

Anleitung zur TBarcode-Komponente für DELPHI (16- und 32-Bit-Version)

Wichtige Informationen!

Bitte lesen!

Inhalt dieses Dokuments:

1. Allgemeine Beschreibung
2. Installation der Komponente
3. Demoprogramm/Gebrauch der Komponente
4. Autorenkontakt
5. Nutzungsrecht
6. Technische Informationen für lizenzierte Anwender

1. Allgemeine Beschreibung

TBarcode ist ein Programmierwerkzeug für das DELPHI-Entwicklungssystem von Borland und ist sowohl in einer 16-Bit-Version (Delphi 1.0) als auch in einer 32-Bit-Version (Delphi-2) lieferbar.

TBarcode unterstützt das Anzeigen und Ausdrucken von Barcodes in allerhöchster Qualität. Als visuelle Komponente fügt sich Tbarcode bei der Installation in die vorhandene Komponentenpalette ein und kann genauso einfach verwendet werden wie alle anderen visuellen Komponenten, die bereits mit Delphi zusammen ausgeliefert werden.

TBarcode unterstützt zur Zeit folgende Codetypen:

- EAN-8
- EAN-13
- UPC-A
- UPC-E
- Code 39
- Code 2/5 interleaved
- Code 2/5 interleaved (Frachtpostversion)

Die gesamte TBarcode-Funktionalität ist **in der kompilierten EXE-Datei enthalten**, sie brauchen daher zusammen mit Ihrer fertigen Anwendung **keine zusätzlichen Dateien** weitergeben.

2. Installation der Komponente

Bitte beachten Sie, daß TBarcode als 16-Bit-Version für Delphi 1.0 sowie als 32-Bit-Version für Delphi-2 erhältlich ist und installieren Sie bitte die "richtige" Version.

Warnung: Wenn Sie das im Sourcecode beigefügte Beispielprogramm BARDEMxx.DPR in den Compiler laden, ohne daß die TBarcode-Komponente vorher korrekt installiert ist, dann wird der Delphi-Compiler den Fehler entdecken und die im Formular vorhandene TBarcode-Komponente automatisch aus dem Beispielprogramm entfernen. Das Demoprogramm wäre dann unbrauchbar und müßte neu installiert werden.

Deshalb: Installieren Sie erst die TBarcode-Komponente und rufen Sie dann erst das Beispielprogramm BARDEMxx.DPR auf, machen Sie es auf keinen Fall umgekehrt!

- Außerdem ist darauf zu achten, daß eine neuere Version der TBarcode-Komponente erst dann installiert werden kann, wenn die alte Version vorher de-installiert wurde. Hinweise zur De-Installation von Komponenten finden Sie weiter unten im Text.

Erster Schritt: Verzeichniswahl

- Machen Sie sich Gedanken, in welchem **Verzeichnis** Sie die Komponente installieren möchten.
- Es ist nachträglich nicht ohne weiteres möglich, die Komponentendateien in ein anderes Verzeichnis zu verschieben. Sobald Delphi die Komponentendateien nicht mehr in dem Verzeichnis finden kann, von dem aus sie installiert wurden, werden Ihre Anwendungen, die diese Komponente verwenden, nicht mehr einwandfrei kompilierbar sein.
- Vorschlag: Vermischen Sie die Komponente nicht mit irgendwelchen Projekt-, Formular- oder Quellcodedateien, sondern spendieren Sie dafür ein eigenes Verzeichnis. Verschieben Sie dazu folgende Dateien in das gewünschte TBarcode-Verzeichnis (das ist ein Vorschlag, keine Bedingung):

BARCODxx.PAS	{installierbare xx-Bit-Komponente}
BARCODxx.DCR	{Bitmap für die xx-Bit-Komponentenpalette}
BARxx.RES	{xx-Bit Resourcendatei}
BARxxLL.DCU	{Barcode Low-Level-Routinen xx-Bit}
BARxxHL.DCU	{Barcode High-Level-Routinen xx-Bit}

In der 16-Bit-Version steht "xx" für "16".

In der 32-Bit-Version steht "xx" für "32".

Eine vollständige Dateiliste finden Sie in der Datei FILES.TXT beigefügt.

Zweiter Schritt: Komponente Installieren

- Wenn sich diese fünf Dateien im gewünschten Verzeichnis befinden, starten Sie bitte Delphi und rufen Sie den Menüpunkt "**Komponente - installieren**" auf.
- Klicken Sie dann auf "**Hinzufügen**". Dann klappt der Dialog "Modul hinzufügen" auf.
- Klicken Sie nun auf "**Durchsuchen**" und wählen Sie im Dateidialog die Datei **BARCOD32.PAS** (bzw. **BARCOD16.PAS**) in dem Verzeichnis aus, in dem sich die fünf oben genannten Dateien befinden. Bestätigen Sie den Dialog mit **OK**. Delphi benötigt dann einige Zeit, um die Komponentenbibliothek neu aufzubauen. Danach steht Ihnen unter der Notetab-Beschriftung "Schlottke" die Komponente "TBarcode" zur Verfügung. Den Eintrag finden Sie als zuletzt eingefügten ganz rechts in der Komponentenpalette.

Wichtig: Bei der Installation der Komponente darf NUR die Datei BARCOD32.PAS (32-Bit-Version) oder BARCOD16.PAS (16-Bit-Version) hinzugefügt werden. Wenn Sie zusätzlich andere Dateien hinzufügen, könnte dies zu Fehlermeldungen beim Aufbau der geänderten Komponentenbibliothek führen!

De-Installation

Wenn Sie die TBarcode-Komponente nicht mehr benötigen oder bevor Sie eine aktualisierte Lizenzversion installieren oder bevor Sie die Komponentendateien in ein anderes Verzeichnis verschieben, entfernen Sie die installierte Komponente wie folgt:

- Starten Sie Delphi und rufen Sie den Menüpunkt "**Komponente - installieren**" auf.
- Suchen Sie in der Liste "**Installierte Module**" den Eintrag "**Barcode**" und klicken Sie den Eintrag an.
- Klicken Sie nun auf "**Entfernen**"
- Entfernen Sie ggf. auch den Eintrag für den TBarcode-Pfad aus dem "Suchpfad"
- Bestätigen Sie den Dialog mit **OK**, die Komponente wird dann entfernt.
- Es ist möglich, gleichzeitig mit dem "Entfernen" auch "Hinzufügen" auszuführen. Damit kann das Update einer vorhandenen Komponente auch in einem Schritt erfolgen.

Wenn Sie eine Update-Version mit einer geänderten TBarcode-Komponente installieren, ist es sinnvoll, vorher **alle Projekt-, Formular- und Quellcodedateien zu sichern**, die diese Komponente enthalten. In manchen Fällen zerstört Delphi alte Projekt-Formulare ganz oder teilweise, wenn neuere Komponenten-Versionen geladen werden. Im Notfall müssen Sie beim Update alle "alten", visuellen TBarcode-Objekte aus den Formularen entfernen, dann das TBarcode-Update in der Komponentenbibliothek installieren und anschließend die "neuen" TBarcode-Objekte wieder im Formular einfügen. Im Normalfall sollte dieser Umweg aber nicht erforderlich sein.

3. Demoprogramm/Gebrauch der Komponente

Wenn die Komponente wie unter Punkt 2 beschrieben installiert ist, können Sie das im Quellcode beigefügte Beispielprogramm BARCOD32.DPR (bzw. BARCOD16.DPR) aufrufen und etwas herumexperimentieren. Im Object-Inspektor können Sie die Eigenschaften der im Formular enthaltenen TBarcode-Komponente ändern und die Auswirkungen direkt auf dem Formular beobachten. Noch komfortabler geht es, wenn Sie das Beispielprogramm kompilieren und starten.

Das Beispielprogramm stellt einen Barcode auf dem Bildschirm dar und ermöglicht die Manipulation aller Barcode-Eigenschaften und Optionen. Außerdem wird anhand von zwei verschiedenen Druckfunktionen beispielhaft gezeigt, wie Barcodes ausgedruckt werden können.

- **Property Bar_Caption**

Die Eigenschaft "Bar_Caption" enthält den eigentlichen Code, d.h. die zu codierenden Zeichen. Eine Eingabeüberprüfung findet nicht statt, aber der darzustellende Code wird ggf. korrigiert, wenn Bar_Caption nicht als gültiger Barcode dargestellt werden kann. Je nach

Barcodetyp können nur Ziffern codiert werden (EAN, UPC, Code 2/5i) oder Ziffern und Buchstaben (Code39). Außerdem unterscheidet man Codes mit fester Stellenzahl (EAN, UPC) und solchen mit variabler Anzahl von Nutzzeichen (Code39, Code2/5i). Sofern Bar_Caption nicht direkt in einen gültigen Barcode umgesetzt werden kann, wird die interne Korrektur nach dieser Logik durchgeführt:

- Buchstaben werden zu Nullen, wenn nur Ziffern erlaubt sind (EAN, UPC, Code2/5)
- Zu lange Zeichenfolgen bei EAN und UPC werden entsprechend gekürzt
- Zu kurze Zeichenfolgen bei EAN und UPC werden mit Nullen aufgefüllt
- EAN/UPC-Prüfziffern werden automatisch erzeugt, falsch angegebene korrigiert
- Kleinbuchstaben werden zu Großbuchstaben (Code39 erlaubt nur Großbuchstaben)
- Code 2/5interleaved benötigt eine gerade Anzahl von Ziffern, falls eine ungerade Anzahl angegeben wird, wird automatisch eine Prüfziffer erzeugt.

Die mögliche Stellenzahl bei EAN- und UPC-Codes beträgt:

- EAN-8: 7 Nutzzeichen
- EAN-13: 12 Nutzzeichen
- UPC-A: 11 Nutzzeichen
- UPC-E: 6 Nutzzeichen
- Code39: beliebige Stellenzahl
- Code2/5: beliebige gerade Stellenzahl

Bei EAN- und UPC-Codes wird immer automatisch eine Prüfziffer errechnet und an die Nutzzeichen angehängt. Beim Code2/5i wird nur dann eine Prüfziffer berechnet, wenn eine ungerade Stellenzahl vorgegeben wird. Bei der Frachtpostversion des Code2/5i erfolgt die Berechnung der Prüfziffer nach dem von der Deutschen Post AG für Frachtpostsendungen vorgegebenen Rechenschema, beim normalen Code2/5i nach der üblichen Methode.

- **Property Bar_CodeType**

Je nach Anwendungsfall können unterschiedliche Barcodes zur Anwendung kommen. Im Groß- und Einzelhandel sind in Europa die Barcodetypen EAN-8 und EAN-13 üblich, in Amerika die Codetypen UPC-A und UPC-E. Die Abkürzung EAN steht dabei für "Europäische Artikelnummerierung" und beinhaltet ein europaweit genormtes System der Codierung von Herstellungsland, Herstellernummer und Artikelnummer. EAN-Codes haben eine feste Stellenanzahl: Der EAN-8 besteht aus sieben Nutzziffern und einer Prüfziffer. Der EAN-13 hat 12 Nutzziffern und eine Prüfziffer. Das amerikanische Gegenstück dazu ist der UPC-Code (UPC= Universal Product Code), den es ebenfalls in einer langen und einer kurzen Version gibt.

In der Industrie kommen dagegen Barcodes zur Anwendung, die über eine variable Stellenzahl verfügen. Sofern nicht nur Ziffern, sondern auch Buchstaben codiert werden sollen, stellt der Code39 eine gute Wahl dar. Der Code39 unterstützt folgende Zeichen:

0123456789ABCDEFGHIJKLMNPOQRSTUVWXYZ- . \$ / + %

Aufgrund des erweiterten Zeichenvorrats gegenüber den nur-numerischen Codes druckt der Code aber relativ breit.

Wenn nur Ziffern codiert werden müssen, ist der sehr kompakte Code 2/5 interleaved zu empfehlen. Interleaved heißt, daß immer zwei Zeichen zusammen codiert werden, wobei nicht nur die Balken, sondern auch die Lücken Informationen tragen. Der Code 2/5 interleaved kann immer nur eine gerade Anzahl von Stellen haben. Wenn eine ungerade Anzahl von Stellen codiert wird, wird automatisch eine Prüfziffer ermittelt und angehängt. Wenn Bar_Caption eine gerade Anzahl von Stellen enthält, so wird keine Prüfziffer verwendet.

Eine besondere Abart des Code 2/5 interleaved ist die "Frachtpostversion". Seit dem 01.07.1995 werden alle von der Deutschen Post AG transportierten Pakete mit Barcodes versehen. Selbstbucher können Ihre Pakete bereits bei der Paketbeschriftung mit Frachtpost-Barcodes versehen und so Porto sparen. Dabei verwendet die Deutsche Post AG allerdings eine abgewandelte Form der Prüfziffernberechnung, die von der üblichen Norm abweicht. Wenn eine gerade Stellenzahl in Bar_Caption angegeben wird, wird keine Prüfziffer ermittelt und diese Version unterscheidet sich nicht vom normalen Code2/5i. Um Frachtpost-Barcodes zu verwenden, muß daher eine ungerade Stellenzahl (ohne Prüfziffer) in der Eigenschaft Bar_Caption angegeben werden, nur dann wird der korrekte Frachtpost-Code ermittelt.

Überblick über die möglichen Eigenschaften für Bar_CodeType:

- EAN8 ==> Code EAN-8
- EAN13 ==> Code EAN-13
- Code39 ==> Code 39
- Code25i ==> Code 2/5 interleaved
- Code25iP ==> Code 2/5 interleaved (Frachtpostversion)

- **Property Bar_HeightPercent**

Abhängig von der gewählten Strichcodebreite (siehe Bar_ModuleWidth) hat jeder Barcode eine normale Höhe, die mit 100% angesetzt ist. Für die EAN-Codes ist diese normale Höhe in der Norm DIN 66236 festgelegt. Für die anderen Codes, die nicht so stark genormt sind, wird dieselbe 100%-Höhe verwendet wie für den EAN-8-Code. Der Prozentwert für die Höhe kann im Bereich zwischen 20 und 200% frei eingestellt werden.

- **Property Bar_HighDensity**

Diese Eigenschaft bestimmt, ob Barcodes mit der normalen Strichcodebreite ausgegeben werden (Bar_HighDensity:=false) oder ob HighDensity-Barcodes erzeugt werden sollen, die nur die halbe Breite haben (Bar_HighDensity:=true). In den vergangenen Jahren hat die Entwicklung von Barcode-Lesegeräten hoher Auflösung zu vertretbaren Preisen große Fortschritte gemacht. Deshalb wurde besonders von den Produktherstellern der Wunsch geäußert, die Barcodegröße auf den Produktverpackungen zu verkleinern, um mehr Platz für Werbung und Produkthinweise zu haben. Inzwischen sind drucktechnisch hergestellte High-Density-Barcodes üblich, die mit einer geringeren Strichcodebreite als SC0 (siehe Bar_ModuleWidth) arbeiten. Obwohl die PC-Druckertechnologie die dazu erforderlichen Auflösungen derzeit (noch) nicht bietet, ist bereits eine Option für kleinere Barcodes vorhanden. Das Ausdrucken von Barcodes mit "Bar_HighDensity:=true" kann nur auf

Laserdruckern mit mindestens 600 dpi Auflösung zu befriedigenden Ergebnissen führen. Nach Möglichkeit sollten Sie Bar_HighDensity auf "false" eingestellt lassen und die Standardgrößen verwenden. Auf dem Bildschirm sind HD-Barcodes nur sehr grob darstellbar, wenn Bar_ZoomSize auf True gesetzt wird. HD-Barcodes haben jeweils nur die Hälfte der normalen Modulbreite, wie unter Bar_Modulewidth angegeben.

- **Property Bar_ModuleWidth**

Jeder Barcode setzt sich aus einzelnen Modulen zusammen. Diese bilden dann Striche und Lücken von bestimmter Breite. Jeder Strich und jede Lücke hat eine Breite, die einem Vielfachen der Modulbreite entspricht. Für die EAN-Codes ist die Modulbreite in der DIN66236 bis auf 1/1000stel Millimeter genormt.

SC0: Modulbreite 0.270 mm
SC1: Modulbreite 0.297 mm
SC2: Modulbreite 0.330 mm (Das ist die Normalgröße)
SC3: Modulbreite 0.363 mm
SC4: Modulbreite 0.396 mm
SC5: Modulbreite 0.445 mm
SC6: Modulbreite 0.495 mm
SC7: Modulbreite 0.544 mm
SC8: Modulbreite 0.610 mm
SC9: Modulbreite 0.660 mm

Problem: Der Sprung von einer Strichcodegröße zur nächsten beträgt z.B. nur 0,033 mm etwa zwischen SC1 und SC2. Typische PC-Drucker bieten aber nur eine maximale Auflösung, mit der diese Genauigkeit nicht erreichbar ist. Ein Laserdrucker mit 300dpi Auflösung (also 300 Dots per Inch) setzt umgerechnet ($25,4\text{mm}/300$) genau alle 0,0847 mm einen Punkt. Sofern die Normalgröße SC2 mit 0,330 mm gefordert ist, wird beim Ausdrucken eine Modulbreite von 4 Laserpunkten gewählt. Dies entspricht einer tatsächlichen Strichcodebreite beim Drucken von $25,4 \cdot 4 / 300 = 0,339$ mm.

Die nächstkleinere Modulgröße eines 300 dpi-Laserdruckers wäre 3 Punkte je Modul, also $25,4 \cdot 3 / 300 = 0,254$ mm. Dies liegt schon knapp unter der Größe SC0.

Die nächstgrößere Modulgröße eines 300 dpi-Laserdruckers wäre 5 Punkte je Modul, also $25,4 \cdot 5 / 300 = 0,423$ mm. Dies liegt schon fast bei Größe SC5.

Um mathematisch exakte Barcodes ausdrucken zu können, sind daher je nach Auflösung des Druckers nur bestimmte Barcodegrößen möglich!!! Die Rundung der Modulgröße auf ganzzahlige Geräteeinheiten führt daher trotz verschiedener Einstellungen der SC-Größe zu denselben Barcodes. Faustformel: Je höher die Auflösung des Ausgabegeräts, desto mehr Größenstufen sind tatsächlich darstellbar.

Besonders extrem wird diese Problematik bei der Bildschirmausgabe. Rein rechnerisch nimmt Windows eine Auflösung des Bildschirms von 96 dpi an, das sind 0,265mm und dieser Wert dürfte Ihnen vielleicht bekannt vorkommen: Der Lochmaskenabstand gängiger Monitore

weist diesen Wert in ähnlicher Größenordnung auf. Ein Balken von 0,265mm Modulbreite ist also genau ein Bildschirmpixel breit. Wenn man zwei Pixel als Modulbreite nimmt, sind es bereits 0,530 mm. Mathematisch exakte Barcodes lassen sich am Monitor daher nur in zwei verschiedenen Größen anzeigen.

Nur unter Verzicht auf mathematisch exakt konstruierte Barcodes ist es möglich, auch Zwischengrößen anzuzeigen. Dazu bietet TBarcode die Eigenschaft "Bar_ZoomSize". Nähere Angaben dazu finden Sie weiter unten.

- **Property Bar_WidthReduce**

Die Eigenschaft Bar_WidthReduce berücksichtigt eine Korrektur beim Drucken von Barcodes auf bestimmten Druckertypen, die zur sogenannten "Druckausbreitung" neigen. Unter "Druckausbreitung" versteht man den Effekt, daß gleichbreite schwarze und weiße Balken im Ausdruck nicht die gleiche Breite haben, sondern die schwarzen Balken werden durch Druckausbreitung breiter als gewünscht. Dieser Effekt ist etwa bei Tintenstrahldruckern unvermeidbar: Stellen Sie sich einfach vor, daß die als Balken auf das Papier gespritzte Tinte nach rechts und links in die weißen Bereiche (Lücken) fließt. Die normale Saugfähigkeit von gewöhnlichem Offsetpapier reicht aus, diesen Effekt zu bewirken. Im Endeffekt werden die schwarzen Balken zu breit und die Lücken zu schmal. Mit "Bar_WidthReduce:=true" erfolgt die Ausgabe so, daß schwarze Balken jeweils um eine Geräteeinheit schmäler gedruckt werden, als es rein rechnerisch notwendig wäre. Auch bei Matrixdruckern ist eine Strichbreitenkorrektur erforderlich, aber aus anderem Grund: Aus fertigungstechnischen Gründen kann der Durchmesser der Druckkopfnadeln einen bestimmten Wert von ca. 0,16 mm nicht unterschreiten. Dieser Wert liegt deutlich oberhalb der maximalen Druckerauflösung und führt ebenfalls zu einer Druckausbreitung.

Allein aus diesen Schilderungen zum Problemkreis "Druckausbreitung" dürften Sie erkennen, daß Tintenstrahl- und Matrixdrucker nur sehr eingeschränkt zur Erstellung von Barcodes geeignet sind. Optimale Barcodes lassen sich nur auf Laserdruckern erstellen. Beim Drucken auf Laserdruckern ist die richtige Einstellung "Bar_WidthReduce:=false".

- **Property Bar_ZoomSize**

Unter der Eigenschaft Bar_ModuleWidth wurde die Problematik erläutert, mathematisch exakte Barcodes bestimmter Größe auf einem Ausgabegerät mit beschränkter Auflösung zu erzeugen. Je nach Auflösung sind nur verschiedene "exakte" Größen möglich, zwischen denen jeweils Größensprünge liegen. Dazu ein Rechenbeispiel: Bei einem Barcode EAN-13 sind insgesamt 95 Module im Barcode auszugeben. Auf einem Bildschirm mit 96 dpi Auflösung beträgt nun die Mindestgröße 95 Pixel, also etwa 25 mm. Die nächste Größe verwendet zwei Pixel pro Modul und der Barcode wird 190 Pixel breit, verdoppelt seine Breite also. Eine stufenlose Größenanpassung wie bei anderen visuellen Komponenten ist mit Barcodes daher nicht möglich. Eine Abmilderung dieser Einschränkung stellt die Eigenschaft Bar_ZoomSize dar. Mit "Bar_ZoomSize:=true" kann erzwungen werden, daß ein Barcode mit der exakten Größe ausgegeben wird, die genau der Einstellung von Bar_ModuleWidth entspricht. Dabei wird aber auf eine mathematische Exaktheit des Barcodes verzichtet, die

Einstellung "Bar_ZoomSize:=True" ist daher nur für Bildschirmanzeigen zu empfehlen, wenn es mehr auf eine richtige Proportion als auf eine exakte Darstellung ankommt. **Beim Ausdrucken sollte in jedem Fall "BarZoomSize:=False" eingestellt sein.** Nur dann wird ein exakter Code gedruckt, der mit einem Barcode-Lesegerät die optimale Erkennungsrate bringt.

- **Property Code_License**

Die Eigenschaft Code_License kann von Ihnen nicht gesetzt werden, ist aber jederzeit im Objektivinspektor sichtbar. Falls Sie eine nicht registrierte Sharewareversion installiert haben, lautet der Eintrag "Not registered". Die damit erstellten EXE-Dateien dürfen von Ihnen nicht an Endanwender weitergegeben werden. Näheres finden Sie bei den Lizenzbestimmungen.

- **Property Code_Version**

Die Eigenschaft Code_Version kann von Ihnen nicht gesetzt werden, ist aber jederzeit im Objektivinspektor sichtbar. Sie informiert über die Versionsnummer der verwendeten TBarcode-Komponente. Bitte geben Sie bei allen Anfragen an den Autor an, mit welcher Versionsnummer von TBarcode Sie arbeiten.

4. Autorenkontakt

Autor der TBarcode-Komponente ist:

Jürgen Schlottke
Schönaich-Carolath-Str. 46
25336 Elmshorn
Deutschland/Germany

Sofern Sie irgendwelche Rückfragen haben oder wenn etwas unklar ist, Sie erreichen mich wie folgt:

Tel.: 04121/63109
Fax 04121/63971
CIS 100106,3034
Internet: 100106.3034@compuserve.com

Falls es Ihnen möglich ist, stellen Sie Fragen bitte per Compuserve oder Internet.

Die der TBarcode-Komponente zugrundeliegenden Pascal-Routinen wurden 1993 unter Turbo Pascal für Windows erstmals erstellt und in den vergangenen Jahren mehrfach erweitert und verbessert. Die Leistungsfähigkeit des zugrundeliegenden Konzepts ist in mehreren Softwareprogrammen erfolgreich und eindrucksvoll unter Beweis gestellt worden:

- UNIKETT Etiketten- und Barcodedruck (1993)

- Steierprint/VDA Odette-Transportlabels (1994)
- POSKETT K1 Frachtpostbearbeitung (1995)

Mit der TBarcode-Komponente für Delphi sind diese bewährten Routinen nun für jedermann leicht verfügbar und können zudem vor dem Kauf problemlos mit vollem Leistungsumfang getestet werden (Shareware-Prinzip).

5. Nutzungsrecht

Für die Sharewareversion gilt:

Die Sharewareversion der TBarcode-Komponente ist voll funktionsfähig und darf von Softwareentwicklern für Erprobungs- und Testzwecke verwendet werden. Es ist ausdrücklich erwünscht, Kopien der TBarcode-Sharewareversion herzustellen und an interessierte Programmierer weiterzugeben, sofern folgende Bedingungen eingehalten werden: Der Empfänger der Sharewareversion muß eine vollständige und unveränderte Kopie mit allen zugehörigen Dateien erhalten und außerdem in geeigneter Form darüber aufgeklärt werden, daß es sich um ein Programm handelt, das erst durch Zahlung der Lizenzgebühr an den Autoren lizenziert werden muß (Sharewareprinzip). Die Verbreitung der Sharewareversion darf über CD-ROM und andere Datenträger, Mailboxen, Internet oder beliebige andere Datennetze erfolgen, sofern vorstehende Bedingungen erfüllt sind.

Für die Lizenzversion gilt:

Der rechtmäßige Erwerber der Lizenzversion erhält ein Nutzungsrecht analog der Benutzung eines Buches: Entsprechend der Unmöglichkeit, daß ein Buch an verschiedenen Orten von mehreren Personen gelesen wird, darf die Lizenzversion nicht gleichzeitig von verschiedenen Personen, an verschiedenen Orten und auf verschiedenen Geräten benutzt werden. Kopien der Lizenzversion dürfen lediglich zum Zweck der Datensicherung angefertigt werden. Der Erwerber der Lizenzversion darf eigene Programme, die mit Hilfe der TBarcode-Lizenzversion entwickelt wurden, ohne Einschränkung oder Verpflichtung zu zusätzlichen Lizenzgebühren verwenden, weitergeben oder verkaufen. Er ist ebenfalls nicht verpflichtet anzugeben, daß das Programm mit der TBarcode-Komponente entwickelt wurde. Die Lizenzversion oder Teile davon dürfen jedoch auf keinen Fall als Bibliothek oder als Teil einer anderen Bibliothek weitergegeben werden.

6. Technische Informationen für lizenzierte Anwender

Für **lizenzierte Anwender** mit Compuserve- oder Internet-Zugang ist es durch ein spezielles Lizenzierungsverfahren möglich, bis auf weiteres **Gratis-Updates** der TBarcode-Komponente zu erhalten, indem später herausgegebene Sharewareversionen in lizenzierte Fassungen umgewandelt werden können. Sie brauchen dazu nur Ihre lizenzierte Fassung der Ressourcendatei BARCOD16.RES bzw. BARCOD32.RES in die neueste Sharewareversion hineinzukopieren und die Komponente neu zu installieren. Und schon haben Sie aus der neusten Sharewareversion eine neueste Lizenzversion gemacht!

Die neueste Sharewareversion finden Sie jederzeit in den einschlägigen DELPHI-Foren bei Compuserve oder auf meiner Internet-Homepage.

Die URL-Adresse der Autoren-Homepage von J. Schlottke ist:
<http://ourworld.compuserve.com/homepages/schlottke>

Bitte:

Bewahren Sie immer eine Kopie der Original-Komponente als Backup in Reserve!