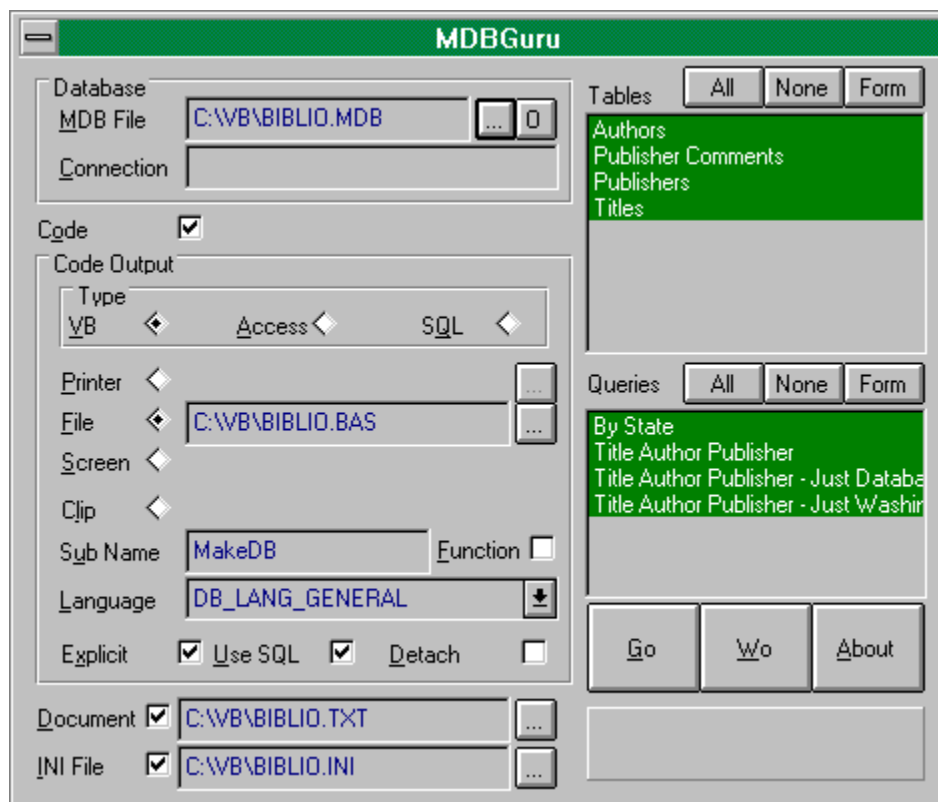# Contents

MDBGuru (previously know as MDB->BAS)   reads a MS Access MDB file (version 1 or 2) or any ODBC source, and creates Visual Basic, MS Access 2 or SQL code to recreate that database.

You can now also create any type of VB form using the new - to version 4 - template based forms generation facility.

- Operation
- Creating Forms
- Creating Form Templates
- Limitations
- Author, author!
- Registration
- Support

Operation
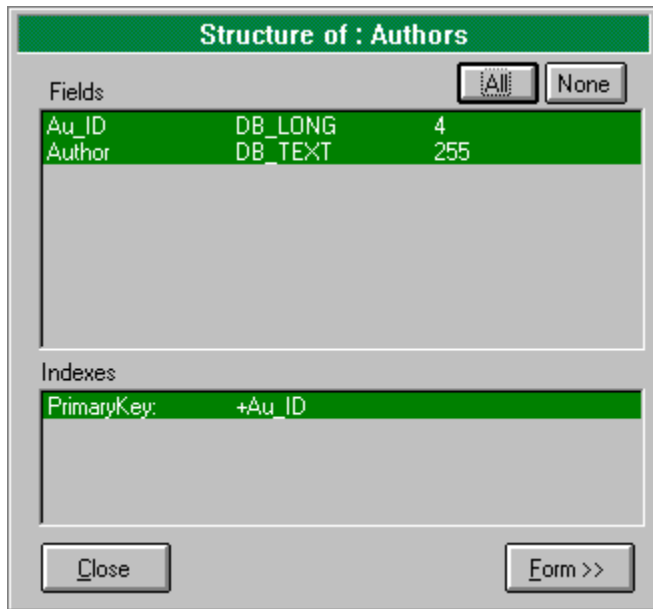
Click on the control you need help with.

☐ Creating Forms

Creating Forms

To create a form with MDBGuru, double click a table or query from the main form.   MDBGuru will read the structure of the table and display it in the Table Structure window.   (It is not possible to create forms for Parameter queries).
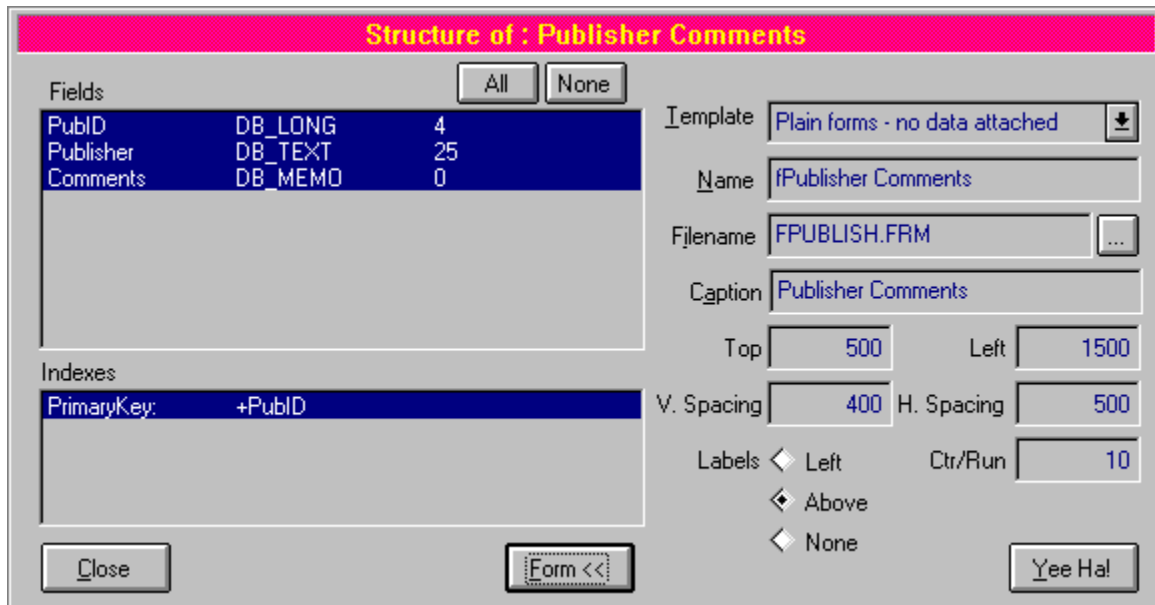
Note: You cannot view the stucture of a Parameter query.

Structure of : Authors

Fields   [All] [None]

| Au_ID | DB_LONG | 4 |
| Author | DB_TEXT | 255 |

Indexes

PrimaryKey: +Au_ID

[Close]   [Form >>]

Click the Form button to access the Form Generation options.

Structure of : Publisher Comments

Fields   [All] [None]

PubID	DB_LONG	4
Publisher	DB_TEXT	25
Comments	DB_MEMO	0

Indexes

PrimaryKey: +PubID

Template | Plain forms - no data attached ▼
Name | fPublisher Comments
Filename | FPUBLISH.FRM [...]
Caption | Publisher Comments
Top | 500   Left | 1500
V. Spacing | 400   H. Spacing | 500
Labels ◇ Left   Ctr/Run | 10
    ◆ Above
    ◇ None

[Close]   [Form <<]   [Yee Ha!]

The Template combo displays a list of all available form templates.

Creating Templates

Creating your own templates is very simple.

- Create a form template
- Modify the form template
- Create a TEM file
- Create a control library
- Create control templates

The last two steps would typically only be need to be done once.

Create a form template

1   Use VB to create a form.   Include all the standard non-data aware controls (eg buttons, frames etc).

2   If the form is to be used to browse a data source, add a data control.

3   Add any required code.

4   Save the form as text with a file extension of .FRT.

Modify the form template

A typical unmodified form might look like this:

```
VERSION 2.00
Begin Form EditForm
    Begin CommandButton bOk
        Caption     =   "&Ok"
        Default         =   -1  'True
        FontBold    =   0   'False
        FontItalic =   0   'False
        FontName    =   "MS Sans Serif"
        FontSize    =   8.25
        FontStrikethru  =   0   'False
        FontUnderline   =   0   'False
        Height          =   375
        Left        =   3000
        TabIndex    =   4
        Top         =   420
        Width           =   975
    End
End
Option Explicit

Sub bOk_Click ()
    Unload Me
    Exit Sub
End Sub
```

To prepare this template for use with MDBGuru, you need to insert two symbols.

Firstly, the name of the form is determined by MDBGuru, so EditForm above should be replaced with <<formname>>.   You also need to place a symbol where the data field control are to be inserted.   This symbol is <<fieldcontrols>>.

The modified form template would then look like:

```
VERSION 2.00
Begin Form <<formname>>
    Begin CommandButton bOk
        Caption     =   "&Ok"
        Default         =   -1  'True
        FontBold    =   0   'False
        FontItalic =   0   'False
        FontName    =   "MS Sans Serif"
        FontSize    =   8.25
        FontStrikethru  =   0   'False
        FontUnderline   =   0   'False
        Height          =   375
        Left        =   3000
        TabIndex    =   4
        Top         =   420
        Width           =   975
    End
```

```
    <<fieldcontrols>>
End
Option Explicit

Sub bOk_Click ()
    Unload Me
    Exit Sub
End Sub
```

Create a TEM file

The TEM file is read by MDBGuru and is used as the starting point for creating form.   It's format is a standard INI file format and it includes the following sections and fields:

Form Section

Name	Displayed in the template combo box
Form	Name of the form template file
Controls	Name of the control library file
Labels	Position for control labels (None, Above or Left)
Top, Left	Top left position for first field control
VSpacing, HSpacing	Vertical and horizontal spacing between field controls
CtrPerRun	Controls per vertical run. Controls are added to the form in vertical runs.

ControlTypes Section

LABEL	Control tag for field labels
DB_BOOLEAN	Control tag for Yes/No fields
DB_BYTE	Control tag for Byte fields
DB_INTEGER	Control tag for Integer fields
DB_LONG	Control tag for Long fields
DB_CURRENCY	Control tag for Currency fields
DB_SINGLE	Control tag for Single fields
DB_DOUBLE	Control tag for Double fields
DB_DATE	Control tag for Date/Time fields
DB_TEXT	Control tag for Text fields
DB_LONGBINARY	Control tag for OLE fields
DB_MEMO	Control tag for Memo fields

A typical template would look like:

```
[Form]
Name            = Browse a datasource
Form            = browse.frt
Controls        = browse.ctr
Labels          = Left
Top             = 800
Left            = 500
VSpacing        = 400
HSpacing        = 1500
CtrPerRun       = 8

[ControlTypes]
LABEL=Label
DB_BOOLEAN=CheckBox
DB_BYTE=TextRight
DB_INTEGER=TextRight
DB_LONG=TextRight
DB_CURRENCY=TextRight
DB_SINGLE=TextRight
DB_DOUBLE=TextRight
DB_DATE=Date
DB_TEXT=Text
DB_LONGBINARY=Image
DB_MEMO=TextMulti
```

Create a control library

The control library is used to associate control tags with control classes.

```
[Label]
Class            = Label
Properties       = labels.pro
Code             =
Prefix           = lbl

[Text]
Class = TextBox
Code             = dtxtbox.cod
Properties       = dtxtbox.pro
Prefix           = ctr

[TextRight]
Class = TextBox
Code             = dtxtbox.cod
Properties       = dtxtboxr.pro
Prefix           = ctr

[TextMulti]
Class = TextBox
Code             = dtxtbox.cod
Properties       = dtxtboxm.pro
Prefix           = ctr

[Date]
Class = DateEdit
Code             = dateedit.cod
Properties       = dateedit.pro
Prefix           = dte

[Image]
Class = Image
Code             = dimage.cod
Properties       = dimage.pro
Prefix           = img

[CheckBox]
Class = CheckBox
Code             = dcb.cod
Properties       = dcb.pro
Prefix           = chk
```

The control library is an INI file.

When MDBGuru adds control to the form, each control is always part of a control array.   Thefore, each class of control must have a unique name.   MDBGuru uses the control tag prefix and the form name to name controls.   Each control class is indexed uniquely by the use of the prefix and class name.

Create control templates

The code field of controls listed in the control library is used to specify the name of a file containing code for the control.   This code is added to the form only once for each control of the specified tag.   The following code file is used for TextBox controls:

```
; gotfocus to change background color
sub  <<controlname>>_GotFocus(Index as integer)
 <<controlname>>(Index).BackColor = HICOLOR
 <<controlname>>(Index).SelLength =
      len(<<controlname>>(Index).Text)
 <<controlname>>(Index).SelStart = 0
end sub

; lostfocus to change background color
sub  <<controlname>>_LostFocus(Index as integer)
 <<controlname>>(Index).BackColor = NORMCOLOR
 <<controlname>>(Index).SelLength = 0
end sub
```

The Properties field of controls listed in the control library is a filename.   The file contains the properties for the control:

```
; default textbox properties
Height         =    315
Index          =    <<index>>
Left           =    <<left>>
MaxLength      =    <<fieldlength>>
TabIndex       =    <<tabindex>>
Tag            =    "<<fieldname>>"
Text           =    "<<fieldname>>"
Top            =    <<top>>
Visible        =    -1    'True
Width          =    2400
```

Lines in properties or control files that start with a semi-colon are comments and are not included in the actual form file.

See also:
MDBGuru fields

MDBGuru Fields

MDBGuru will replace the following symbols in control and form templates:

buildSQL   For large multiline queries queries, use this in the form_load event
caption   Window caption as entered
connectstring   Connection string as entered
controlname   Name of the control, based on the control prefix and the datasource form name.
databasename   Name of the database file with full path
fieldcontrols   All controls as generated (for the form control)
fieldlength   Datasource table field length
fieldname   Datasource table field name
formfile   Form filename as entered
formfrx   Form FRX filename (based on form filename)
formname   Form name as entered
index   Current control array element index
labelindex   Current label control array element index
left   Left co-ordinate position of the current control
password   Password as entered (if entered!)
recordsource   Name of the table or the SQL from the query
tabindex   Tab index for the current control
top   Top co-ordinate position of the current control
user   User name as entered (if entered!)

All fields must be bracketted with << and >>, eg: <<index>>.   Case of the fields is not significant.

Limitations

Requires Jet Engine v 2.0 and Compatability Layer

This version of MDBGuru requires the Jet Engine 2.0 (or later) compatability layer for VB to be installed.   The compatability layer is available from the usual Microsoft sources and the ODK CD Rom.

This version has not been tested with the Jet Engine v 2.5.

Registration

The unregistered version of MDBGuru will only process the first two tables in a database file.

Author

MDBGuru is produced by:

Peter G Jones
Zero-1 Software Ltd
      53 Prestons Road
      PO Box 4475
Christchurch
New Zealand

Ph/Fax: 64, 3, 352 6481
CIS: 100032,40.
Internet: jonesp@zero-1.co.nz

Registration

MDBGuru can be registered thus:

- Compuserve SWREG: 3070 for EXE only
- Compuserve SWREG: 3738 for EXE & Source.
- Compuserve SWREG: 9146 for Upgrade from previous versions, EXE or Source.
- Visa or Amex: Send me your credit card details via EMail.   My CIS address is 100032,40. My Internet address is jonesp@chch.planet.co.nz You should send: Card number, expiration date, card holder name, address and your EMail address.

I will EMail you MDBGuru as soon as payment is received.

The charge is $US25 for the EXE only, $US50 for EXE & Source Code.   Upgrades are $US20.

Support

REGISTERED users may contact the Author by EMail only for Tech support.   Typically you can expect a reply within 36 hours.

Un-Registered users may contact the Author by EMail for technical questions or otherwise but don't expect the same response rate.

Code Generation

Uncheck this option if you do not want source code output at all.

Connection String

A connection string may be required if you are connecting to a remote data source via ODBC or you wish to specify user name or passwords.

Code Generation Destination

Select the destination for the generated code.

You can use the Pick buttons to choose a printer or file.

If you choose screen, you can then cut and paste the code to another application.   If the database is especially large, the screen output textbox may not hold all the code.

If you choose Clipboard, the code will be copied directly to the clipboard.   You can also copy code to the clipboard when you choose screen output, using the normal copy keys.

Detaching Tables

This option allows you to create code for attached tables, as if they were really stored in the MDB. This will, for example, enable downsizing of server based databases (SQL Server etc) to Access MDB's.

Document

Check this option if you require a text file description of the database.

You can specifiy the file name by typing into th etext box or by using the Pick button.

Explicit

For VB or Access, check this option to insert the OPTION EXPLICIT command in the generated code.

INI Filename

Check the INI File option if you wish to use the data dictionary routines to create the database in your application.

Enter the name of the INI file to generate.

WARNING!

The INI File will be DELETED prior to creation so be very careful what name you enter here. For example, WIN.INI or SYSTEM.INI would not be a good choice.

Language

Select the locale for the created database.

Database Filename

You should enter the name of the database file (MDB) here or use the Pick button to choose a file from disk.

You can also click the "O" button to use ODBC to connect to a remote data source.

Queries

This list displays all the Query Definitions stored in the selected database.

Only the selected querydefs will be included when generating code.

See also:

[Table List](#)

Tables

This list box displays a list of the tables in the currently selected database.

Only the selected tables will be included when generating code.

To generate a form for a table, double click on an entry in the list.

See also:

Query List

Sub Name

Enter the name of the Subroutine to generate.

Code Type

Select the type of output required.

- VB: Visual Basic 3.0
- Access: MS Access 2.0
- SQL: Pure MS SQL satements

Use SQL

Select this option to generate code that uses SQL to create the database, rather than Objects.

See also:

[Limitations](#)