectable in the control toolbox in the dialog editor (❸).
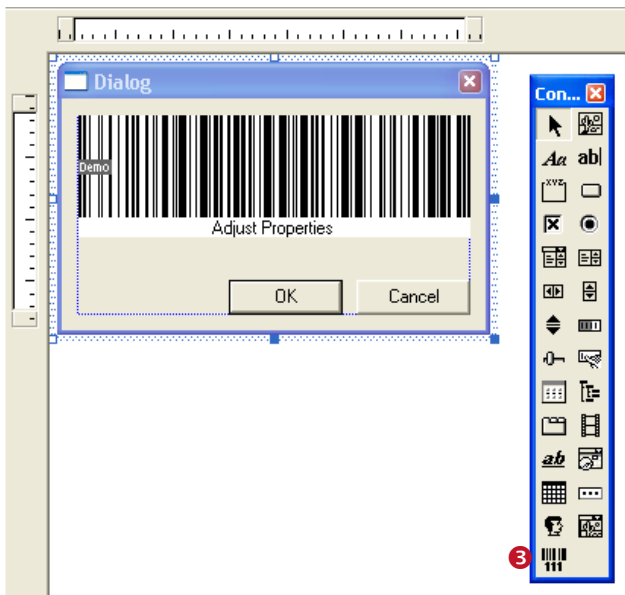- Work with the assistance of the *ClassWizard* if necessary.

Figure 3: Using TBarCode in MSVC++ 6.0 (MFC Application) (3)

5.2.2   Non MFC Application

For non MFC applications we recommend to use **TBarCode DLL**.

5.3 Microsoft Visual Studio .NET 2003

► **TBarCode OCX** can be used in .NET applications. However, we recommend to use **TBarCode .NET** instead. **TBarCode .NET** includes a .NET barcode class and .NET controls for Windows Forms and ASP.NET 2.0 applications.

5.3.1 Use TBarCode OCX as Control

In order to use **TBarCode OCX** as a control and to embed it in a form in Visual Studio .NET (e.g. in a *"Windows Application"* or in an *"ASP.NET Web Application"),* please follow these steps:

- Open or create a project.
- In the Toolbox switch to *Components* (❶).
- Right-click inside the Toolbox and select *Add/Remove Items….* A dialog appears.
- In the dialog switch to the *COM Components* tab ❷.
- Select *TBarCode8* ❸ (check the checkbox).
- Confirm with *OK* (❹).
- Now **TBarCode OCX** is added to the Toolbox (❺).
- To use the control just drag it from the Toolbox (❺) to your form (❻).
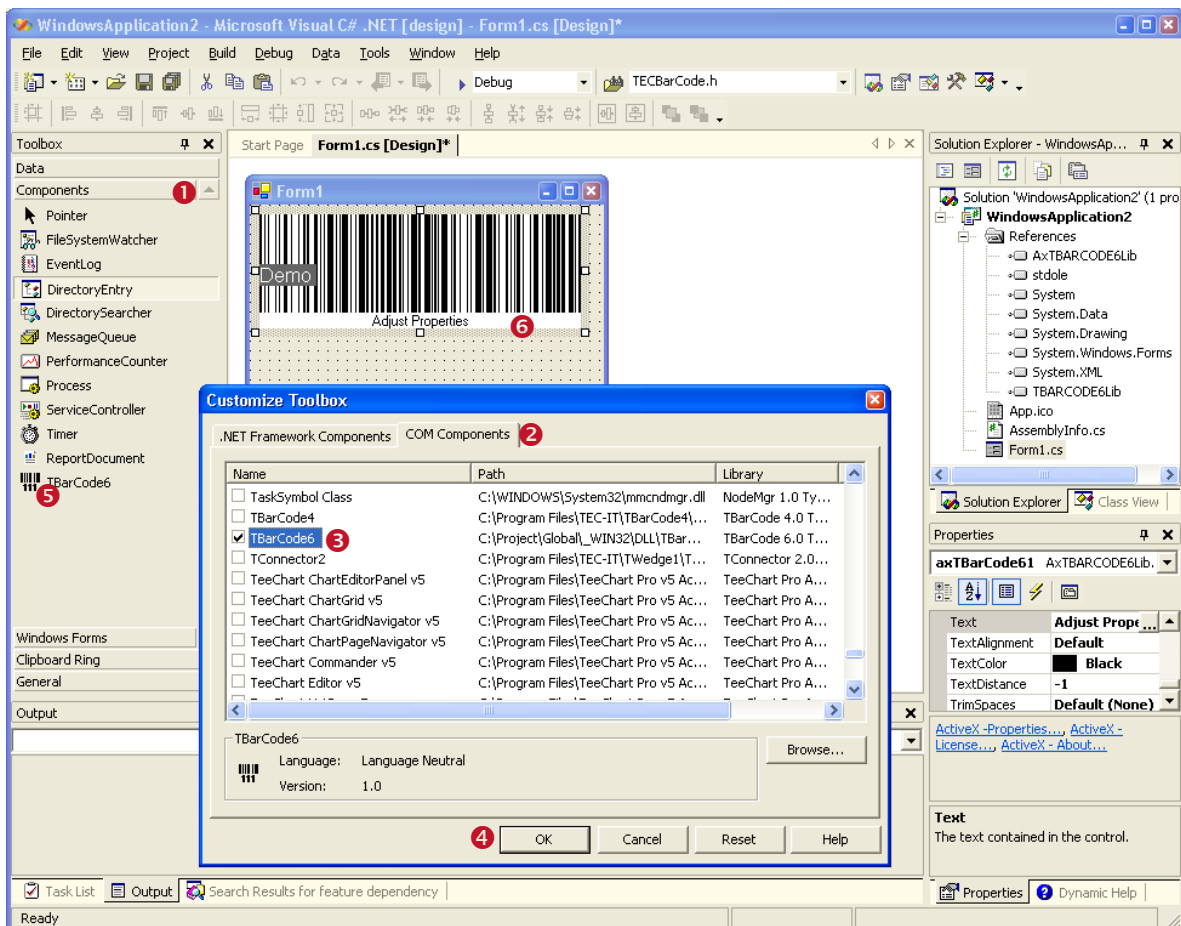

Figure 4: Using TBarCode in Visual Studio .Net 2003 (as Control)

Microsoft Visual Studio .NET automatically generates a wrapper class for an existing ActiveX control. You can use it like any other control in .NET.

5.3.2 Use TBarCode OCX as Library

In order to use **TBarCode** per code only (without embedding it as ActiveX control in a form), please follow these steps:

- Open or create a project.
- From the menu select *Project ► Add Reference…*
- In the upcoming dialog switch to the *COM* tab ❶.
- Select *TBarCode* from the list and click *Select* ❷.
- Confirm with *OK* (❸).
- Now *TBarCode8Lib* is added to the References of your Project (and can be found in the *"Solution Explorer"* Window ❹).
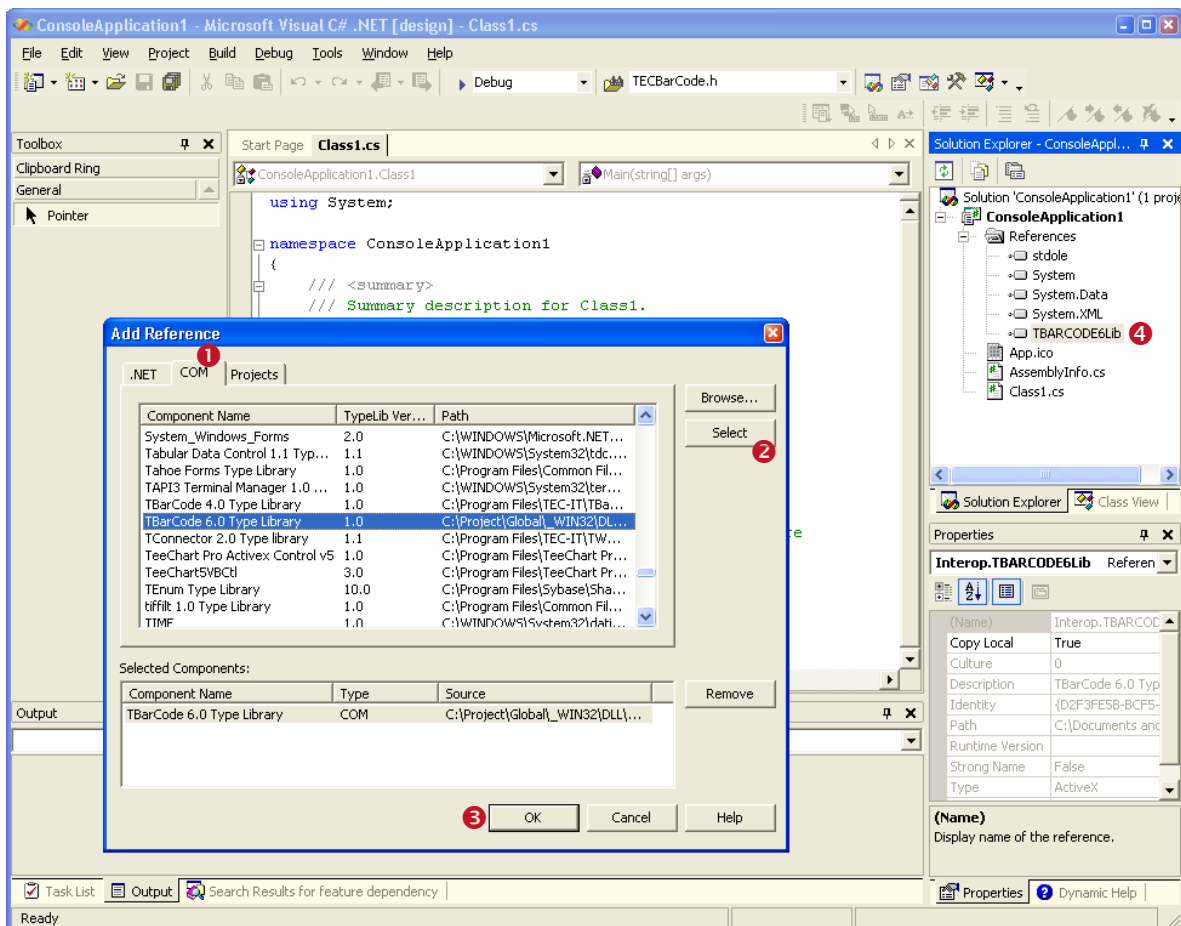- The **TBarCode** class can now be used in your code.

Figure 5: Using TBarCode in Visual Studio .Net 2003 (as Library)

Sample usage of the **TBarCode** class in C#:

```
TBarCode8Lib.TBarCode8Class obj = new TBarCode8Lib.TBarCode8Class();
obj.BarCode = TBarCode8Lib.e BarCType.eBC 2OF5IL;
obj.Text = "12345";
```

5.4 Microsoft Visual Studio .NET 2005

5.4.1 Use TBarCode OCX as Control

In order to use **TBarCode OCX** in Visual Studio .NET (e.g. in a *"Windows Application"* or in an *"ASP .NET Web Application"),* please follow these steps:

- Open or create a project.
- Right-click inside the Toolbox ❶ and select *Choose Items….* A dialog appears.
- In the dialog switch to the *COM Components* tab ❷.
- Select *TBarCode* ❸ (check the checkbox).
- Confirm with *OK* (❹).
- Now **TBarCode OCX** is added to the Toolbox (❺).
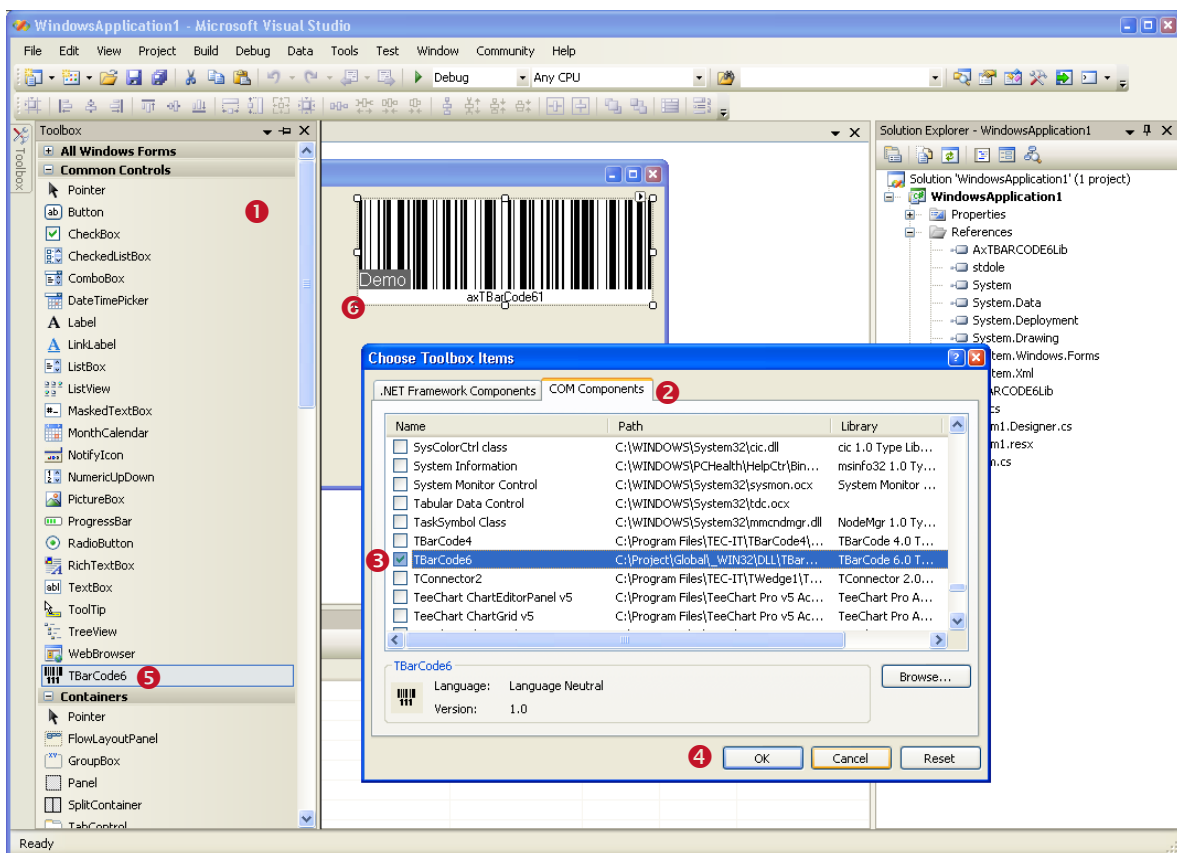- Drag the control from the Toolbox (❺) to your form (❻).

Figure 6: Using TBarCode in Visual Studio .Net 2005 (as Control)

Microsoft Visual Studio .NET automatically generates a wrapper class for an existing ActiveX control. You can use it like any other control in .NET.

5.4.2 Use TBarCode as Library

Use the same steps as described in 5.3.2.

5.5 Microsoft Visual FoxPro

5.5.1 Inserting TBarCode into a FoxPro-Project

In order to insert **TBarCode** in Visual FoxPro, please execute the following steps:

- Open or create a FoxPro Project.
- Open or create a Form.
- Select *OLE Container* from the Standard Toolbox.
- After drawing the outlet of the OLE container, an *"Insert object"* dialog is opened. Select *TBarCode* (Insert Control) and confirm with *OK.*
- The characteristics can be modified in the properties window.
- If you prefer the **TBarCode OCX** property pages, right-click the control and choose *Properties* from the pop-up menu.
- Within the Project Manager you can add the newly created form to the project.

5.5.2 Reports & Labels

If you want to print barcodes in reports or labels, you need a table with a specific column, in which **TBarCode** must be stored as OLE Control (data type "general"). This data field must be initialized with the data structure of a barcode object before use. For this purpose you need an instance of a **TBarCode** object as template (e.g. on a form, it may be invisible).

Example code for the initialization of the data fields of a barcode object:

```
FOR n = 1 TO 10
     INSERT INTO Table1 FROM MEMVAR
     APPEND GENERAL Table1.BC CLASS TBarCode8.TBarCode8
     WITH THISFORM.[Name of the TBarCode object instance]
     .CONTROLSOURCE = "Table1.BC"
     .REFRESH
     .barcode = 20 && Barcode-Type = Code128
     .TEXT = "000070000041"
     .printdatatext = .T.
     .BACKCOLOR = 16777215
     .FORECOLOR = 0 && 16711680 -> (blue)
     .REFRESH
     ENDWITH
NEXT
```

After this code has been executed, an OLE Picture/ActiveX Bound Control can be inserted into a Report or Label Form. In this sample, you have to specify "Table1.BC" as data source (field). Please take a look at our FoxPro-Sample, which can be downloaded from http://www.tec-it.com/Download ► *Bar Code Software for Windows* ► *Samples!*

6 Microsoft ERP Systems

6.1 Microsoft Navision® / Dynamics® NAV

Under Navision **TBarCode OCX** cannot be inserted into a report directly. So you have to use the following workaround:

- Create or open a report and insert a *Picture Box.*
- Create a barcode image (per code) using the **TBarCode OCX** Control.
- Load the image to the *Picture Box.*

Sample code for creating a barcode image:

```
COM objBC;                                    // this is com object instantiation
objBC = new COM("TBarCode8.TBarCode8.1");     // create class TBarCode8

objBC.Text = "121212"                         // set barcode data
objBC.Barcode = 20;                           // 20 ... use Code-128

objBC.SaveImage(DateiNameBmp, 4, pxWidth, pxHeight, nXRes, nYRes); // 4 = eIMBmp (Bitmap)
```

For more detailed information, please take a look at our Navision Tutorial, which can be downloaded under http://www.tec-it.com/Download ► *Bar Code Software for Windows* ► *Samples* or contact our technical support.

7 Borland Delphi / Borland C++ Builder
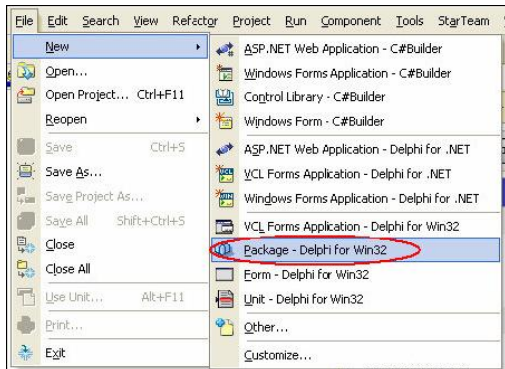
7.1 Using TBarCode OCX in Borland Delphi 6/7

To insert **TBarCode OCX** in Delphi or Borland C++ Builder, please follow the steps described below:

- Select *Components ► Import ActiveX Control* from the menu.
- Select *TBarCode OCX Type Library* from the list.
- Click the *Install* button and confirm the following pop-up windows with *OK.* The **TBarCode** control can now be used like a standard component (Toolbar ActiveX).

7.2 Using TBarCode OCX in Borland Delphi 2005 (Delphi for Win32)

To insert **TBarCode OCX** in Delphi 2005, please follow these steps:
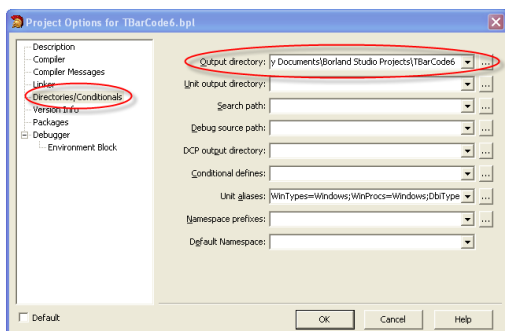
7.2.1 Create and Save a Package for TBarCode OCX

Create Package:
- Choose *File ► New ► Package – Delphi for Win32* from the menu.

Save Package:
- Choose *File ► Save All* from the menu.
- The *Save Package As* dialog will open.
- Choose the directory for saving the package. Hint: It is recommended to create a separate directory for the package.
- Then assign a name to package-project (e.g. *TBarCode8*).
- Confirm with *OK.*

7.2.2 Set the Project Options

- Choose *Project ► Options…* from the menu.
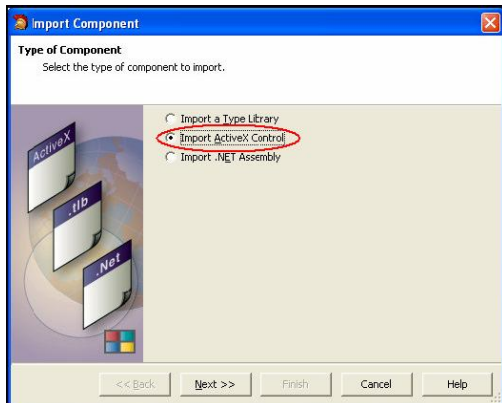- The *Project Options Dialog* will open.

Description:
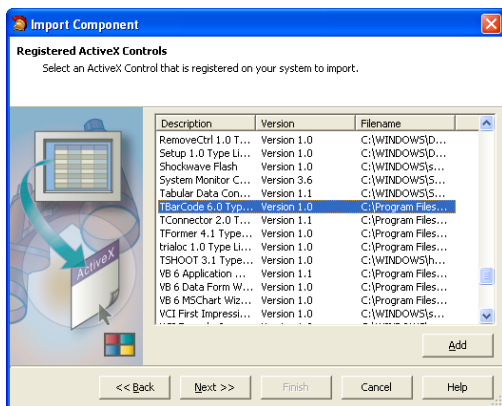- Select *Description* in the tree on the left. The description text can be edited on the right.

Output Directory:
- Select *Directories/Conditionals* in the tree on the left.
- Under *Output directory,* choose the directory where you have saved your project.
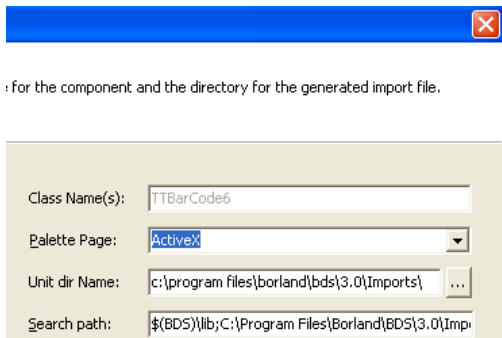- Confirm with *OK.*

7.2.3 Import TBarCode Component

- Choose *Component ► Import Component…* from menu.
- The *Import Component* wizard will open.
- In the Wizard, choose *Import ActiveX Control.*
- Then click *Next.*

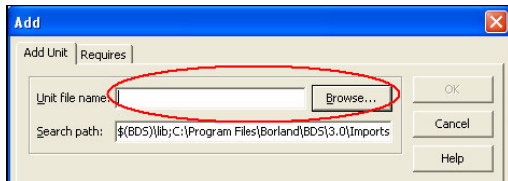- Choose *TBarCode OCX* from the list.
- Click *Next.*

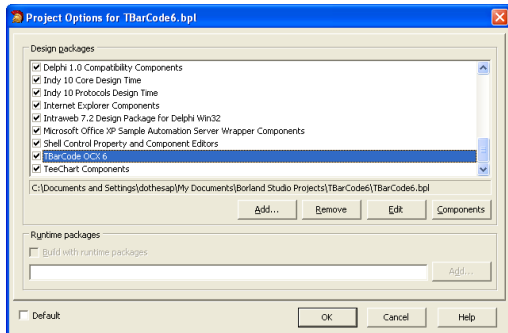- On this page leave all default values.
- Click *Next.*

- This is the last page of the wizard.
- Choose *Create Unit.*
- Then click *Finish.*
- You will find *TBARCODE8Lib_TLB* in the main window of Delphi.
- Save this Unit in the same directory where the package-project is located.

7.2.4 Add Components to the Package

- Choose *Project ►Add to Project…* from the menu.
- The *Add* dialog will open.
- Select the "*Add Unit*" tab.
- Click *Browse* to look for *TBARCODE8Lib_TLB.PAS.*
- Click *OK* to add **TBarCode**.

7.2.5 Create and Install Package

Create the package:
- Select *Project ► Build <ProjectName>* from the menu.

Install the created Package:
- Select *Component ► Install Packages* from the menu.
- Click *Add*.
- Choose the created package (here *TBarCode6.BPL*) and click *Open.*

- The package (*TBarCode8 OCX*) is now listed in *Design Packages.*
- Click *OK* to finish the installation.

7.2.6 Create a new VCL Forms Application

- Choose *File ► New ► VCL Forms Application…* from the menu.
- In the *Tool Palette* you can find the new category *ActiveX*.
- **TBarCode OCX** was added in this category and is now ready to use.

8 Sybase PowerBuilder

8.1 TBarCode OCX in PowerBuilder DataWindow

In PowerBuilder[1] the use of ActiveX controls (like **TBarCode OCX**) is not possible within so-called DataWindows if the control is bound to the band layer. This limitation is not caused by **TBarCode OCX**.

To get dynamic "data driven" barcodes you have to use the following workaround.

Instead of placing the **TBarCode OCX** control inside the DataWindow, you have to insert it into the dialog Window. Thus it can be programmed for our workaround as follows.

By using this control you can create a bitmap file for each barcode, save it to disk and load it to a picture data field in the DataWindow. That has to be done for each data row/record set.

This workaround is described in detail in the following sections.

8.2 Insert TBarCode OCX

To insert **TBarCode OCX** into a PowerBuilder dialog window, please execute the following steps:

- Select *Controls* ► *OLE* from the menu.
- Select the page *Insert Control* from the tab (PowerBuilder 10).
- Select *TBarCode* from the list.
- If desired, change the properties in the context menu (OCX Properties).

► In PowerBuilder Version 7, 8, 9 and 10 always insert **TBarCode** with *Insert Control* because *Create New* does not work with scripting.
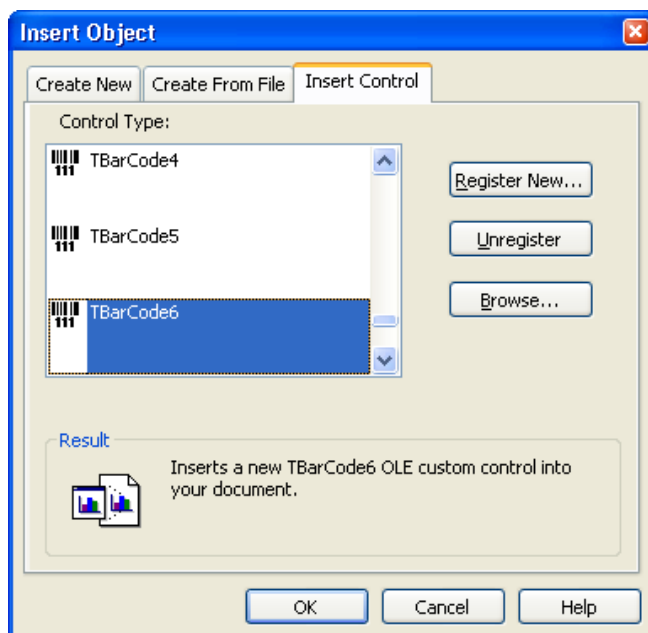

Figure 7: Inserting TBarCode in PowerBuilder

---

[1] PowerBuilder version 7-10

8.3 Script the Control within PowerBuilder 10

To access the properties and methods of the ActiveX control use the following syntax:

```
ControlName.Object.Property
```

Example:

```
ole 1.object.text = "hello"
ole_1.object.barcode = 20            ' 20 = Code 128
```

> ► The barcode type (the symbology) is set by assigning a number to the barcode property. Please refer to the *"Barcode Reference"* (http://www.tec-it.com*)* to learn more.

8.4 Generate Barcode Images for the DataWindow

8.4.1 Generate the Barcode Images (from the Dialog Window)

You need a **TBarCode OCX** object in your dialog window which can be programmed. In our sample it is called "w_barcode_printing".

Open your source code. In the section where you prepare the data for the output you have to do something like the following to save the barcode images:

```
//Image path and filename
ls image = "C:\ APPS\FeeSystem\bc1.jpg"

//Saves the barcode as a JPEG
w_barcode_printing.ole_barcode.SaveImageEx (0, ls_image, 4, 100, 300, 80, 127, 127)
```

> ► Currently PowerBuilder only supports BMP, JPEG, and GIF. You cannot use the EMF file type.

With the *SetItem()* function you can finally assign the name of the generated barcode bitmap file to the column used in the DataWindow (*bc_pic*).

Using this approach will allow you to simulate the data binding inside the DataWindows.

8.4.2 Insert the Barcodes into your DataWindow

Inside your DataWindow you have to create a column with a *STRING* data type. Here we use the name *bc_pic*. This column will contain the path and the name for each barcode bitmap file for each record. Then insert a column control for the created data field.

In the "General" properties of the column control check the *DISPLAY AS PICTURE* box as shown below. This will make the control show the picture and not the filename inside the control frame.

For example: If the column contains the value "c:\picture.jpg", the control will display the picture and not the text "c:\picture.jpg".
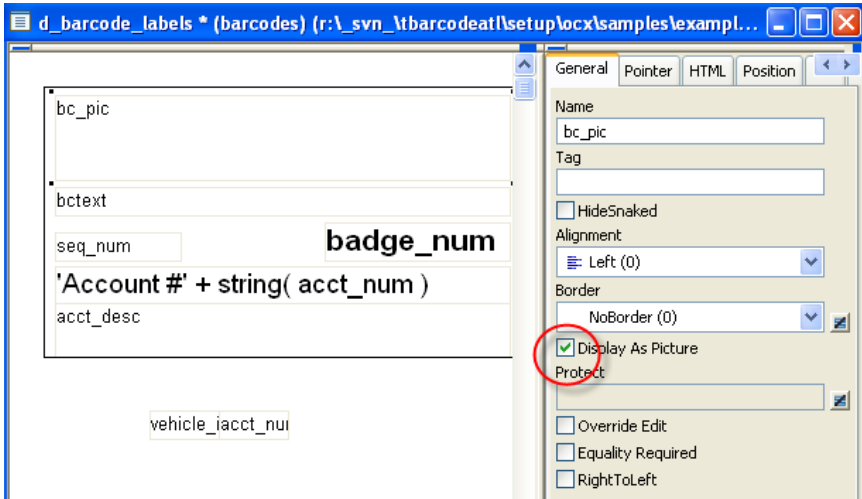
Figure 8: PowerBuilder DataWindow "Display As Picture"

Instead of using a column control you may also insert a computed field control and use the *BITMAP()* function:
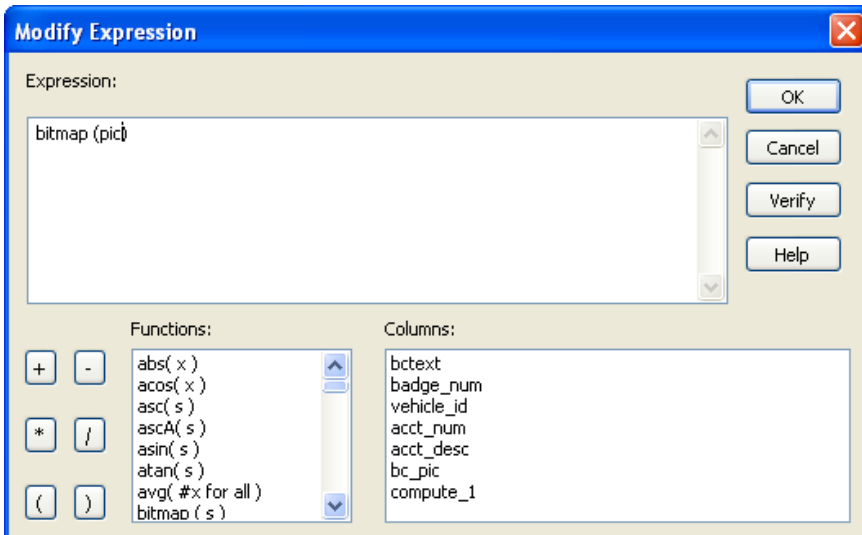

Figure 9: PowerBuilder DataWindow Bitmap function

9   Oracle Forms / Oracle Reports

9.1   Barcodes in Forms

In order to use **TBarCode OCX** in an Oracle Form please execute the following steps:

- Create a new form.
- In the layout editor:
  a) Draw an ActiveX control.
  b) Right-click with the mouse and select *Insert Object.*
  c) Select *TBarCode*.
- Select *Menu ► Program OLE Importer* from the menu and import the control.

Optional:

- Create a button and add a *when-button-pressed* trigger on the canvas (where you can call the ActiveX control).
- Close the layout editor (This step is VERY IMPORTANT!).
- Run the form.
- Press the button to run the ActiveX control.

9.2   Barcodes in Reports

► Oracle Reports do not support ActiveX controls.

Workaround: The user can call an executable from within a report. For instance you can generate barcode bitmaps with the TEC-IT product **Barcode Studio** which can be executed in command line mode (without a visible window). **Barcode Studio** creates and saves a bitmap whenever it is called. Afterwards the generated bitmap can be loaded into an image inside the report.

Another possibility is to use the **TFORMer** product family (advanced label and form reporting tool).

Use **TFORMer Designer** to design your barcode label or report. Thereafter use **TFORMer Runtime** (or the command line tool *TFPrint.exe)* to generate and print the complete report. **TFORMer** supports direct printer output, PostScript, PDF and HTML output directly (without any additional software).

10 Redistributing TBarCode OCX

This chapter explains what is important when redistributing a custom application that uses the **TBarCode OCX** control.

►	Please note that in most cases you need a developer license for re-distribution of **TBarCode OCX** (except for in-house applications which are bound to one or more sites).

10.1 Dependencies

An application that uses **TBarCode OCX** requires the following files:

- **TBarCode8.ocx**
  This is the ActiveX component. This file is mandatory.
- **TBarCode8.dll**
  This is a native Win32 library that performs barcode generation. This file is mandatory.
- **zlib1.dll**
  This is a 3rd-party library that performs data compression. This file is optional. It is only required, when data compression in barcodes is enabled.

These files are located in the folder *C:\Program Files\TEC-IT\TBarCode8\Bin* (or in the folder you have chosen during setup).

TBarCode OCX requires also the **Microsoft VC80 CRT and ATL DLLs**. See next section how to distribute them with your application.

10.2 Redistribution

When redistributing a custom application the files described above need to be redistributed together with the application. The DLLs should be located in the same folder as the executable.

Other files than those listed above must not be redistributed.

You may have to redistribute the **Visual C++ 2005 SP1 runtime components** (MS CRT8.0 and ATL 8.0 DLLs) with your application and ensure they are installed on the target computer.

There are two options to install them:

- You can install these components with the *Microsoft Visual C++ 2005 SP1 Redistributable Package (x86)* available at
  http://www.microsoft.com/downloads/details.aspx?familyid=200b2fd9-ae1a-4a14-984d-389c36f85647
  The package installs all required runtime DLLs.
- If you use a setup tool and your installer supports merge modules (*.msm files) you can add the required msm packages to your setup:
  - *Microsoft_VC80_CRT_x86.msm*
  - *policy_8_0_Microsoft_VC80_CRT_x86.msm*
  - *Microsoft_VC80_ATL_x86.msm*
  - *policy_8_0_Microsoft_VC80_ATL_x86.msm*

Please contact TEC-IT Support if you need help.

11 Contact and Support Information

TEC-IT Datenverarbeitung GmbH

Address:	Wagnerstr. 6
	AT-4400 Steyr
	Austria/Europe
Phone:	+43 / (0)7252 / 72 72 0
Fax:	+43 / (0)7252 / 72 72 0 – 77
Email:	mailto:support@tec-it.com
Web:	http://www.tec-it.com

Appendix A: FAQ

If you have questions not covered in these FAQ, you can also check our website http://www.tec-it.com/FAQ or contact our support team: mailto:barcode@tec-it.com.

A.1   How to set a specific Module Width (15 mils) in Web Applications?

Please check out the document *"Barcode Reference"* (http://www.tec-it.com*).*

A.2   License TBarCode in my Application

Assume you got the following license key:

```
Product ID:      32 (1D Barcodes in V8)
Licensee:        Barcode Inc.
License Kind:    2 (Site)
Licenses:        1
License-Key:     12A3B5C6
```

Now implement a call to the method *LicenseMe* before you draw a barcode (we suggest at startup of your application).

Example in Visual Basic (enumerators are used):

```
TBarCode81.LicenseMe "Barcode Inc.", eLicKindSite, 1, "12A3B5C6", eLicProd1D
```

The Product ID depends on the version of **TBarCode** (see your license data).

A.3   Clipped Barcode Image (SaveImage)

If the *ModuleWidth* property is set to a specific value the barcode may be clipped because it does not fit into the image dimensions specified in *SaveImage*.

A.3.1   Solution

If you need a specific module width for your application (e.g. 3 pixels), use the number of barcode modules (*CountModules* property) and multiply this value with 3 (to get 1 module = 3 Pixels wide). Use this value as horizontal size of the image in *SaveImage*. Using this method, the image size will adapt automatically depending on number of horizontal barcode modules.

For 2D symbologies use *Get2DXCols* and *Get2DXRows* for finding the best horizontal and vertical size values.

A.4   Unreadable Barcodes with PrintForm (VB)

In Visual Basic the form objects (like **TBarCode**) are rendered with (low) screen resolution and not with the (usually high) printer resolution. Therefore the printed barcode may become unreadable.

To work around this problem adjust the module width of the barcode to the resolution of the screen:

> ►   This can be achieved by using the *CountModules* property.
> ►   Or use the **TBarCode** property *OptResolution* = true

For details please see our Visual Basic Sample Code, which is installed together with the product setup. The samples contain also the PrintForm workaround to get a readable barcode.

A.5   Save MaxiCode with a higher Resolution

You have to set the *ModuleWidth* property to a higher value (as it is by default) – and increase the values for XSize / YSize when saving the barcode.

Example:

```
FlName = "Maxicode.bmp"
TBarCode81.Text = "BarCodeText"
TBarCode81.BarCode = eBC MAXICODE
'Use nRatio to enlarge or reduce image size
nRatio = 2
TBarCode81.ModuleWidth = 1500 * nRatio
DoEvents
YSize = 208 * nRatio
XSize = 215 * nRatio
TBarCode81.SaveImage "C:\Users\User\Documents\" & FlName, TBarCode8Lib.eIMBmp, XSize,
YSize, 72, 72
'Hint  : When printing we recommend to use the standardized size of MaxiCode
'Height: ~24.4 mms;
'Width : ~25,2 mms
```

A.6   SaveImage does not work

Please check if you have write permissions to the directory you want to write to (e.g. important in web applications).

Please check if the internet zone was unlocked when using **TBarCode** OCX as part of a web page. See the *TBarCode OCX Developer Reference* for details.

A.7   Barcodes created with SaveImage are unreadable

Please check out the document *"Barcode Reference"* (http://www.tec-it.com*).*